NU LAGUNA

PAPERDOYLO: A YOLOv8 - BASED SYSTEM FOR IDENTIFYING RECYCLABLE PAPER WASTE

A Thesis Presented to the

Faculty of School of Computer Studies

NU Laguna

In Partial Fulfillment of the Requirements

for the Degree of Bachelor of Science in Computer Science

Submitted by

Gomez, Christian Sydney Earl R.

De Guzman, Serge Trever B.

March 2025

NOT FOR CIRCULATION

APPROVAL SHEET

This thesis project titled PAPERDOYLO: A YOLOv8 - BASED SYSTEM FOR IDENTIFYING RECYCLABLE PAPER WASTE has been prepared and submitted by GOMEZ, CHRISTIAN SYDNEY EARL R., DE GUZMAN, SERGE TREVER B. in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science.

ROSE ANN C. MALABORBOR, MSCS
Research Adviser

Approval in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science. by the Oral Examination Committee.

VINCENT S. RIVERA
Panel Chair

JOSEPH L. DOMINGO, MIT
Panel Member

EDISON M. ESBERTO, MIT
Panel Member

Accepted in partial fulfillment of the requirements for the degree of BACHELOR OF SCIENCE IN COMPUTER SCIENCE.

____
(Date)

JOSEPH L. DOMINGO, MIT
Dean, School of Computer Studies

NOT FOR CIRCULATION

ABSTRACT

The improper classification of recyclable and non-recyclable paper waste remains a significant challenge in waste management and recycling. Under Philippine recycling guidelines, the study presents PAPERDOYLO, a YOLOv8-based model to identify and categorize recyclable and non-recyclable paper trash. The model detects the presence of pollutants like tape, glue, and food residues that impede recycling operations using a dataset of 5,298 images covering eight classes of paper types (recyclable and non-recyclable) and 899 background images. Self-captured photos, AI-generated samples, and web scraping were used to gather the data. The data was ready for training by preprocessing procedures, including augmentation, scaling, and annotation. Metrics, including mean Average Precision (mAP), precision, and recall, were used to assess the YOLOv8 model. The findings demonstrate the system's efficacy in real-time paper waste categorization, which shows a remarkable mAP of 96.2% with accuracy and recall values up to 95%. The custom dataset trained a YOLOv8 model and achieved 92.4% precision, 90.2% recall, and mAP accuracy of 92.1%. The model can detect and classify recyclable and non-recyclable papers with these results. This study creates a scalable framework for automated garbage classification and demonstrates the usefulness of YOLOv8 in recycling programs. The results substantially contribute to waste management procedures, supporting computer vision research.

Keywords: YOLOv8, Waste Management, Recyclable Paper, non-recyclable paper, Object Detection, Sustainability

NOT FOR CIRCULATION

DEDICATION

This research is humbly dedicated to God, whose grace and guidance have been our source of inspiration and strength throughout this journey. His presence has given us the courage and determination to persevere in times of difficulty.

We dedicate this work with heartfelt gratitude to our families, who have stood by us with unwavering love, patience, and encouragement. Their belief in our abilities has been a constant source of motivation.

To our loved ones for their steadfast commitment, shared vision, and collaborative spirit, which have been instrumental in the success of this endeavor. The journey would not have been the same without their dedication and teamwork.

Finally, I would like to thank each member of the research group. Christian Gomez and Serge De Guzman, for showing perseverance, determination, and resilience in overcoming challenges and pursuing this project to completion. This work stands as a testament to our efforts and growth.

NOT FOR CIRCULATION

ACKNOWLEDGEMENT

First and foremost, we extend our deepest gratitude to God for His wisdom, guidance, and blessings, which have been our foundation throughout this research journey. His light has illuminated our path and sustained us in moments of doubt.

We express our heartfelt appreciation to Ms. Rose Ann C. Malaborbor, our research advisor, whose expertise, mentorship, and constructive feedback have been invaluable to the development and completion of this study. Their guidance has shaped the quality and depth of our work.

Our sincerest thanks go to Mr. Vincent S. Rivera, our Research Professor, whose encouragement and insights have significantly influenced the direction of our study. Their support has been a source of inspiration and confidence.

Lastly, we are deeply grateful to Mr. Joseph L. Domingo, the Dean, for their unwavering support, encouragement, and provision of resources that have been instrumental in realizing this research. Their leadership and trust have made this achievement possible.

NOT FOR CIRCULATION

TABLE OF CONTENTS

NOT FOR CIRCULATION

NOT FOR CIRCULATION

LIST OF TABLES

NOT FOR CIRCULATION

LIST OF FIGURES

NOT FOR CIRCULATION

Chapter 1

INTRODUCTION

Background of the Study

Recyclable wastepaper refers to any paper product that can be reused or repurposed through recycling processes to create new paper products. This includes newspapers, magazines, cardboard, office paper, and packaging materials made from paper. The National Solid Waste Management Commission (2018) also released a guideline for identifying the quality standard for acceptable wastepaper. The Integration of Solid Waste Management Tools in Specific European and Asian Communities (ISTEAC) releases paper recycling schemes in the Philippines that review, assess, study, and propose solutions for paper recycling in the Philippines. In their study, they assess and show the quantity of annual paper mill capacity, which shows that different types of paper, industrial grade newsprint, printing and writing, assorted, and cardboard, are the paper produced annually. In this sector, ISTEAC's research also shows that the main types of paper used daily in the Philippines are office white paper, newspapers, and cardboard. ISTEAC also shows the process of how Eco-aides employs the collection of recyclable paper, which indicates that junk shops accept and segregate assorted paper, newspaper, cardboard, and white paper and sell it to dealers, where dealers accept and segregate it again for pulping and milling the recyclable paper. Dealers segregate the wastepaper from white paper (with print), Old Newspaper, Mixed/Assorted paper, and cardboard.

NOT FOR CIRCULATION

Detecting and classifying recyclable and non-recyclable paper offers numerous benefits for environmental sustainability and efficient waste management. By distinguishing between the two, it ensures that recyclable paper products are correctly processed and repurposed, reducing demand for virgin materials and conserving natural resources. That is why the researchers focused on building this study, which addresses the problem of detecting and classifying recyclable and non-recyclable paper using YOLOv8, which is focused on detecting if the recyclable paper has the presence of contaminants that can be classified as non-recyclable paper. Related studies about classifying solid waste detect and classify different solid waste, but not the contaminants that recyclable paper can have.

