Efficient Waste Segregation Using Deep Learning Technique

MSc Research Project
Data Analytics

Samiksha Shirbhate
Student ID: X21213615

School of Computing
National College of Ireland

Supervisor: Cristina Hava Muntean

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Samiksha Shirbhate
Student ID:	X21213615
Programme:	Data Analytics
Year:	2023
Module:	MSc Research Project
Supervisor:	Cristina Hava Muntean
Submission Due Date:	14/12/2023
Project Title:	Efficient Waste Segregation Using Deep Learning Technique
Word Count:	8235
Page Count:	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Samiksha Shirbhate
Date:	31st January 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Efficient Waste Segregation Using Deep Learning Technique

Samiksha Shirbhate
X21213615

Abstract

Waste generation is increasing day by day and causes harm to the nature and growth of the economy as the maximum of things are directly dumped instead of recycled. On the other hand, the recycling rate is rising but so is the waste, and thus the recycling rate should be as maximum as possible. This can be achievable only if the waste is properly segregated so that this classified waste is sent to recycling. Machine learning has the potential to enable the creation of highly accurate models that help to achieve this objective. Numerous studies have been carried out to automate trash classification with minimal human involvement. Therefore, the goal of this research is to contribute to the creation of a better model. So the model proposed is a novel technique which a combination of two models YOLOR-YOLOv8. As a result, there are two models that have shown great performance at 70 epochs which are YOLOv8 and the combined model YOLOR-YOLOv8. Whereas YOLOR_P6 is also improved over epochs.

Keywords— Waste segregation, deep learning, YOLOR, YOLOv8, YOLOR-YOLOv8, CNN-based models, YOLOv series model, YOLOR_P6, YOLOR_W6, implicit knowledge, explicit knowledge

1 Introduction

The prime concern towards clean earth is the way to manage the waste that is generated every day in billions of tons. It was estimated that waste generation is rising, and is around 2.24 billion tons per day. And if this goes on, it is expected to increase by 73% in 2050 [1]. Thus, this is high time to manage waste efficiently to lower waste and increase the recycling of waste. The more effectively the garbage is separated, the more valuable it is for recycling. Therefore, the main purpose of this research is to create a deep-learning method for garbage classification. This problem is also addressed by researcher Chepa et al. (2021) who mentions that the situation gets worse as the population increases which can lead to an increase in waste collection. Hence there is a requirement to design an automated system to handle waste efficiently without much human interaction as exposure to waste can cause harm to humans. Numerous researchers have made contributions to waste management; they have produced a machine-learning model to separate waste using various methods and have demonstrated advancements in this field. Even said,

[1] Solid Waste Management: `https://www.worldbank.org/en/topic/urbandevelopment/brief/solid-waste-management`

there is room for improvement in terms of developing a better paradigm for segregation that works better.

As the segregation of waste cannot be possible manually as it can harm humans and will take an enormous amount of time, this process needs to be automated. And the best way to automate is to develop a machine learning model that learns with previous data and predicts based on the past. Many other researchers have contributed to this area and have done a great job. The researcher Kannangara et al. (2018), has created models, namely decision tree and neural network which predicts the waste generation and how much of it can be recycled. This researcher has trained the models based on municipal solid waste data from Canada and concluded that out of two models, the neural network has shown better results with less error rate. And thus its prediction helps municipalities to evaluate further. This research is based on the amount of waste generated while other researchers have worked on image processing of waste data to segregate. The research project Kumar et al. (2020) has performed techniques YOLOv3 and YOLOv3-tiny and has produced positive outcomes yet there are a few limitations. The researcher has tested multiple iterations up to 50000 for both YOLOv3 models and stated that the YOLOv3 has performed better. The model YOLOv3-tiny at one point of iteration which is 12000, stabilizes and reaches the optimal values as 51.95% while in contrast to another model which achieves 94.99%. The researcher also faces the complexity of the difference in the nature of the material which makes it difficult to identify the object.

In search of a better model to fit in for waste classification, this research has taken the initiative to use a unique technique YOLOR which has not been implemented in this area. YOLOR stands for ”You Learn One Representation”. This technique is discovered to fasten the process and detect the object more accurately [2]. In addition, it not only identifies a single object but identifies different types of objects even with different classes. YOLOR provides a unified network that encodes explicit knowledge along with implicit knowledge which predicts the multiple tasks from the image. Explicit knowledge is knowledge that is obtained by the model using neural networks. Implicit knowledge can be identified by humans which describes the behavior, that makes the YOLOR technique better than others. To enhance and explore more in this technique further, the YOLOv8 model is integrated with the technique YOLOR. Thus, the new technique to be designed in this research project is named YOLOR-YOLOv8. YOLOv8 is the recently published version of the YOLO (YOLO stands for You Only Look Once) model. YOLOv8 is a great choice for a variety of object recognition and tracking, instance segmentation, image classification, and pose estimation tasks because of its quick, precise, and user-friendly architecture [3]. YOLOv8 is the latest version of the YOLOv series and proves to be better than the previous model it majorly focuses on fastening the training process and reducing the complexity that occurs in traditional YOLOv series models. It not only simplifies the process but it accentuates the most essential features. When both of these techniques are combined which are robust independently, the resulting strategy might turn out superior to either one used individually. Let us examine the methodology 3 part to have a detailed understanding of how this technique is constructed.

In an effort to develop a better model, the research study attempts to apply the novel

[2] YOLOR: https://viso.ai/deep-learning/yolor/

[3] YOLOv8: https://github.com/ultralytics/ultralytics

combination technique known as YOLOR-YOLOv8. As a result, the research aims to address the question that follows.

Research Question

Whether the deep learning technique brings a change in detecting multiple types of objects in a single image and classifying waste?

The following section reflects the structure of this research paper. This document is organized into the below section which shows a one-liner description of steps followed.

1. *Introduction:* 1 This section introduces the scope of this research and also the motivation behind this research paper. This gives a summary of the document.
2. *Related Work:* 2 This section discusses the previous related work that has contributed to this research paper. This research learns about various types of techniques that bring unique contributions in this area.
3. *Design Specification:* 4 This section explains the architecture of the new functionality followed in this research project.
4. *Implementation:* 5 This section includes the research regarding the development of the deep learning model and its performed output.
5. *Evaluation:* 6 This section discusses the comparative analysis of the results of deep learning models designed in the research paper.
6. *Conclusion and Future Work:* 7 This section will be an answer to the objective and results achieved from analysis and work done. This section will also discuss the future work for this project.
7. *References:* 8 This section shows the list of research paper referenced which has given a part of contribution to better understand the work done in this area as well with the deep learning technique.