The research of Pan et al. (2024) utilizes the YOLOv8 for a classification detection algorithm. Their study focuses on detecting and classifying recyclable garbage waste, food waste, toxic and hazardous waste, and other garbage. Their study thoroughly examines the existing YOLOv8 object identification technique to address the tiny target identification issue. Their study focuses on the design and optimization of YOLOv8, and several creative approaches are suggested. Their study focuses on reviewing the fundamental chapters, which provide a detailed analysis of the use of convolutional neural networks in object detection, with particular attention to the YOLO8 algorithm, a significant advancement in the field that establishes a strong basis for further study. The study of Shrestha, Shiva & Shrestha, Samman & Paudel, Prateek & Gurung, Sudarshan & Gaire, Sulav & Adhikari, Smita. (2024) presents a novel deep learning-based waste classification system using the state-of-the-art object detection framework You Only Look Once v8 (YOLOv8) to address urgent environmental concerns. Their study highlighted that waste categorization is

NOT FOR CIRCULATION

essential to efficient waste management because it allows different waste kinds to be separated for proper disposal, recycling, or composting. Their method uses YOLO v8's capabilities to identify and group waste materials into four classes: biodegradable, paper, plastic, and metal. Utilizing the improved accuracy and real-time processing of YOLO v8, their study offers a workable automated garbage-sorting solution. The study of Rahim, Leena & Abidin, Nor & Aminuddin, Raihah & Abu Samah, Khyrina Airin Fariza & Ibrahim, Asma & Yusoh, Syarifah & Mohd Nasir, Siti. (2024) focused on building a model that identified and detected different solid waste products like plastic, paper, etc., using YOLOv5 but did not classify each type if they had a presence of contaminants for each type of solid waste. The study of Altıkat, Aysun & Gülbe, Alper & Altikat, Sefa. (2021) also identifies and detects different types of solid waste, which are paper, glass, plastic, and organic waste, using deep convolutional neural network algorithms. Their study focused on identifying and detecting different types of solid waste, which we came up with using the presence of contaminants that recyclable paper can have. That is why the researchers of this study aim to create and test a computer vision model using the YOLOv8 algorithm to detect and classify recyclable and non-recyclable paper, which is focused on detecting if the recyclable paper has the presence of contaminants that can be classified as non-recyclable paper.

The researchers focused their study on classifying recyclable paper by assessing the presence of contaminants using the insoluble contaminants according to the study of Tandon Rita, Bhardwaj Nishi K., Mathur R.M., and Kulkarni A.G. (n.d.) as their benchmark. This approach distinguishes its research from others that primarily detect and classify solid waste without reference to specific standards. By aligning the considered

NOT FOR CIRCULATION

recyclable paper with NSWMC (2016) and ISTEAC with the study of Tandon Rita, Bhardwaj Nishi K., Mathur R.M., and Kulkarni A.G (n.d.), which tells those insoluble contaminants need to be closely monitored and helps to differentiate and classify the non-recyclable paper, the study aims to provide a method for identifying recyclable paper, enhancing the reliability and comparability of their findings within the field of recycling and waste management research. This emphasis on standardized criteria underscores the study's contribution to advancing practices and understanding in recycling efforts.

Research Objectives

The primary objective of this research is to create and test a computer vision model utilizing the YOLOv8 algorithm to detect and classify recyclable and non-recyclable paper trash.

Specific objectives:

1. To gather, preprocess, and create a data set of recyclable and non-recyclable paper images.
2. Implementation of object detection and classification using the YOLOv8 algorithm.
3. To evaluate the YOLOv8 model using the prepared dataset and validate its performance through testing, employing metrics like recall, precision, and mean Average Precision (mAP).
4. To evaluate the real-time performance of the YOLOv8 model based on the testing procedures.

NOT FOR CIRCULATION

Theoretical Framework

Computer Vision Theory

Computer vision is a field in artificial intelligence that focuses on enabling machines to replicate the human ability to understand visual information from the world through digital images and videos (Szeliski, 2011). Computer vision aims to build models that could perform tasks like human vision and have the possibility even to surpass that. It includes image acquisition, preprocessing, feature extraction, object detection, and classification.

According to the study of Chen, Wang, Lin, Cui, and Zong (2024) regarding the processes in computer vision. Image acquisition uses cameras or other devices to convert real-life images into digital form for computer processing. This is followed by image preprocessing, which processes the acquired images through different techniques such as resizing, annotating, enhancing, and normalization to prepare them to pass through the CNN for feature extraction. Next is feature extraction, the core of computer vision that converts complex image information into feature maps.

Object detection is the localization of specific objects in an image or video. This became a challenge due to the variations in object sizes, colors, and orientations. To address this problem, feature-based deep learning techniques have been developed. Convolutional Neural Networks (CNNs), in particular, are a type of deep learning technique suitable for object detection tasks because of their ability to learn features on large scales (Shelke et al., 2023). Integrating deep learning into computer vision by its ability to automatically

NOT FOR CIRCULATION

learn and optimize features from datasets has been an advancement in object detection and classification (LeCun et al., 2015).

These principles are applied throughout the study by incorporating the state-of-the-art object detection architecture, YOLOv8, for its ability to detect and classify with high speed and accuracy, making it suitable for real-time object detection (Ultralytics, 2023a). In this study, YOLOv8 is a deep learning architecture to develop a real-time object detection model that classifies paper as recyclable or non-recyclable.

Convolutional Neural Network (YOLOv8)

The foundation of this research is theories and concepts of object detection and classification of the YOLOv8 algorithm. The YOLOv8 algorithm was implemented in this study for object detection and to classify whether the paper is recyclable, depending on the parameters this study used. According to Huang, Pedoeem, and Chen (2018), the simultaneous completion of the bounding box and class prediction sets YOLO apart from other conventional systems.

In the study of Zhang & Hong (2019), YOLO employs a convolutional neural network and can be classified as a one-stage detector since it predicts using a single feed- forward convolutional network or a reduced convolutional network with extra multi-scale layers, object classes, and bounding boxes may be determined quickly. After the features are extracted, they go to feature combination, as seen in Fig. 1. Then, they are separated into an S x S grid, and the grid where the item's center lies is in charge of the object's prediction.

NOT FOR CIRCULATION

As stated by Huang, Wang, Xiao, and Zhu (2024), the convolutional neural network is responsible for processing the feature map after feature extraction from the basic level and converting feature maps into detection bounding boxes.

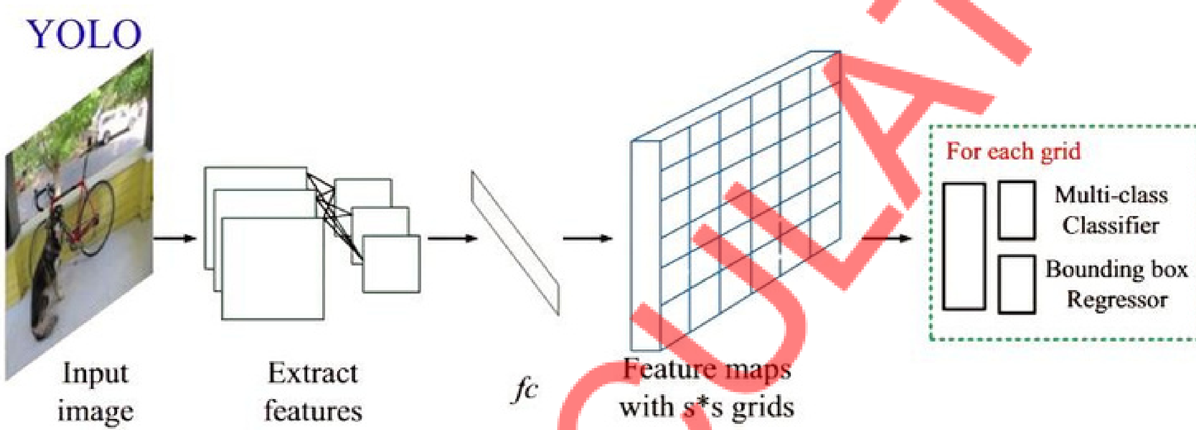


Figure 1. Detection Pipeline of YOLO (Zhang & Hong, 2019)

According to the study of Ranjbarzadeh, Ramin & Crane, Martin & Bendechache, Malika. (2025), the backbone of object detection and classification models is primarily responsible for feature extraction. It utilizes convolutional layers to capture the input images' hierarchical spatial and semantic features. The YOLOv8 uses a modified convolutional neural network (CNN) and the backbone, CSPDarknet53. The neck component further enhances these extracted features using structures like PAN and FPN, which contribute to improved object localization accuracy. Finally, the head performs the bounding box regression and classification tasks, resulting in precise localization predictions and confidence scores. The YOLOv8 model architecture, as highlighted in the study of Ranjbarzadeh, Ramin & Crane, Martin & Bendechache, and Malika. (2025) showcases significant advancements compared to its predecessors. While retaining the core YOLO principle of combining feature extraction, bounding box prediction, and class probability assignment into a single end-to-end framework, YOLOv8 introduces various enhancements to improve performance and precision. The detailed structure of the YOLOv8 model is illustrated in Figure—1 of the referenced works. The backbone of YOLOv8 is designed to identify initial features from input images, with its architecture differing from earlier versions by offering multiple backbone options ranging from extra small (XS) to large (L). These options enable the model to balance computational efficiency and feature detection capabilities optimally. This flexibility allows YOLOv8 to adapt to diverse computational environments, from edge devices with limited processing power to high-performance computing systems. These advancements are comprehensively discussed in prior research.