2 Related Work

This section includes a discussion and critical analysis of related work done. Many researchers have contributed to this area, and this section briefly examines it according to this paper's angle. The research paper Bobulski and Kubanek (2019) initiates building a convolutional neural network model with different layers and has discussed their comparison concerning the resolution of images. The researcher has built CNN models with 15 layers and 23 layers based on the Alex-Net network using different image resolutions, which are 120x120 pixels and 227x227 pixels. This researcher has discovered a unique way to reduce the complexity of the model by researching different resolutions of the image dataset. This research paper focuses on how the high resolution of images can affect model complexity and calculation while being easily identified with the lower resolution images. All models were generated with all the combinations, and the highest accuracy was obtained by the 23-layered CNN model with a resolution of 227x227 pixels, but the computational time is much higher. The second highest accuracy was obtained by a

15-layer CNN model with a resolution of 120x120 pixels at the early stages of epochs. Observations noticed from this are that the more the number of layers, the better the results, and the other comprehensive analysis acquired is that lower resolution of the image reduces complexity and provokes better results. Still, it also depends on the type of images. If the object to be detected is very tiny or with a complicated design in the image then it might affect the identification of the object and might lower the accuracy. Thus, it also depends on the image dataset used for the model training.

The researcher Sreelakshmi et al. (2019) presents a capsule neural network on two different types of datasets. Dataset 1 is collected from public places, and Dataset 2 is collected from private materials. Capsule-Net has a unique quality for predicting an item; it splits into small packets and identifies its features with rotation, tilt, or other orientations. It overcomes the limitations of conventional CNN. The research paper has compared the capsule-net and conventional CNN models. It was observed that the Capsule-Net has shown better results for both datasets even though the difference is less. The capsule network overpowers the CNN basic model because of its unique feature. The initiative of comparison between two different datasets that are captured in public and private areas is to better understand the object detection found in different areas. Although the difference in accuracy is less, the dataset with public data was detected better than the dataset with private data. This might be because the public data could be more of a similar type but there might be variation in objects to some extent. Public waste data such as plastic bags, water bottles, paper, etc can appear commonly while private waste data like different types of electronics items, different home design items, various stationary objects, food waste, etc can appear differently. Thus different types of datasets with the same models can have different results.

The Literature review is further broken down into sub-sections according to the model family like CNN, YOLOv series, etc. Each subsection discusses the pros and cons. There is a summarised table for each subsection to have a quick overview. As obvious three subsections are CNN based model, the YOLOv series model, and the YOLOR model, this shows the incremental progress observed over time. These subsections will briefly tell about the critical and comparative observations found by different researchers in this area or related to the technique used in the research project.

2.1 Convolution neural network models and their comparison

The relevant CNN model research papers are covered in this section. The research paper Thanawala et al. (2020) has designed a unique CNN-based model and five standard CNN architectures and assessed the comparison between them. The proposed model consists of five convolution layers, four max-pooling layers, one flattened layer, and two connected layers. According to the researcher, the activation function introduces nonlinearity which helps to learn the complex function and provides distinctiveness during back-propagation. and thus uses ReLu as a function and derivative with the final layer softmax activation function which limits the output to 0 and 1, also created the physical model for waste segregation with an 8051 microcontroller, ultrasonic sensor, GSM and GPS module, servo motor, and IR sensor. The data is collected to the ultrasonic sensor, constantly transferred to the microcontroller, and then sends the signal to the base station GSM module. GPS module is used to get latitude and longitude to send the exact location of the bin to the concerned authority. IR sensors are used with a servo motor to rotate the lid of the garbage bin. Then the researcher compared these proposed models with famous CNN models like VGG-16, Res-Net, Mobile-Net, Inception-Net, and Dense-Net. All the models have been tested with different epochs and with and without augmentation. The proposed model has performed well but the accuracy is less when compared to another five models with the highest accuracy achieved by Mobile-Net. This research not only brings the hardware models but also tries to improve the architecture of the CNN-based model by increasing a few layers like the convolution layer, max-pooling layer, and connected layers. However, increasing the good features of the CNN model made it complex and thus it might affect detection.

The research Gyawali et al. (2020) reflects the comparative analysis of multiple deep learning techniques such as ResNet50, ResNet18, and VGG16. Before training, it is important to build a strong foundation that prepares the image data and pre-train the model. The researcher has chosen to perform transfer learning to leverage preexisting knowledge from a large dataset. This technique involves the enhancement of the classification of waste by knowledge gained by one task to improve the performance of related tasks. Observing the weakness of all the models, the researcher came up with the stronger model ResNet18. ResNet50 model which introduces more layers does not give good results and so is the VGG16 model while this also depends on understanding the dataset of which type of prediction is held. It was noticed from the train loss graph that ResNet50 and VGG16 models are not as great as ResNet18 as per the dataset used in this study while also, observed that ResNet18 is neither overfitted nor underfitted and achieved better accuracy which is 87%. Examining from this research, accuracy could have been better if fine tuning is performed on the image dataset along with transfer learning so that model training will done on per processed and fine-tuned dataset and thus performance can be improved.

The data preparation in the research paper Shetty (2022) has been built from a unique angle and has done comparative analysis with models VGG19, Xception, and Inception ResNet-v2. Each model expects the individual requirements of data pre-processing of images, and thus researcher not only compares the model accuracy but carries different data pre-processing steps as per the requirements of each model. Thus each model gives the result on its own merits. The ideal scenario of freezing a few layers and using the pretrained weights can be beneficial in transfer learning where as the per-trained model trained on a larger dataset can adapted by a smaller dataset to restrain the specific feature of the dataset. Inception ResNet-v2 is a combined model of Inception and Resnet which makes it powerful but also adds up complexity and thus takes longer computational time for training but gives good accuracy. On comparing, the VGG19 model rules over other models by giving the highest accuracy and taking decent computational time.

Another researcher Pandey et al. (2023) proposed a strong convolution neural network model like DensNet-169 which has 169 layers that include extracting features, pooling layers for summarizing features, and fully connected layers for classification. Along with this, it was made stronger by applying data augmentation on the image dataset. This artificially increases the image dataset by creating multiple images from one existing image such as flipping, zooming, and rotating to avoid overfitting. As discussed in paper Gyawali et al. (2020) it faced overfitting issues, this might be a solution to avoid. Along with this, two vital techniques are used during training such as early stopping and checkpoints. This helps to get rid of time-consuming events like overfitting and running the same runs repeatedly. This model is compared with models VGG16 and Logistic regression and it concluded that DenseNet169 achieved the highest accuracy.

The table 1 summarises the research paper who has implemented CNN based models. It also shows the column as "Better model" where respective author found out particular model as superior among the other model chosen by authors which are listed in column "Technique"

Table 1: CNN based models comparison summary

Authors	Techniques	Better model
Thanawala et al. (2020)	Custom CNN, VGG16, ResNet, MobileNet, InceptionNet & DenseNet	MobileNet
Gyawali et al. (2020)	ResNet50, ResNet18 & VGG16	ResNet18
Shetty (2022)	VGG19, Xception & Inception	VGG19
Pandey et al. (2023)	DenseNet169, VGG19 & logistic regression	DenseNet169

2.2 YOLOv series models and their comparison

This section addressed related YOLOv series research papers. Apart from CNN-based models mentioned in the above section 2.1 in this area has proven to give better accuracy, but there is always a scope for betterment, and the YOLOv series shown better and fastest than CNN models, although the YOLOv series is also CNN based but with its unique identification technique without making the model much complicated by adding more and more layers. The paper Patel et al. (2021) proposes YOLOv5M and compared with EfficientDet-D1, SSD ResNet-50 V1, Faster R-CNN ResNet-101 V1 and CenterNet ResNet-101 V1. YOLOv5M is being introduced with improvements in model size, computational speed, and efficiency in terms of memory and accuracy. This paper compared this model with developed CNN models to make a fair comparison. And concluded that YOLOv5M is the most efficient variant of all models compared with. Regardless, the researcher trained the model on 500 images, while the researcher should have increased the dataset size to know how robust the model YOLOv5M is actually as 500 images might not covered all the types of features. The affirmative aspect of this study is that all CNN-based models were twisted to make them more powerful whereas Faster R-CNN ResNet-101 V1 and CenterNet ResNet-101 V1 models are closer to the YOLOv5M model.