In the study of Sung, Tien-Wen & Li, Jie & Lee, Chao-Yang & Fang, Qingjun. (2025), the YOLOv8 model's neck section combines features taken from different layers of the backbone section and upsample feature maps. It has a concat block that combines the output channels of concatenated blocks while preserving resolution and an upsample layer that doubles the feature map's size without changing the output channels. The final output of the model is produced by the head portion, which also creates predictions for bounding boxes and object classes. Additionally, using inputs from the C2f block in Figure 2, the head section's initial detect block is tuned for tiny item detection.
Studies highlight that the "neck" of object detection models employs advanced feature aggregation algorithms to merge information across multiple backbone layers. This fusion significantly enhances the model's ability to detect objects at varying scales, an essential characteristic for identifying small and large objects within the same image. (Ranjbarzadeh, Ramin & Crane, Martin & Bendechache, Malika. 2025).

NOT FOR CIRCULATION

The "head" of the YOLOv8 model is specifically designed for predicting bounding boxes and class probabilities. Research indicates that the prediction mechanism in YOLOv8 includes refined loss functions, which address the complexities of object localization and classification more effectively. These enhancements improve precision detection and reduce false positives (Ranjbarzadeh, Ramin & Crane, Martin, and Bendechache, Malika, 2025).

Furthermore, YOLOv8 integrates advanced training methodologies. Notably, transfer learning through pretraining on the COCO dataset has been optimized. This approach enhances the model's ability to generalize object detection capabilities, serving as a strong foundation for fine-tuning specific datasets. Studies show that pretraining accelerates the training process and reduces computational resource requirements while achieving high accuracy (Ranjbarzadeh, Ramin & Crane, Martin, and Bendechache, Malika, 2025).

NOT FOR CIRCULATION

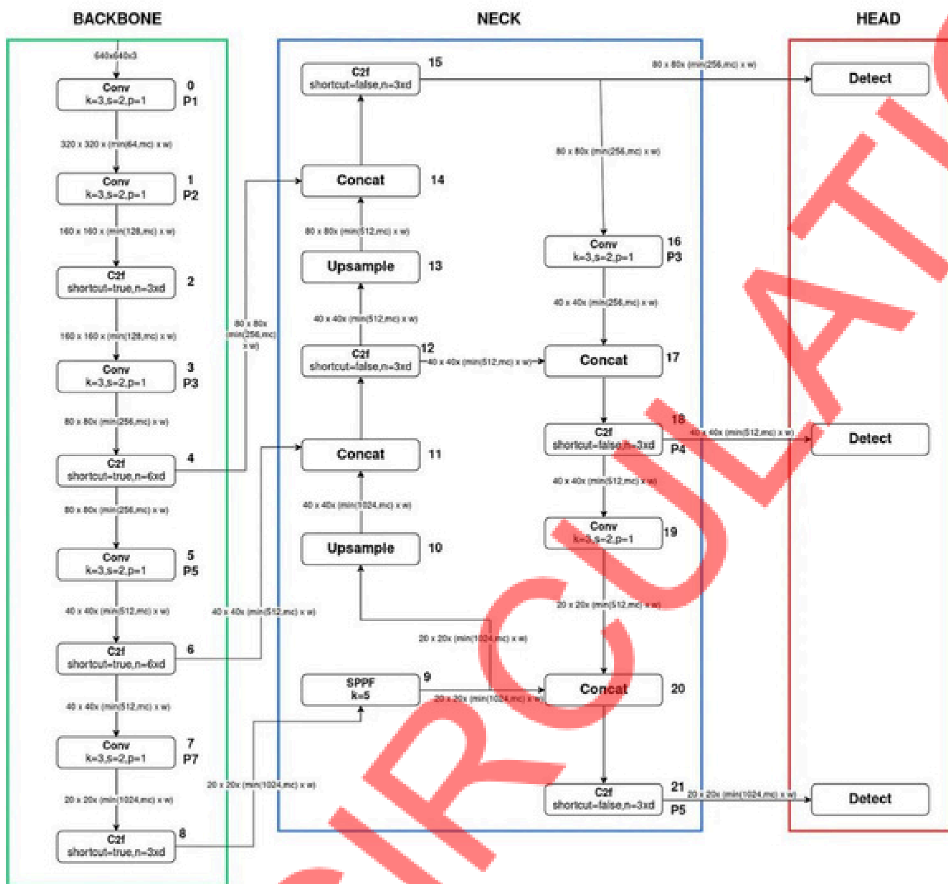


Figure 2. YOLOv8 Architecture from Abin Timilsina (2024)

Convolutional Block

Each stage of YOLO's architecture consists of multiple convolutional blocks, known as fundamental building blocks or structures. Each step produces features and processes data from a different resolution (Hidayatullah et al., 2025). YOLOv8 contains a 2-dimensional convolutional layer, a 2-dimensional batch normalization, and a SILU activation function. The model begins with a convolutional layer that transforms the input from 3 feature maps to 16, using a kernel size (K) of 3x3, a stride (S) of 2, and padding (P) of 1 (Wang, 2023). Each convolutional block has a set of learnable weights updated every

NOT FOR CIRCULATION

epoch during training. These updates allow the model to learn and improve its performance continuously.

The Conv2d module is a convolutional module specialized in processing two-dimensional data (Zhou et al., 2024). It operates on two dimensions: height and width. According to the analysis by Abin Timilsina (2024) on the YOLOv8 architecture, *k* represents the number of kernels, *s* is the stride, which determines how the kernel slides over the input, *p* is the padding, referring to the additional border of zeroes added to each side. C is the number of channels in the input.

According to the study by Chen et al. (2019), Batch Normalization improves training stability and enhances convergence speed. They also stated that batch normalization stabilizes the training by controlling the distributions of the layer outputs and is especially helpful for training intense networks with many layers. Normalization of activations resulting from convolution. This involves calculating averages and standard deviations across the batch to stabilize the distribution of activations (Pedro, 2023). For instance, Cai et al. (2019) provided a quantitative analysis showing that BN allows for larger learning rates and improves convergence speed in gradient descent methods. That is why batch normalization was used in the convolutional block in YOLOv8.