The study Andhy Panca Saputra (2021) performed YOLOv4 and YOLOv4-tiny with Darknet-53 and has done a comparative analysis. It is observed that YOLOv4 has shown better outcomes in terms of prediction and mAP whereas YOLOv4-tiny is much faster in terms of prediction. The dataset is classified into 3 classes which are glass, paper, metal, and plastic. the percentage of correctly identified is better for all classes while for plastic and metal are less than the others, it might be because plastic and metal come with various variants of materials, colors, or something stickers on them. It was also observed that the distribution of images is not equal, and the smaller number of images occurs under plastic and metal with 482 and 410 images respectively. This might be a reason for the lesser correct percentage for plastic and metal. This might be solvable by training more images of this class or doing hyper-parameter tuning with more images available with various features so that the model is trained to identify better.

The paper Lin (2021) modified the YOLOv4 model to reduce its complexity and increase its efficiency. It adapted SqueezeNet fire module with darknet convolution layer for the balancing where the 3*3 convolution layer is replaced with a 1*1 convolution layer. It also uses several YOLOv4 capabilities, which the researcher claims have benefits. Mainly, the researcher has focused on those features which are beneficial according to the dataset. Every dataset is different, and the same solution might not produce the best outcomes all the time. It is important to understand the dataset the model is trained in, and the researcher has followed this modified accordingly and named the modified YOLOv4 model as YOLOv4 green. This was compared with YOLOv4 and YOLOv3 models and as a result, achieved as better model.

Table 2 shows the summary table for all the research paper who has worked with the YOLOv series model and done the comparative analysis.

Table 2: YOLOv series models comparison summary

Authors	**Techniques**	**Better model**
Patel et al. (2021)	YOLOv5M, EfficientDet-D1, SSD ResNet-50 V1, Faster RCNN ResNet-101 V1, CenterNet & ResNet-101 V1	YOLOv5M
Andhy Panca Saputra (2021)	YOLOv4, YOLOv4-tiny & Darknet-53	YOLOv4
Lin (2021)	YOLOv4 green, YOLOv4 & YOLOv3	YOLOv4 green

2.3 YOLOR models and their comparison

The research article that used the YOLOR model is included in this section. Since this method isn't used in the "waste segregation" area, the research studies in this section include references to the YOLOR model in other contexts. The paper Zhang et al. (2021) has done a comparative analysis between the two different variants of YOLOR, namely YOLOR-P6 and YOLOR-W6. YOLOR-P6 is the fastest variant while YOLOR-W6 is wider and has more parameters than YOLOR-P6. The pre-trained model trained on the COCO dataset is used and subsequently fine-tuned on the Waymo dataset which is used in this study. The COCO dataset is a large dataset that is built for object detection, segmentation, and various other tasks. This dataset has covered around 80 classes. If the pre-trained model on the COCO dataset is tested on another dataset, this might produce a correctly identified object to be predicted and fine-tuning can help to some extent but would not work in all cases like if the classes of the desired dataset do not match most of the classes of COCO dataset. If this pre-trained model is used for waste segregation, then it might not mainly focus only on waste classification.

The paper Tran et al. (2022) performed YOLOR and YOLOv5 models. These models are trained at 500 and 100 epochs and observed the increase in accuracy decrease in train and validation loss. Also at first 100 epoch there was significant rise while after 100 till 500 improvement is slow for both the models. Talking about the losses observed over the epocs, YOLOv5 models has less loss comparatively. But the highest accuracy achieved by YOLOR model. The study Yan et al. (2022) modifies the YOLOR model with the YOLOv4-csp on darknet layer to lower down the computation. Where it merges YOLOv4-csp with the implicit knowledge. Thus, this approach increased the accuracy by 3.5%. And has done comparative analysis with YOLOR-p6, YOLOR-w6, YOLOv4-csp, YOLOv4-p6. It also has trained and tested on multiple dimensions of images and occurred the images with low and medium resolution is fast and has better accuracy surprisingly. This is obvious that for know dimensions , computation units are less utilised and thus is fast. On the other hand, it also shows better accuracy. Well this is because implicit knowledge might have limited improved in detection on the large dimension when compared with smaller dimension images.

In Table 3 summarises the research paper referred to who has implemented the YOLOR model and has done comparative analysis.

Table 3: YOLOR models comparison summary

Authors	**Techniques**	**Better model**
Zhang et al. (2021)	YOLOR-P6 & YOLOR-W6	YOLOR-P6
Tran et al. (2022)	YOLOR, & YOLOv5	YOLOR
Yan et al. (2022)	YOLOv4-csp,YOLOv4-p6, YOLOR-p6, YOLOR-w6	YOLOR-P6

3 Methodology

This section initiates with the actual process of this project and describes more about the steps followed and the reasoning behind every step. This research is based on waste segregation and it is further classified into 42 classes listed in section 3.2. This section mainly discusses the rationale for selecting the proposed approach and how to implement it to carry out the proposed investigation. Additionally, a comparison analysis between the suggested model and the other models will be shown to determine whether the proposed model performs better.

Commencing with this innovative approach, the section is further divided into subsections, as demonstrated below.

3.1 Understanding the Project and Application Domain

The initial phase of this research is to understand the proposed technique and how it can be implemented practically. And to better understand the deep learning techniques and their applications on different types of data is achieved by reading the related papers and articles. In accordance with the literature review conducted as part of this project, the proposed approach has not been implemented in this area "Waste Segregation". As the technique introduced in research YOLOR-YOLOv8 needs a high GPU and more compaction units, thus this project is carried out in Google Collab as it provides a paid subscription to get the desired environment. This helped to save quite a lot of time in executing more than one model for more epochs. This requires a high computation unit as it has a large number of image data with its label.

3.2 Data Exploration and analysis

The dataset used in this project is "YOLO Waste Detection Image Dataset" from the roboflow website ProjectVerba (2022) which is a public dataset. This image dataset consists of 42 classes such as cans, glass bottles, plastic bottles, cups, plastic bags, cardboard, metal, wood, etc. It also consists of an annotation file for each individual image. There are multiple formats of annotations. As the models, YOLOR and YOLOv8 used in this study need to have annotation files in either YOLOV5 or YOLOv8 format and thus the format used for this research is YOLOv5 format. The roboflow website provides various types of formats including the YOLOv5 format. The images and their annotation files are split into train, test, and valid folders. In all, there are 11466 images in the train, 5456 images in the test, and 1092 images in the valid folder. When examining the dataset's class balance, it is evident that certain classes are overrepresented while some are underrepresented. On the positive side, every image has a label that identifies the class of multiple objects within it, together with the precise coordinates of each object.