Ultralytics (2025) also said that SiLU is an activation function used in deep learning models, just like YOLOv8; it was valued for its smoothness and non-monotonic properties, which can help in gradient flow and model optimization. The formula is SiLU(x) = x * sigmoid(x), which acts as a self-gating mechanism where the sigmoid component determines the extent to which the linear input x is passed through. If the sigmoid output is close to 1, the input passes through unchanged, and if the output is close to 0, it is

NOT FOR CIRCULATION

suppressed towards zero. Additionally, Glenn Jocher (2024a), who is a developer in Ultralytics, answered one of the forums that they chose to use the SiLU activation function for YOLOv8 because it showed to improve the performance and helps the model learn more complex patterns due to its non-linearity, while also being smooth and continuous, which is beneficial for gradient-based optimization.

In deep neural networks, a vanishing gradient problem occurs when the gradients become very small, making it difficult for the model to learn effectively. Unlike other functions, SiLU maintains a non-zero gradient even for large input values, which helps to alleviate the vanishing gradient problem (Shamsan, 2024). It allows for more effective training of deep neural networks. SiLU combines the properties of the sigmoid and linear functions, leading to YOLO learning more complex patterns and improving object detection accuracy (Shamsan, 2024).

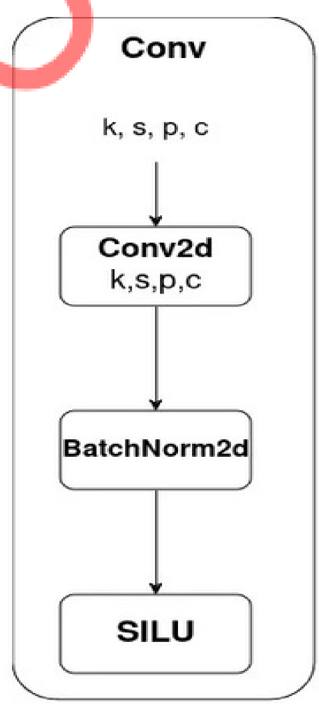


Figure 3. Convolutional Block from Abin Timilsina (2024)

NOT FOR CIRCULATION

Bottleneck Block

The bottleneck block consists of two convolutional blocks with a shortcut connection. A shortcut connection, also known as a skip connection, bypasses one or more layers in the network, as shown in Fig. 4, and passes through the two convolutional blocks. A shortcut connection enhances the model's training to learn complex features and prevents the vanishing gradient problem (Timilsina, 2024). If the shortcut is valid, the input goes through two parallel paths; one goes through the convolutional blocks, while the other is preserved as-is, then the two paths are merged, which preserves the original feature with the additional extracted feature (He et al., 2016). When the shortcut is false, the input goes through the convolutional blocks, and only the extracted features are passed forward without adding their original feature. Why bottleneck block has two 3x3 convolutional blocks was answered by the creator of YOLOv8, Glenn Jocher (2024b), in one of the forums, that while using 1x1 convolutions indeed reduces the parameter and computational cost, using two 3x3 convolutions offers a better deal with maintaining a high detection accuracy. The amount of bottleneck block used differs by the number of depth_multiple parameters of the model (Timilsina, 2024).

NOT FOR CIRCULATION

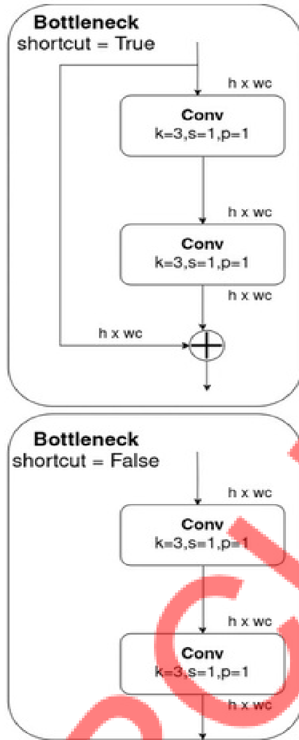


Figure 4. Bottleneck Block from Abin Timilsina (2024)

C2F (Cross-Stage Partial with Fusion) Module

The study of Xiao, Bingjie & Nguyen, Minh, and Yan, Weiqi. (2023) revealed that the C2f modules of the YOLOv8 model significantly increased classification results, and their model reached an impressive accuracy rate of 99.5%.

YOLOv8 uses an optimized version of the C2 module, which is a C2f module that focuses on improving execution speed without sacrificing performance and features, passing through connections to enhance information flow while keeping the structure lightweight (Shroff et al., 2023). The C2F block, as shown in Figure 5, is an essential part of the YOLOv8 architecture, designed to enhance the model's ability to learn features at different scales while remaining computationally efficient. The process begins with the input feature map passing through a convolutional layer, reducing the number of channels.

NOT FOR CIRCULATION

This transformed feature map is then split into two parts. One part is processed through a series of bottleneck layers designed to extract compact and hierarchical features, while the other remains unprocessed. After the bottleneck layers have processed their part of the feature map, the outputs are concatenated with the unprocessed feature map. This combination allows the block to integrate the original and newly learned features, preserving critical information. Finally, the concatenated feature map undergoes another convolutional layer, consolidating the combined information into the final output. This design allows the C2f block to efficiently merge low-level and high-level features, enabling the model to capture fine-grained details from the input data without significantly increasing computational complexity. Fusing processed and unprocessed features ensures the model retains and enhances essential details for better performance. (Xiao, Bingjie & Nguyen, Minh, and Yan, Weiqi. 2023).

NOT FOR CIRCULATION

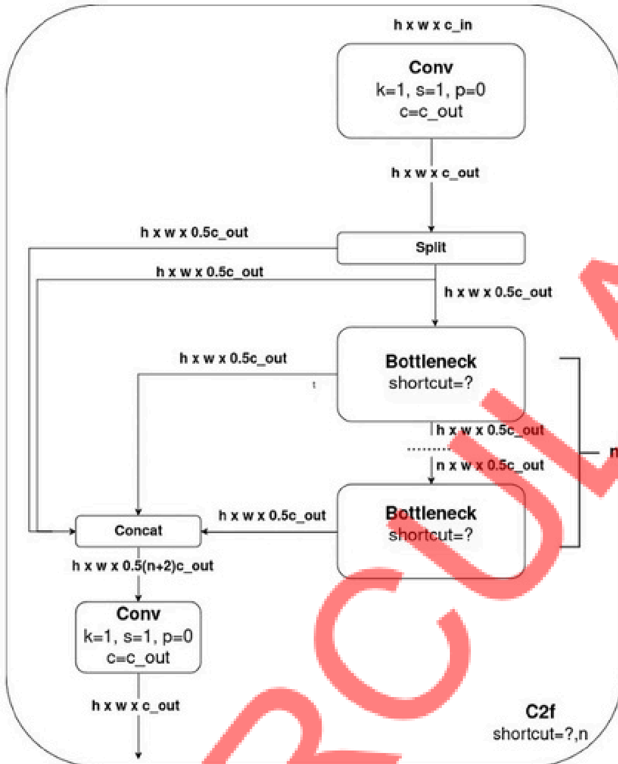


Figure 5. C2F Block from Abin Timilsina (2024)

Spatial Pyramid Pooling Fast (SPPF) Block:

The SPPF Block consists of a convolutional block followed by three MaxPool2d blocks, concatenated, and another convolutional block at the end. Spatial Pyramid Pooling – Fast, or SPPF, represents the feature maps across multiple scales, which allows the model to capture features at different levels of abstraction (Hidayatullah et al., 2025). SPPF was first introduced in YOLOv5, and it has been implemented in YOLOv8 as well. In YOLOv8 (introduced initially in YOLOv5 v6.0), the traditional Spatial Pyramid Pooling (SPP) block was replaced by a more efficient variant called Spatial Pyramid Pooling - Fast (SPPF). Instead of applying three parallel MaxPool2d operations with kernel sizes of 5, 9, and 13, SPPF uses three sequential MaxPool2d layers, each with a fixed kernel size of 5, which doubles the speed of the process while maintaining the same output (Ultralytics, 2023b).