3.3 Prepossessing of image data

This section describes the preparation of the images of the waste dataset. Before training a model, it is crucial to know the data. As Roboflow website provided the data with different resolutions but most images are with resolution 263*225. However it also provided the label files for each image present in the dataset, and thus the decision has been made not to augment the data, as it might hamper the location of objects present in the images so the label files make no sense. However, there are a few tools to create label files manually for all images individually. In all the "YOLO Waste Detection Image Dataset" consists of 15K images, so it will be a time-consuming task. There is another way to create a label that is by auto-annotating images using a pretrained file, but the available pretrained file is not trained as per the classes that exist in this dataset.

So next necessary step would be rearranging the dataset structure according to the model requirements. Models YOLOR and YOLOv8 are expected to have different structures and should be present in the same directory. Thus two different structure of the dataset is uploaded on Google Drive on a respective model directory. For training YOLOR, the GitHub repository [4] is uploaded on Google Drive to fetch the Python files and other requirement files for execution. And the YOLOv8 model is trained by importing the ultralytics package.

[4] https://github.com/WongKinYiu/yolor

3.4 Model Building

The novel technique used in this research on the image dataset is YOLOR with the interpretation of YOLOv8. Before moving on the the methodology, there are a few advantageous features based on selecting the two individual techniques YOLOR and YOLOv8, aiming to get better results. The following are the points in favor of the combined technique YOLOR-YOLOv8 and how individual techniques can benefit.

1. **YOLOR:**

- YOLOR technique includes implicit knowledge that learns which is present but not visible but is identified by human behavior. Thus, this model can determine items based on human experience to some extent.
- It performs multiple tasks across different areas which depicts the cognitive behavior of humans.
- YOLOR is an improved model of YOLOv7 and it proves to be better and faster than any other YOLOv series model.
- Figure 1 shows the working architecture of model YOLOR which shows combine unified network.

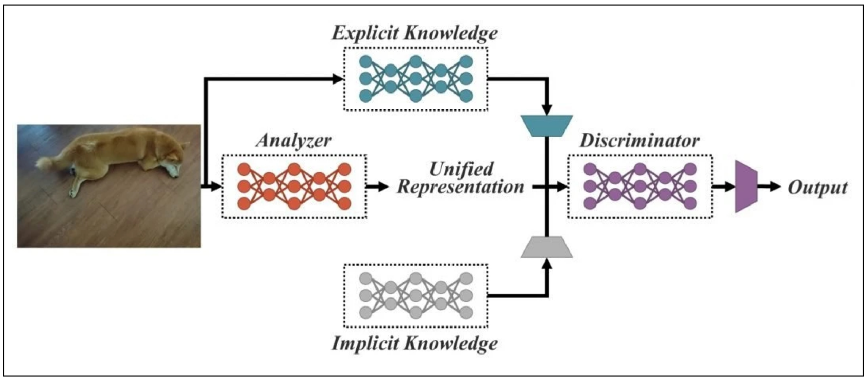


Figure 1: YOLOR working with multi-task learning, Source: Wang et al. (2021)

2. **YOLOv8:**

- YOLOv8 is designed to simplify the process which focuses on highlighting the most relevant features from the image while giving better efficiency and speed. It identifies the object at different scales.
- It can handle different sizes of images, other challenging situations like blurred images, intricate datasets and also provides various model versions for certain tasks and circumstances.
- It's a novel neural network design that makes use of the Path Aggregation Network (PAN) and Feature Pyramid Network (FPN), together with a new labeling tool that makes annotation easier.
- To put in briefly, it is a fusion of speed, accuracy, flexibility, and consistency.

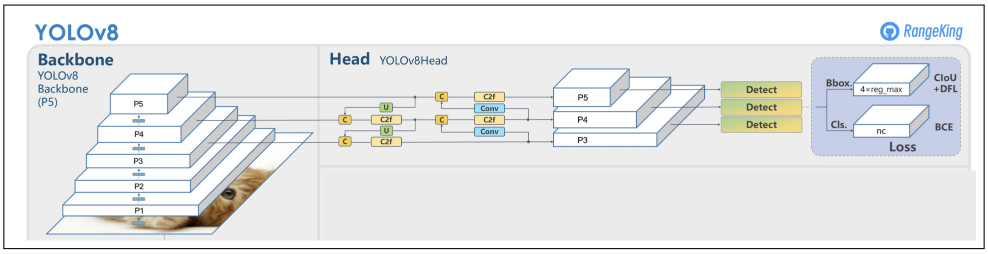


Figure 2: YOLOv8 Architecture, Source: Reis et al. (2023)

- Figure 2 shows the architecture of model YOLOv8.

These two models are trained individually. Then these models are combined through ensemble learning by using trained files generated during the training. The detailed design and execution are followed in sections 4 Design Specification and 5 Implementation.

4 Design Specification

This section describes the design of individual models and their training process. The main purpose of this research is to design an efficient model by combining two different techniques YOLOR and YOLOv8. As mentioned earlier, these two techniques are trained individually to get the benefits of both at the same time. Before coming, let's understand how individual model works and are trained.

4.1 YOLOR

As previously noted, YOLOR is a unified network that consists of explicit and implicit knowledge. Explicit knowledge has been used in various models like the CNN models like VGG19 ResNet, etc and the YOLOv series model. As suggested by its name, it employs two separate learning mechanisms to create a singular representation for identifying and locating objects within an image. This technique swiftly recognizes a wide range of objects in a picture and other tasks, with a more meticulous examination to discern every subtle distinction. YOLOR demonstrates proficiency in executing kernel space alignment, prediction refinement, and multi-task learning within a convolutional neural network. The incorporation of implicit knowledge is a novel concept that enhances performance across all tasks by leveraging deeper neural networks capable of subconscious learning, representing human behavior. Thus the conventional neural network which in this case is explicit knowledge can be expressed as 1

$$\begin{gathered} y = f_\theta(x) + \epsilon \\ \text{minimize}\, \epsilon \end{gathered} \tag{1}$$

where y is the task's objective, x is the observation, θ is the neural network's set of parameters, f_θ is the neural network's operation, and ϵ is the error term.