NOT FOR CIRCULATION

SPPF extracts dominant features by looking at the feature maps in different-sized areas, which helps the model detect objects of different sizes.

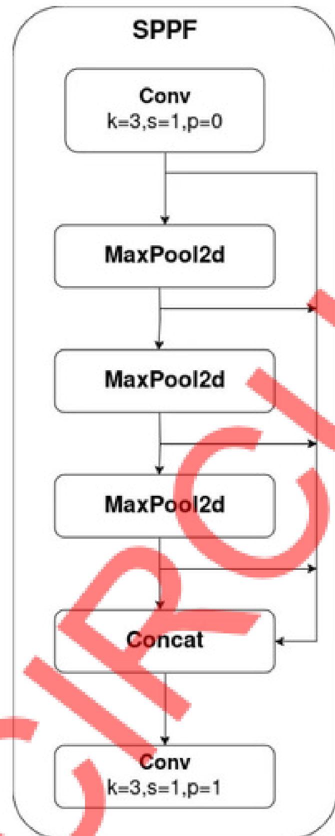


Figure 6. Spatial Pyramid Pooling Fast (SPPF) Block from Abin Timilsina (2024)

Detect Block

A detection block consists of two convolutional blocks and a Conv2d layer for the Bounding Box and Class Loss, respectively. The detection block contains two paths; the top path is for bounding box prediction, and the bottom is for class predictions. YOLOv8 is an anchor-free model that does not have predefined anchor boxes because it predicts the center of an object within the grid cell rather than an offset (Terven et al., 2023). The "reg_max" parameter defines the maximum range of height and width parameters used in prediction; even if YOLOv8 is anchor-free, it still needs to be constrained to prevent

NOT FOR CIRCULATION

degenerate predictions (Jocher, 2023a). The default value is 16, the balance between small and large bounding boxes.

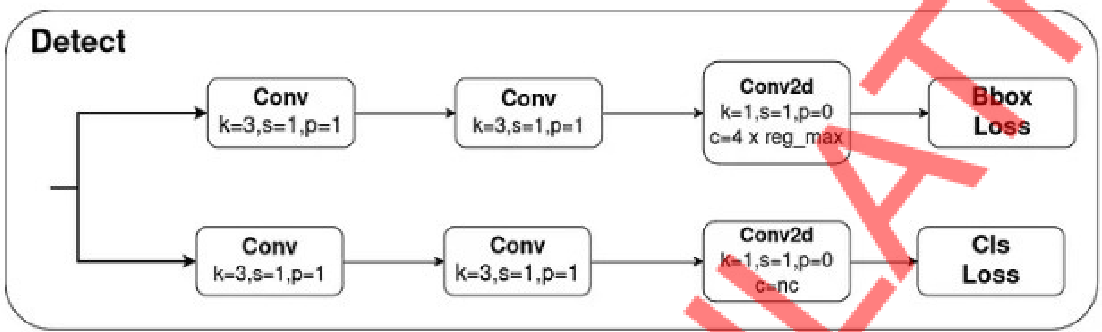


Figure 7. Detect Block from Abin Timilsina (2024)

Post-Processing

Subsequently, YOLOv8 utilizes the Non-Maximum Suppression (NMS) algorithm to address the issue of overlapping predictions. For example, the NMS algorithm successfully reduces bounding box overlap in situations where more than one paper is found in the image, ensuring that each object is distinctly and accurately localized. YOLO's concept is to use NMS (non-maximum suppression) to frame and classify objects once the Algorithm has detected them to choose the proper prediction bounding boxes (Gong et al., 2020).

Additionally, according to the study of Aishwarya Singh (2024), the overlap, or IoU, of the bounding boxes and the objectiveness score provided by the model are the two factors that the non-maximum suppression (NMS) considers. After removing all other bounding boxes with a high overlap, the NMS choose the bounding box with the highest objectiveness score. To guarantee that each object is recognized only once and that predictions of higher quality are kept, NMS is used to eliminate those with lower confidence scores and keep just the best ones (Jocher, 2023b).

NOT FOR CIRCULATION

As stated by the creators of YOLO, Redmon, Divvala, Girshick, and Farhadi (2016), the input image is divided into an S x S grid using the YOLO algorithm, and each grid cell is in charge of detecting an object if the object's center falls inside it. The same statement by an educational organization called GeeksforGeeks (2022) that the YOLO architecture splits the image into an S x S grid, and the grid is in charge of identifying the item if the object's bounding box center is within the grid. According to the analytics of Prashant Malge (2024), YOLOv8 divides the input image into grid cells. This strategy allows the model to predict bounding boxes and each cell's associated class probabilities.

Multi-class classifiers may categorize data into more than two classes and are generally based on user-defined labels (Srivastava et al., 2021). YOLO can classify more than one object in an image. According to Zoumana Keita (2022), the next step in determining the bounding boxes corresponding to rectangles highlighting every object in the image is called bounding box regression. YOLO uses a single regression module to determine these bounding boxes' properties based on the vector Pc, bx, by, bw, bh, C1, C2, C3, ...

According to Roy, Pronoy & Islam, Md. Riadul, and Rafi, Alam. (2024), the YOLOv8 model divides an image into grids and pulls features out of each grid cell. A bounding box and a confidence score are predicted for each cell with an object in the center. They said the bounding box includes five predictions (x, y, w, h) and a confidence score, where w and h represent the image's width and height, and x and y represent the bounding box's center coordinates. Also, the creators of YOLO (Redmon et al., 2016) and (GeeksforGeeks, 2022) stated that each bounding box consists of five predictions (x,y,w,h) and a confidence score. The confidence score is also the probability score.

NOT FOR CIRCULATION

Additionally, in the study of Mukherjee (2023), the image is fed into the convolutional neural network (CNN) by the YOLO algorithm, and the neural network produces a vector as its output. The attributes of the vector are pc, bx, by, bw, bh, C1, C2, C3... Pc, or the class probability, indicates whether an object is present. The bounding box's coordinates from the object's center point are specified by the variables bx, by, bw, and bh, and c1, c2, c3... represent the several classes seen in the picture. If the cell contains no objects, the vector will be 0.,? ...? The remainder of the vector's elements will not matter because the PC will be 0. YOLO will generate many of these bounding boxes for each possibility in the image if there are several objects, each with its bounding box. This indicates that a single image will contain S x S x n, where n is the object's vector size and S x S is the number of grid cells. A similar study by Lavanya and Pande (2023) used the same procedure on how the CNN of YOLO uses a vector for detecting and classifying objects by their attributes of pc, bx, by, bw, bh, c1, c2, c3...

Furthermore, the study of Asmaa Mirkhan (2023) shows that the structure of YOLO is that the input image is passed through a convolutional neural network (CNN) to extract features from the image, and the features are sent through a series of fully connected layers, which predict bounding box coordinates and class probabilities. Then, a grid of cells is created from the image, and the task assigned to each cell is to predict a set of bounding boxes and class probabilities. A set of bounding boxes and probabilities is what the network outputs. After that, the bounding boxes are filtered using the non-max suppression post-processing algorithm to eliminate any overlapping boxes and select the box with the highest probability. The final output is a set of predicted class labels and bounding boxes for each object in the picture. In the study of Liu, Anguelov, Erhan, Szegedy, Reed, Fu, and Berg

NOT FOR CIRCULATION

(2016), on which Mirkhan based her article, and the Figure of the structure of YOLO, Given that they use the very first version of YOLO, they test it on various datasets and come up with a result of 63.4% mAP with 45 frames per second (FPS), which shows that YOLO can detect and classify an object.