In the training process, ϵ is the term that is wrongly predicted that why recognized as the error term. So as to minimize the error term, the researcher Wang et al. (2021)

replaced the ϵ with the implicit knowledge. Thus extended formula for the unified network can be expressed as follows 2

$$\begin{gathered} y = f_\theta(x) + \epsilon + g_\Phi\left((\epsilon_{\text{ex}}(x), \epsilon_i m(z)\right) \\ \text{Minimize}\, \epsilon + g_\Phi\left((\epsilon_{\text{ex}}(x), \epsilon_i m(z)\right) \end{gathered} \tag{2}$$

where g_θ is a task-specific operation that functions to aggregate or choose information from explicit knowledge and implicit knowledge. Here, ϵ_{ex} and ϵ_{im} are operations that represent, respectively, the explicit error and implicit error from observation x and latent code z. The selection of implicit knowledge is contingent upon the type of dataset and can be produced by vectors, deep neural networks, and matrix factorization. In vectors, each dimension is independent of another dimension and has a single base; in neural networks, each dimension is dependent on another dimension and has single or multiple bases; and in matrix factorization, each dimension has multiple bases and is independent of another dimension. Considering the dataset utilized in this study, neural networks appear to be the most appropriate option Which is represented as 3

$$W_z \tag{3}$$

Where z is the vector as the prior of the implicit knowledge and W is the weight matrix that performs linear/non-linear combination. It possesses several dimensions, all of which are interdependent, this can also be generated as complex. Regardless of the complexity of the implicit model g_θ, it can be reduced to a set of constant tensors prior to the execution of the inference phase, since implicit knowledge is irrelevant to observation x.

4.2 YOLOv8

YOLOv8 is designed on the principle of gaining efficiency without compromising accuracy. It contains various models like detection, segmentation, and classification. This research has chosen to use the detection model based on the objective of this research. Because of its multi-scale nature, it can accept objects of various scales by utilizing three multi-scale detection layers. The key components of YOLOv8 architecture are the head, neck, and backbone Wang et al. (2023). As a part of Backbone, CSPDarknet53 is modified where it down-samples input image five times in order to get the feature at different scales B1 to B5. This forms a pyramid-like structure and also with enhancement in the original CSPDarknet53 model. The neck is inspired by PANet which has PAN-FPN. It then eliminates convolution operation after up-sampling for lightweight architecture. Henceforth, by combining top-down and bottom-up methods, AN-FPN improves semantic information which is both shallow and deep. The head structure has two separate branches for object classification and bounding box regression which uses a decoupled head structure. Distribution focal loss (DFL) and CIoU are utilized for bounding box regression, whereas binary cross-entropy loss is used for classification. Thus to achieve benefit, the head and backbone are combined together through the neck. The CIoU loss is applied to bounding box regression to enhance the model's detection capabilities. EIoU improves on CIoU by treating the length and width separately as penalty terms.

4.3 YOLOR-YOLOv8

YOLOR and YOLOv8 are trained on a YOLO Waste Detection Image dataset individually. Better prediction performance than could be attained from any one of the individual learning algorithms alone is gained by combining these two models using ensemble approaches, which employ several learning algorithms. The Figure 3 is the YOLOR-YOLOv8 working model designed for the proposed approach. Both the models take the same input and train individually while the output is then put together on Ensemble to produce the final detection output.

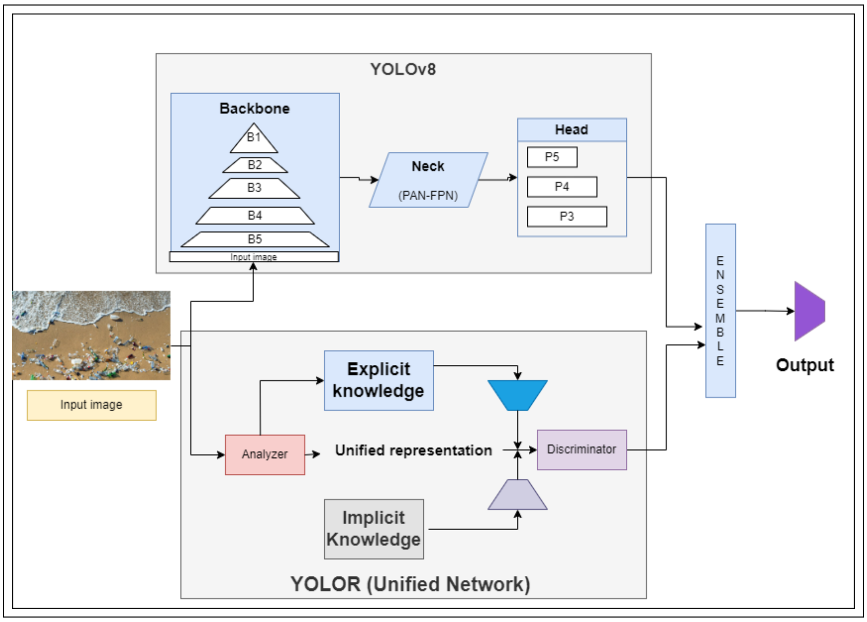


Figure 3: YOLOR-YOLOv8 working architecture

5 Implementation

This section will provide a thorough explanation of the model-training process. The YOLOR model has different variants like YOLO P6, YOLOR W6, etc. While YOLOR P6 is the smallest model among all and is known for speed, efficiency, and accuracy. As the YOLO Waste Detection Image dataset has 42 classes, and thus changes have been made in multiple files such as ”YOLOR_P6.cfg”, ”custom.yaml”, and ”custom.names” to train the model as per the classes present in the dataset. These files help to direct the code while executing the ”train.py” file cloned from the YOLOR git repository. The model is trained for 30 epochs after trying a few iterations and has observed improvement in loss, precision, recall, and m@P. The YOLOR W6 model is also trained for comparative analysis with that of P6 model. YOLOR W6 is known for its larger size and wider than the YOLOR P6 model. As YOLOR W6 creates the pretrained file with the larger files, thus it is only trained with epochs 15.

YOLOv8 is trained from ultralytics library code on the YOLO Waste Detection Image dataset from scratch [5]. YOLOv8 also consists of various pretrained files which are trained

[5] https://github.com/ultralytics/ultralytics

on the COCO dataset with 80 classes in it. Thus, it is also trained with a pretrained YOLOv8 file on the coco dataset and then trained on the custom dataset used in this study. This is done to analyze how both models worked during testing.

6 Evaluation

Five models are implemented in all, whereas out of which four models are trained. Two different variants of YOLOR P6/W6 models and two YOLOv8 models (one trained from scratch/ another trained on a pretrained file) are trained individually. Namely, models are YOLOR_P6, YOLOR_W6, YOLOv8 on dataset, YOLOv8 on dataset and pretrained file, and the combined model YOLOR-YOLOv8. The comparative analysis is performed between these models using m@P, precision, and recall values. Where m@P stands for Mean Average Precision which is a metric to evaluate the performance of object detection model. These model's achievements are followed in the below Experiment subsections in detail.

6.1 Experiment 1: YOLOR_P6

YOLOR_P6 model is the smallest and most efficient YOLOR model whose base channels are set for 128, 256, 384, 512, 640. It is trained starting with 5 epochs which gave average results. Further, it was trained for 30 epochs and observed that loss is decreasing while there is improvement in the values of precision, recall, and m@P which is good. During the training, it created various ".pt" files whereas the model itself elects the file as "best.pt" out of all files. Further, it is then trained for 70 epochs. But at 58 epochs it fails at first run due to time out, as it takes longer time to train. Then this is executed again for 70 epochs, and it is successful and performance is slightly better in terms of precision, recall and m@p.