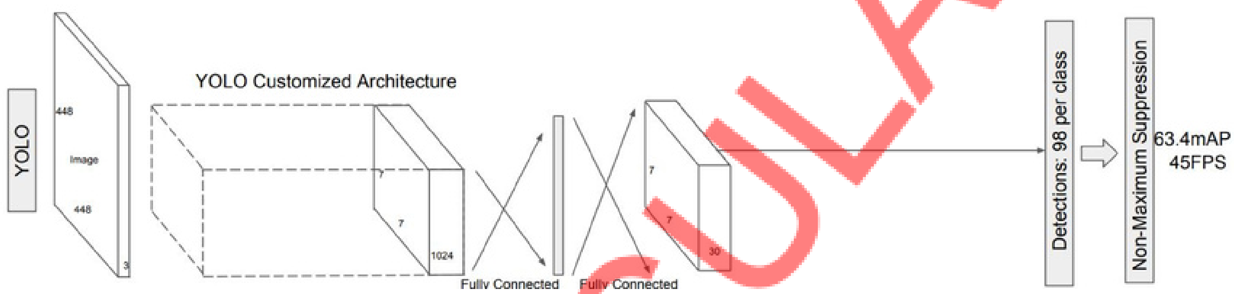


Figure 8. Structure of YOLO (Mirkhan, A., 2023)

According to Figure 9, the flow is how YOLO detects and classifies an image. First, the input image is automatically preprocessed by YOLO into a 640 x 640 resolution, as it was the default resolution. Next is the feature extraction part, which consists of multiple convolutional layers to extract features in the image with a kernel passing through each pixel to construct feature maps and goes to pooling layers that are used to downsample the spatial dimensions of the input volume, reducing the computational complexity of the network and extracting dominant features (Timilsina, 2024). Next is grid creation, bounding box, and classification prediction that divides the image into grids with bounding box and class predictions for each grid, and after that is the non-max suppression, which removes overlapping boxes and retains close-to-ground truth boxes (Arvio, Kusuma, and Sangadji, 2024).

NOT FOR CIRCULATION

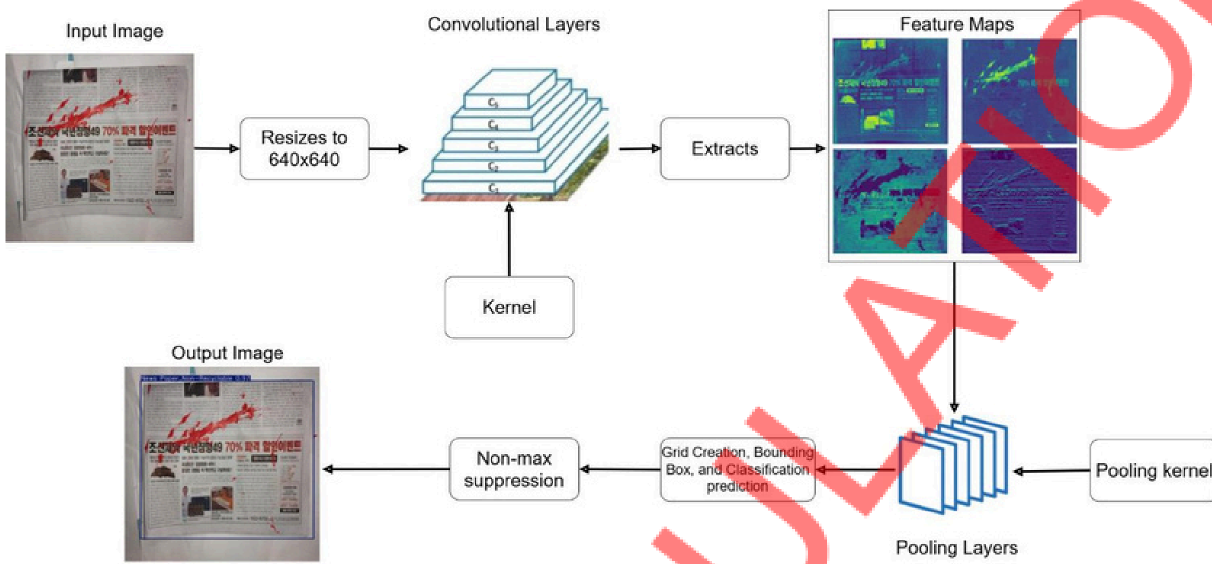


Figure 9. Process Flow of YOLO

Waste Classification Theory

This study's researchers researched the paper industry's status in the Philippines.

According to the study of Parayno, P. P., & Bustamante, M. G. (2006), paper is a fundamental part of modern society, widely used in products like books, newspapers, packaging, tissue, and even in less obvious applications such as insulation, surgical gowns, and cellulose-based derivatives in plastics and toothpaste. Despite its integration into daily life and advanced technologies, paper remains essential for data storage and countless practical uses. It contributes to a global consumption exceeding 600 billion pounds annually, with significant portions dedicated to printing, packaging, and other applications. The study of Parayno, P. P., and Bustamante, M. G. (2006) also indicates that wastepaper is the second most produced solid waste in the Philippines, contributing nineteen percent (19%) of the municipal solid trash in the area, which is made up of wastepaper. After kitchen garbage, it is the second most generated solid waste. Despite the enormous volume

NOT FOR CIRCULATION

of wastepaper production, the Philippine paper industry remains mostly a net importer of its primary raw material, discarded paper goods.

Recyclable wastepaper includes products that can be reused to create new paper goods, such as newspapers, magazines, cardboard, office paper, and packaging materials. According to the National Solid Waste Management Commission (2018), recyclable items include newspapers, magazines, books, cardboard (e.g., tissue boxes, milk cartons), office paper, kraft paper, brochures, catalogs, calendars, envelopes, paper bags, binders, containers (e.g., cigarette boxes, egg cases), phone books, posters, and paper cores (e.g., toilet paper rolls). Properly sorting recyclable paper from other waste and ensuring it is handled by recycling facilities helps conserve resources and minimize environmental impact.

The study of Parayno, P. P., & Bustamante, M. G. (2006), Integration of Solid Waste Management Tools in Specific European and Asian Communities (ISTEAC), supports paper recycling efforts in the Philippines. Their research highlights that office white paper, newspapers, magazines, and telephone directories are common paper types used daily in the country. ISTEAC also evaluates paper mill capacities, showing significant production of industrial-grade packaging paper, newsprint, and kraft paper. Additionally, ISTEAC outlines how Eco-aides and junk shops collect, sort, and sell recyclable paper to dealers, who further separate it into categories such as newspapers, cartons, and white paper for pulping and milling. This process underpins the recycling infrastructure in the Philippines.

NOT FOR CIRCULATION

In the study of Iosip, Alina & Dobon, Antonio & Hortal, Mercedes, and Bobu, Elena. (2012). shows the influence of different contaminants on the recycling of paper. Their study aims to analyze and quantify the environmental impacts of different types of raw paper materials with different levels of contaminants, such as food residues, oils, adhesive residue, and the presence of plastics in wastepaper, which can significantly hinder the recycling process. Their study emphasizes that contaminants increase the environmental impact of recovered paper by necessitating additional processing steps, thereby reducing the overall efficiency of recycling operations.

The study by Tandon, R. & Bhardwaj, N.K. Mathur, R.M., & Kulkarni, A.G. (2001) identify two types of contaminants in the paper: external and internal. External contaminants, which are not chemically or physically attached to paper, can be removed mechanically or manually. Internal contaminants, however, are embedded in the paper and divided into soluble and insoluble types. Among these, insoluble contaminants—further categorized as stickies (e.g., tape, adhesive, food residue, and dyes) pose greater challenges to papermaking and require closer monitoring. Wang, Yun & Marcello, Cornellius & Sawant, Neha & Salam, Abdus & Abubakr, Said & Qi, Dewei, and Li, Kecheng. (2022) further elaborates on stickies' composition, reinforcing their significance in recycling.

This study focuses on classifying recyclable paper by assessing insoluble contaminants, as Tandon et al. (2001) outlined, setting it apart using standardized benchmarks, including NSWMC (2016) and ISTEAC guidelines. By emphasizing the need for monitoring insoluble contaminants, the research enhances methods for distinguishing recyclable from non-recyclable paper, contributing to more precise recycling practices.

NOT FOR CIRCULATION

Conceptual Framework

The conceptual framework applies computer vision, the convolutional neural network of YOLOv8, and waste classification to create an object detection model that can detect and classify recyclable or non-paper waste in real time. In line with waste classification theory, the allocated recyclable paper types align with the standards the National Solid Waste Management Commission (NSWMC) set. This recyclable paper type is the most used daily in the Philippines and shows that this paper type needs to be sorted and recycled (Parayno and Busmente, 2006). With the limitations of computer vision, which needs to be solely on external visual features, insoluble contaminants (e.g., tape, adhesive, food residue, and dyes) are the determinant of recyclable paper being non-recyclable (Tandon et al., 2001). The presence of these contaminants is considered a hindrance and needs to be removed in order to be recycled (Wang et al., 2022). The images created in the model are clean, recyclable paper types based on NSWMC standards, and those containing insoluble contaminants. In image preprocessing, the images were sorted into recyclable and non-recyclable folders to separate the two classes.

In the same phase of image preprocessing, there are applications of resizing and annotating to prepare the image for a deep learning model, which is YOLOv8, following the process of computer vision theory, which aims to make the features in the image extractable in the CNN architecture (Szeliski, 2011).

YOLOv8's CNN applies various blocks to extract the features and then outputs a prediction, which goes to post-processing. The various blocks were organized throughout three main sections in the architecture: the backbone, neck, and head. The backbone is

NOT FOR CIRCULATION

responsible for the initial feature extraction and generates the feature maps from the image. The neck enhances and fuses the features from different levels to handle objects at varying scales. The head takes the refined feature maps and generates the final prediction of the model, which includes the class, bounding box, and confidence score.

In post-processing, where the part of using YOLOv8 is, the NMS is applied to reduce the overlapping boxes and sets a confidence threshold for the final output (Jocher, 2023b).

Finally, the output would be a bounding box of the object along with the predicted label, whether it was recyclable or not, with a confidence score on how confident the model was that it was correct.

The conceptual framework is shown in Figure 10. proposes to detect and classify recyclable and non-recyclable paper and utilizes the YOLOv8 Algorithm by:

- *Image Input:* Images of paper waste (recyclable and non-recyclable) according to NSWMC and ISTEAC guidelines and insoluble contaminants in a paper by Tandon et al. (2001).
- *Image Preprocessing:* This stage involves various techniques like resizing and annotation to prepare data for training.
- *YOLOv8 CNN:* The architecture to train the model, which contains three main parts: the Backbone, the Neck, and the Head. The number of blocks and modules can vary depending on the size of the YOLOv8 variant. Figure 10 displays the total for the YOLOv8 minor variant.

NOT FOR CIRCULATION

- *Backbone*: This part of the architecture extracts features from the input image through several convolutions.
  - Convolutional block x 29: Consists of three layers that perform feature extraction by applying convolutional filters (kernels) across the image, helping to learn abstract features and output feature maps. (Ultralytics, 2023b).
  - C2F (shortcut = true) x 4: The C2F block improved gradient flow and reduced computational complexity. With shortcut=true, it helps prevent the vanishing gradients and better convergence during training (Wang et al., 2023).
  - Bottleneck Block x 8: This reduces dimensionality while retaining the important features with added new features, which enhances the network depth with fewer parameters (Jocher et al., 2024b).
- *Neck*: It acts as the feature aggregator, which combines and enhances multi-scale features to be passed to the head for detecting objects of various sizes.
  - SPPF (Spatial Pyramid Pooling – Fast): It uses pooling operations at different kernel sizes to help recognize objects of their scale or position, enabling multi-scale object detection (Hidayatullah et al., 2025).

NOT FOR CIRCULATION

        - Convolutional Block x 28: Mostly for retaining and adjusting the resolution and depth before passing to the next block (Wang et al., 2023).
        - C2F (shortcut = false) x 4: With shortcut=False, it focuses on the transformed features without residual connections for enhanced localization and blending of features (Wang et al., 2023).
        - Bottleneck Block x 8: Similar to those in the backbone section, but focusing more on refining the fused multi-scale features.
    - *Head*: Is responsible for generating predictions, which include class probabilities, bounding boxes, and confidence scores.
        - Convolutional block x 12: This final block focuses on refining the feature maps and prepares them by mapping them to bounding boxes and class scores.
        - Detect Block x 3: It is produced at three different scales, and each scale helps detect small, medium, and large objects (Redmon et al., 2016).
- *Post-Processing:* YOLOv8 application of NMS and the confidence threshold in the model's inference.
    - *Confidence Threshold:* This filters out detections with low confidence scores to reduce false positive detection, with 0.5 being the default threshold.

NOT FOR CIRCULATION

- ○ *Non-max Suppression:* Eliminates overlapping bounding boxes and retains the highest confidence score bounding box to be outputted.

- *Final Bounding Box with Class:* The model determines if the paper is recyclable or not based on the model's prediction power.