6.2 Experiment 2: YOLOR_W6

YOLOR_W6 is wider than YOLOR_P6 and has more parameters with base channels set at s 128,256, 512, 768, 1024. This model is trained on the dataset used and noticed that it takes a longer time to execute each epoch than model YOLOR_P6. As the size of the base channel is almost double, it requires more space to store various ".pt" generated during epochs. That is why this model is trained for 15 epochs and gives nearly less results when compared with YOLOR_P6 model at epoch 15. It was also observed that this takes a longer time to run.

6.3 Experiment 3: YOLOv8

As mentioned earlier, YOLOv8 is implemented from ultralytics code and is trained for 5 to 30 epochs. It gave comparatively good results with previous epochs and took less time to train the model than the YOLOR model. This also proves that YOLOv8 is a lightweight and less complex model. And so this was observed while training the model as it took less time to execute. Looking at good results then trained for 70 epochs but observed the same scenario as YOLOR_P6. Thus after successfully running for 70 epochs, the performance is slightly better when compared to the values of precision but the m@P and recall values are slightly reduced.

6.4 Experiment 4 YOLOv8 trained on pretrained file

It loads the pretrain file which is "YOLOv8n.pt" used for object detection. and then this model under training on the dataset used for 30 epochs. This model has shown surprisingly better results than all models described on top. However the Precision, recall, and m@P values are similar to the YOLOv8 model trained on the dataset used. But this model is faster than all models.

6.5 Experiment 4 YOLOR-YOLOv8

This is the combination of trained models YOLOR and YOLOv8. As YOLOv8 trained on a pretrained file is recognized as a better model, so this model is used in the YOLOR-YOLOv8 model. YOLOR_P6 is selected based on the compatibility of two models in accordance with the base channel size. This combined model is then tested and demonstrated that the values of recall, and m@p are better than all other models individually but the precision value is reduced compared to the YOLOv8 model. Figure 4 shows a few of the images that undergo object detection from the test images. It detects multiple images from a single image with high confidence. For class cardboard, it not only detects the cardboard with khaki color but also the booklet with another color. Talking about plastic bottles, it detects all kinds of bottles with or without labels, even with squeezed plastic bottles. The same scenario is also observed for aluminum cans class.

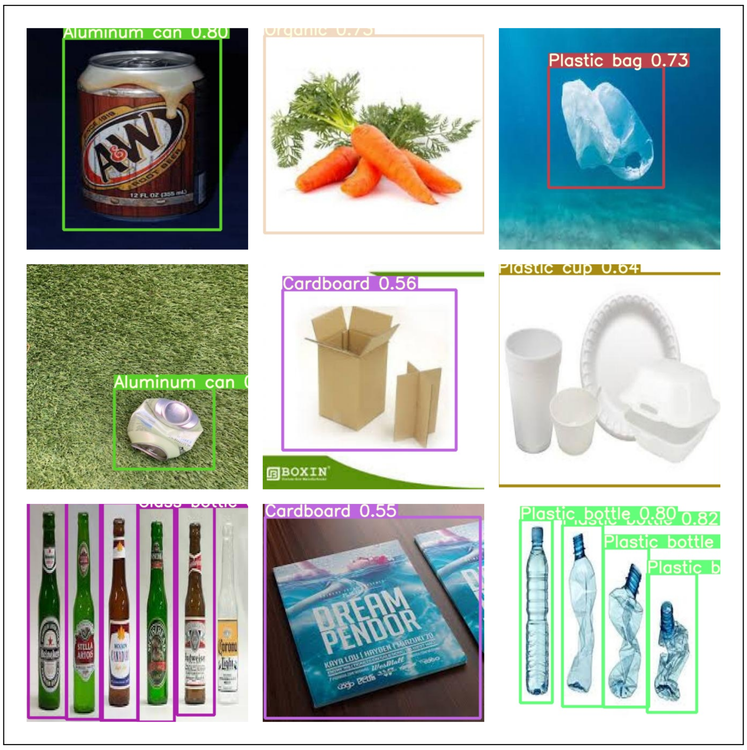


Figure 4: YOLOR-YOLOR testing on images

6.6 Discussion

Table 4 shows the precision, recall, and m@P values of all the models trained at multiple epochs. It is recognized from this table that there are two models that are outperforming which are YOLOv8 and YOLOR-YOLOv8. Although the precision value is relatively low for the YOLOR-YOLOv8 model. However, the m@P values are not affected by a slight increase in recall. Thus it is hard to conclude the best model out of these models. Talking about efficiency the YOLOv8 is a very efficient model compared to other models. But the performance is adequately decreased or is the same after 30 epochs shown in figure 5. while losses look good in the graph as they decrease after every epoch. Apart from this, the YOLOR model has shown good performance over the epochs as seen in figure 6. The YOLOR model is more complex than the YOLOv8 model and thus takes more time to train. But if the YOLOR is trained for more epochs than 70, it is expected to give better results. And as observed for YOLOv8 it is a lightweight model and gives better results at early epochs but loses the performance after a specific epoch for this dataset.

Table 4: Comparison between models

Model	**epochs**	**precision**	**recall**	**maP**
YOLOR_P6	30	0.133	0.308	0.251
YOLOR_P6	70	0.225	0.352	0.345
YOLOR_W6	15	0.121	0.262	0.19
YOLOv8	30	0.518	0.351	0.389
YOLOv8	70	0.587	0.318	0.354
YOLOv8 on .pt file	70	0.587	0.318	0.354
YOLOR-YOLOv8	-	0.247	0.368	0.354

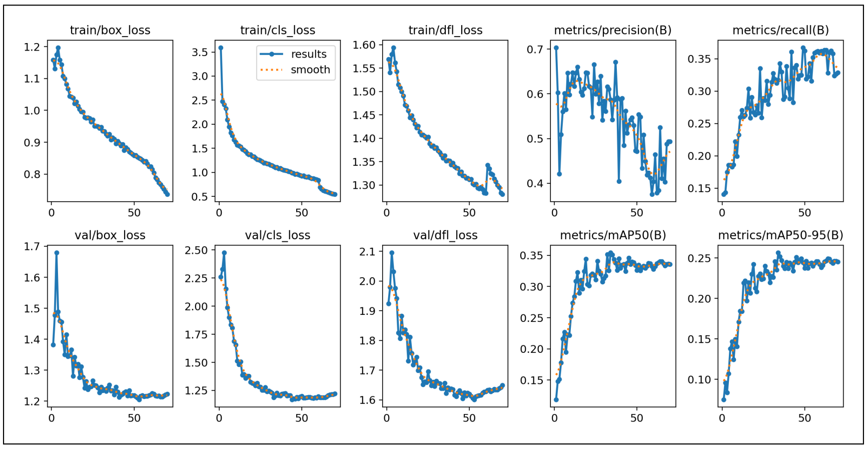


Figure 5: YOLOv8 result graph at 70 epochs

As Figure 7 illustrates, an unbalanced dataset appears to be the cause of the observed drop in the YOLOv8 model's performance. The precision, recall, and mAP values that were found during testing are probably depicted in this figure. After careful examination of the precision, recall, and mAP values for every class in the dataset, it was found that the

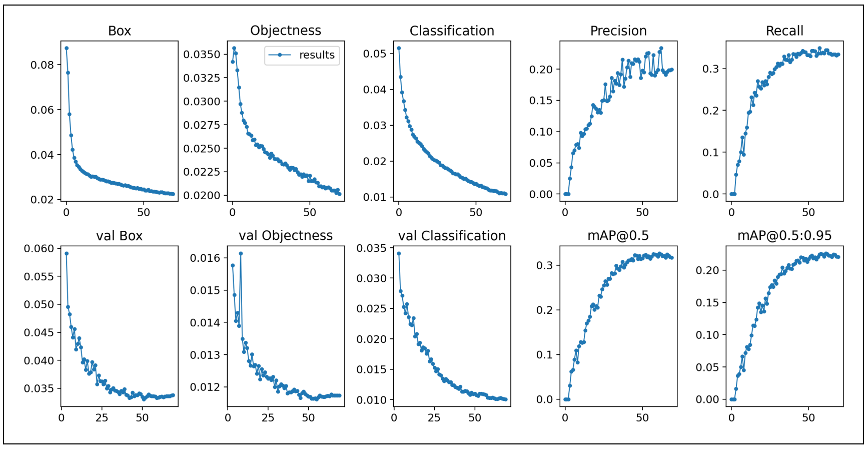
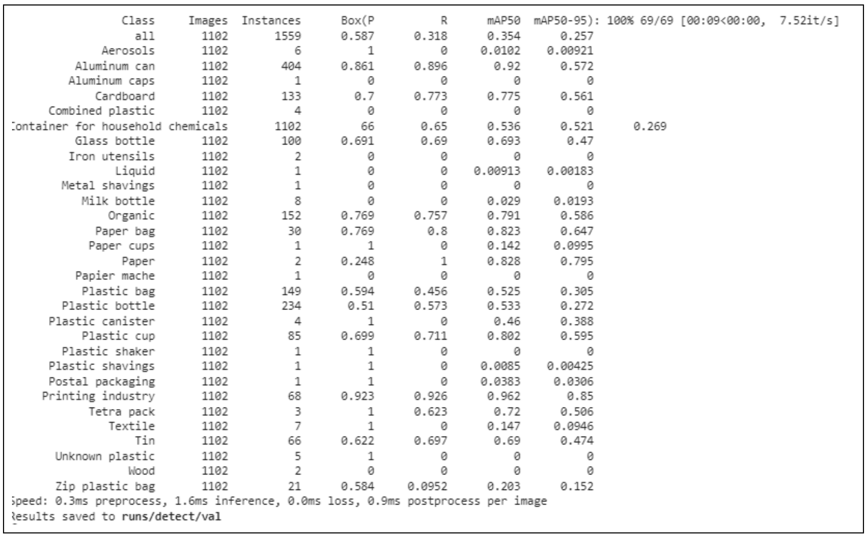


Figure 6: YOLOR_P6 result graph at 70 epochs

majority of the classes performed quite well. The model shows a high degree of confidence and accuracy in classifying unknown photos for these classes. But a major problem arises when classes with a comparatively small number of photos in the training dataset are used. The model's overall performance is negatively impacted by this class imbalance, especially for the underrepresented classes.It is difficult for the model to generalize and categorize objects within these classes accurately because it was not exposed to them enough during training. To potentially improve the model's performance across all classes, this problem could be addressed by using methods like data augmentation, gathering extra data for underrepresented classes, or using data balancing procedures during training.



```
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 69/69 [00:09<00:00,  7.52it/s]
                   all       1102       1559      0.587      0.318      0.354      0.257
              Aerosols       1102          6          1          0     0.0102    0.00921
          Aluminum can       1102        404      0.861      0.896       0.92      0.572
         Aluminum caps       1102          1          0          0          0          0
             Cardboard       1102        133        0.7      0.773      0.775      0.561
      Combined plastic       1102          4          0          0          0          0
:ontainer for household chemicals       1102         66       0.65      0.536      0.521      0.269
          Glass bottle       1102        100      0.691       0.69      0.693       0.47
         Iron utensils       1102          2          0          0          0          0
                Liquid       1102          1          0          0    0.00913    0.00183
        Metal shavings       1102          1          0          0          0          0
           Milk bottle       1102          8          0          0      0.029     0.0193
               Organic       1102        152      0.769      0.757      0.791      0.586
             Paper bag       1102         30      0.769        0.8      0.823      0.647
            Paper cups       1102          1          1          0      0.142     0.0995
                 Paper       1102          2      0.248          1      0.828      0.795
           Papier mache       1102          1          0          0          0          0
           Plastic bag       1102        149      0.594      0.456      0.525      0.305
        Plastic bottle       1102        234       0.51      0.573      0.533      0.272
      Plastic canister       1102          4          1          0       0.46      0.388
           Plastic cup       1102         85      0.699      0.711      0.802      0.595
        Plastic shaker       1102          1          1          0          0          0
      Plastic shavings       1102          1          1          0     0.0085    0.00425
      Postal packaging       1102          1          1          0     0.0383     0.0306
     Printing industry       1102         68      0.923      0.926      0.962       0.85
            Tetra pack       1102          3          1      0.623       0.72      0.506
               Textile       1102          7          1          0      0.147     0.0946
                   Tin       1102         66      0.622      0.697       0.69      0.474
       Unknown plastic       1102          5          1          0          0          0
                  Wood       1102          2          0          0          0          0
       Zip plastic bag       1102         21      0.584     0.0952      0.203      0.152
;peed: 0.3ms preprocess, 1.6ms inference, 0.0ms loss, 0.9ms postprocess per image
Results saved to runs/detect/val
```

Figure 7: Class distribution result for YOLOv8 model

7 Conclusion and Future Work

This research project is dedicated to advancing the capabilities of object detection across a diverse range of items within the "YOLO Waste Detection Image Dataset," which comprises a total of 42 distinct classes. The proposed technique denoted as YOLOR-YOLOv8, introduces a unique approach. It involves training two separate models, namely YOLOR and YOLOv8, individually. These models create multiple weight pretrained file while training and pick the best pretrained file among all. Subsequently, the pretrained files of these models are merged using Ensemble Learning techniques.

The study conducts a comprehensive comparative analysis, evaluating the performance of various models, including YOLOR_P6, YOLOR_W6, YOLOv8 trained on pretrained files, standard YOLOv8, and the ensemble model YOLOR-YOLOv8. Results indicate that both YOLOv8 and YOLOR-YOLOv8 outshine other models in terms of their respective performance metrics. Interestingly, YOLOR_P6 stands out for consistently demonstrating superior m@P values across different epochs, signaling notable improvements in both recall and m@p.

A crucial limitation observed during the project pertains to the imbalanced distribution of classes within the dataset. As already discussed, the YOLOR and YOLOv8 model requires images with their respective annotation files. The first restriction in this field is just a few of the datasets that meet the requirements. It also takes a lot of time to look through thousands of images in the dataset and build the annotation files for all the objects present due to time constraints. Although the selected dataset has the necessary file as per the model generated for this research, the distribution of classes in this dataset is not balanced. There are 42 classes in all, and some include several photos, while others just have one or two, indicating that the class is undersampled. This could lead to erroneous analysis of previously undiscovered features, particularly for underrepresented classes. This could interfere with the model's learning process and lead to incorrect object predictions in classes with underrepresented data since the classes don't cover enough features when applied to real-world objects. Consequently, this has some effect on the model's accuracy. However, the research paper's analysis shows that the class that was mostly covered demonstrated improved object detection.

Looking ahead, the researchers suggest a promising avenue for future work—addressing the dataset's class imbalance. By creating a more balanced dataset, the researchers anticipate an even more substantial performance improvement. The combination of YOLOR and YOLOv8 has shown promise, and with a dataset that is both comprehensive and balanced, along with extended training epochs, the model's capabilities are expected to further excel.

YOLOR, YOLOv8, and the combined model YOLOR-YOLOv8 have effectively recognized the majority of objects under the appropriate classifications. A collection of real-time photos that the author has gathered is included in the dataset. In light of this, the model will be useful for sorting waste in real-time scenarios. Many programs that sort large amounts of trash data quickly and easily without involving humans can benefit from the use of these models. The approach can be applied by organizations such as government collaboration, industries producing large amounts of garbage, private companies recycling waste, and many others, depending on their specific needs. Everybody's life who works with waste will be easier as a result. Integrating the generated models with the robotic system which will be responsible for carrying out the task of sorting. The model can guide robots to identify and segregate different features observed. Thus this segregated waste becomes valuable which can then be recycled to form new things. This can effectively conclude recycling the waste in its appropriate form, as major of waste products are recycled which in return can save the environment. There are many such approaches like installing smart bins that integrate the model to segregate the wastes at that moment, using the model to segregate the loads of waste generated in multiple companies, etc.

In conclusion, the research underscores the crucial role of dataset quality in influencing the effectiveness of object detection models. It recommends future efforts to mitigate class imbalances, offering a pathway for achieving heightened performance in object detection tasks. The study contributes not only to the development of novel model combinations but also emphasizes the significance of dataset curation for successful implementation in real-world scenarios.

8 Acknowledgement

I would like to take this opportunity to thank Prof. Cristina Hava Muntean for being a guide and mentor to me during this endeavor. Please accept my warmest compliments. His unwavering assistance and direction have enabled me to overcome the obstacles that my research has brought about.

References

Andhy Panca Saputra, K. (2021). Waste object detection and classification using deep learning algorithm: Yolov4 and yolov4-tiny, *Turkish Journal of Computer and Mathematics Education (TURCOMAT)* **12**(14): 1666–1677. Cited by 9.

Bobulski, J. and Kubanek, M. (2019). Waste classification system using image processing and convolutional neural networks, *in* I. Rojas, G. Joya and A. Catala (eds), *Advances in Computational Intelligence*, Springer International Publishing, Cham, pp. 350–361. Cited by 73.

Chepa, S., Singh, S., Dutt, H., Sharma, A., Naik, S. and Mahajan, H. (2021). A comprehensive study of distinctive methods of waste segregation and management, *2021 Third International Sustainability and Resilience Conference: Climate Change*, IEEE, pp. 440–444. cited by 3.

Gyawali, D., Regmi, A., Shakya, A., Gautam, A. and Shrestha, S. (2020). Comparative analysis of multiple deep cnn models for waste classification, *arXiv preprint arXiv:2004.02168* . Cited by 27.

Kannangara, M., Dua, R., Ahmadi, L. and Bensebaa, F. (2018). Modeling and prediction of regional municipal solid waste generation and diversion in canada using machine learning approaches, *Waste Management* **74**: 3–15. Cited by 261.
URL: *https://www.sciencedirect.com/science/article/pii/S0956053X17309406*

Kumar, S., Yadav, D., Gupta, H., Verma, O., Ansari, I. and Ahn, C. (2020). A novel yolov3 algorithm-based deep learning approach for waste segregation: Towards smart waste management. electronics 2021, 10, 14. Cited by 7.

Lin, W. (2021). Yolo-green: A real-time classification and object detection model optimized for waste management, *2021 IEEE International Conference on Big Data (Big Data)*, IEEE, pp. 51–57. Cited by 8.

Pandey, A., Khator, B., Agrawal, D., Halim, D. and Kumar, J. S. (2023). Segregation of solid municipal waste using machine learning, *2023 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, IEEE, pp. 1–6.

Patel, D., Patel, F., Patel, S., Patel, N., Shah, D. and Patel, V. (2021). Garbage detection using advanced object detection techniques, *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, pp. 526–531. Cited by 23.

ProjectVerba (2022). Yolo waste detection dataset, `https://universe.roboflow.com/projectverba/yolo-waste-detection`. visited on 2023-12-10.
URL: *https://universe.roboflow.com/projectverba/yolo-waste-detection*

Reis, D., Kupec, J., Hong, J. and Daoudi, A. (2023). Real-time flying object detection with yolov8, *arXiv preprint arXiv:2305.09972* . Cited by 50.

Shetty, T. S. (2022). *Identification and Classification of Industrial Plastic Waste Using Deep Learning Models*, PhD thesis, Dublin, National College of Ireland.

Sreelakshmi, K., Akarsh, S., Vinayakumar, R. and Soman, K. (2019). Capsule neural networks and visualization for segregation of plastic and non-plastic wastes, *2019 5th International Conference on Advanced Computing Communication Systems (ICACCS)*, pp. 631–636. Cited by 41.

Thanawala, D., Sarin, A. and Verma, P. (2020). An approach to waste segregation and management using convolutional neural networks, *Advances in Computing and Data Sciences: 4th International Conference, ICACDS 2020, Valletta, Malta, April 24–25, 2020, Revised Selected Papers 4*, Springer, pp. 139–150. Citet by 13.

Tran, V. T., To, T. S., Nguyen, T.-N. and Tran, T. D. (2022). Safety helmet detection at construction sites using yolov5 and yolor, *International Conference on Intelligence of Things*, Springer, pp. 339–347. Cited by 2.

Wang, C.-Y., Yeh, I.-H. and Liao, H.-Y. M. (2021). You only learn one representation: Unified network for multiple tasks, *arXiv preprint arXiv:2105.04206* . Cited by 427.

Wang, G., Chen, Y., An, P., Hong, H., Hu, J. and Huang, T. (2023). Uav-yolov8: A small-object-detection model based on improved yolov8 for uav aerial photography scenarios, *Sensors* **23**(16): 7190. cited by 6.

Yan, T., Sun, W. and Cui, K. (2022). Real-time ship object detection with yolor, *Proceedings of the 2022 5th International Conference on Signal Processing and Machine Learning*, SPML '22, Association for Computing Machinery, New York, NY, USA, p. 203–210.
URL: *https://doi.org/10.1145/3556384.3556415*

Zhang, Y., Song, X., Bai, B., Xing, T., Liu, C., Gao, X., Wang, Z., Wen, Y., Liao, H., Zhang, G. et al. (2021). 2nd place solution for waymo open dataset challenge–real-time 2d object detection, *arXiv preprint arXiv:2106.08713* . Cited by 12.