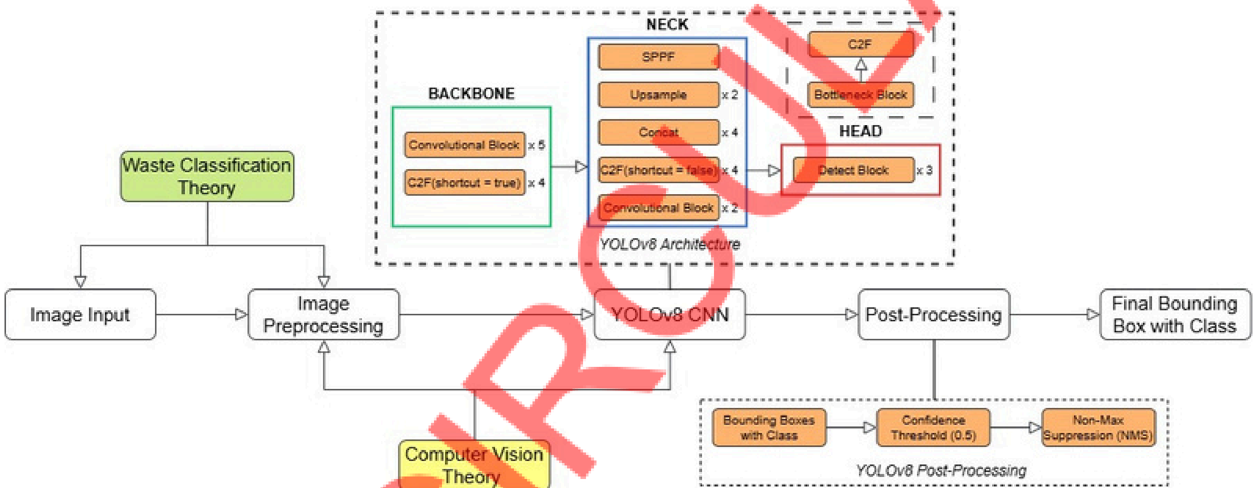


Figure 10. Conceptual Framework

Scope and Limitations of the Study

The study aims to detect and classify non-recyclable papers using the four categories used by dealers in the Philippines to buy and sort recyclable papers. The four categories of papers can be classified as non-recyclable with contaminants that disrupt the recycling and sorting of dealers based on the review of the standard that NSWC releases, which showcases the disruption of the presence of contaminants in the recycling process. The study of Tandon Rita, Bhardwaj Nishi K., Mathur R.M., and Kulkarni A.G. (n.d.) highlighted the insoluble contaminants as more troublesome to papermaking operations, which they say needed closer monitoring.

NOT FOR CIRCULATION

This study aims to build a model with the dataset collected through online sources, self-captured images, and AI-generated images that consist of the four categories of paper with and without insoluble contaminants for detecting and classifying recyclable and non-recyclable paper. The primary focus of this study is to develop a computer vision model using the YOLOv8 algorithm that can detect and classify recyclable and non-recyclable paper based on the standards released by ISTEAC, according to the study of Parayno, Phares P., and Busmente, Mitzi Gay M. (2006).

This study admits some potential biases and limitations. The datasets used to train, validate, and test the model have a significant impact. The quality and variety of the training data determine the YOLOv8 model's accuracy and performance. The researchers' dataset might not cover all possible types of paper, as the dataset only covers the paper types that the dealers of papers in the Philippines accept based on the standard of ISTEAC, according to the study of Parayno, Phares P., and Busmente, Mitzi Gay M. (2006). YOLOv8 assumes that the input images are of high quality and that the objects within the images are distinguishable; any deviation from these assumptions may lead to a decrease in detection accuracy.

The researchers did not change any configuration in the architecture of the YOLOv8 model, as no related research or studies were found at the time of the development of this study to tackle the same concept of detecting and classifying paper recyclability based on contaminants. As the first study to tackle this concept, this study aims to develop a model and test whether YOLOv8 can achieve the objectives.

NOT FOR CIRCULATION

Given that the dataset was created by collecting from online sources, own captured images, and AI-generated images, the dataset may not be reliable enough to be an official dataset available to the public. It can limit the accuracy of correct detection and classification because of the dataset that will be used, as there is still no research that tackles the same concept as this study. There is no available dataset to use or to expand further to train the model.

The study was limited to the versions of Python libraries, software, and hardware specifications that the researchers provided. The training and testing of the model will not be compared with other versions of Python libraries, software, and hardware specification capabilities. The testing of the model was limited to the environment that the researchers provided, and it was not be tested in other environmental conditions.

The study only focused on paper, excluding other types of recyclable materials for accurate classification of recyclable and non-recyclable paper, and this study only utilized YOLOv8; no comparison with other object detection algorithms was performed.

Significance of the Study

The researchers aim for this study to enhance the current body of knowledge in computing and software development while fostering value, social relevance, and awareness among the nation's citizens. Additionally, it contributed to nation-building and the ongoing enhancement of teaching and learning processes, as detailed below.

Promotion of Value and Social Relevance. Studying holds significant value and social relevance. By leveraging advanced AI techniques for waste detection, this research

NOT FOR CIRCULATION

promotes environmental sustainability through improved waste management practices. Identifying paper waste types can streamline recycling processes, reduce contamination in recycling streams, and promote more effective use of resources.

Contribution to Nation Building. The study helped enhance people's awareness and judgment since paper is a common waste, especially in the Philippines, where people tend to misclassify recyclable or non-recyclable paper. The creation of this study benefited the proper knowledge of recyclable paper waste.

Contribution to the Existing Body of Knowledge of Computing. The study seeks to contribute considerably to the current body of knowledge in computing. This study showcases YOLOv8's adaptability and efficacy in real-world applications by combining cutting-edge advances in deep learning and computer vision with the practical task of paper waste management. Furthermore, the findings from this study might spur additional innovation in applying computer vision models to various and complicated problem sets in computing.

Continuous Improvement of the Teaching and Learning Process. The study also served as a solid foundation for future reference by other academics and developers working on a similar problem. It is also an excellent basis for them if they are willing to pursue and surpass the study's limitations on a broader dataset and use other object detection and classification algorithms.

NOT FOR CIRCULATION

Definition of Terms

The researchers have established the following terms to clarify the meanings of the upcoming terminology:

Recyclable Paper. Recyclable wastepaper refers to any paper product that can be reused or repurposed through recycling processes to create new paper products. This includes newspapers, magazines, cardboard, office paper, and packaging materials made from paper.

Recyclable.Refers to materials that can be processed and reused to create new products.

Non-recyclable. Materials that cannot be effectively recycled due to the parameters used in this study.

Computer Vision. This entails creating algorithms and models that let computers detect and classify objects in images or videos.

YOLO. It is considered a state-of-the-art model in computer vision because of its object detection, which can identify and classify objects in images or videos.

YOLOv8. YOLOv8 is an improved version of the YOLO model that produces better results than the previous versions.

Accuracy. A measure of how well a model's predictions match the actual outcomes. The result of the Algorithm's performance when detecting and classifying an object.

NOT FOR CIRCULATION

Datasets. Collections of data used for training and evaluating machine learning models. Datasets provide images for the model to learn from.

Data Preprocessing. Changes the size of the initial images to lessen the training time and speed of the model.

Annotation. Drawing the bounding box and labeling the class of the desired object in an image to train the computer vision model.

Augmentation. Modify the images in the dataset with various alterations to increase the size of the dataset and help generalize the model to various conditions.

Hyperparameter. The configurations in training the model match the model's training environment specifications.

Evaluation. The model's performance and results after the completion of the training.

Feature extraction. Inside the YOLO architecture, convolutional layers extract features and put them into feature maps, which are then used to predict an object in an image.

Non-maximum suppression. Used by the YOLO algorithm to increase the accuracy of predicting and classifying an object in an image or video.

Bounding boxes. The regions that the YOLO algorithm uses. Bounding boxes are commonly used for object localization and classification.

NOT FOR CIRCULATION

Confusion matrix. The table that compares expected and actual labels provides an overview of a classification model's performance, whether the detected object is a paper, and whether it classifies recyclable or non-recyclable.

Acronyms

The following acronyms and their meaning were used throughout the study:

YOLO	You Only Look Once
YOLOv8	You Only Look Once Version 8
OpenCV	Open-Source Computer Vision Library
CNN	Convolutional Neural Network
CS	Confidence score
NSWC	National Solid Waste Commission
SPPF	Spatial Pyramid Pooling Fast
C2F	Cross-Stage Partial
ISTEAC	Integration of Solid Waste Management Tools in Specific European and Asian Communities
mAP	Mean Average Precision

NOT FOR CIRCULATION

Chapter 2

REVIEW OF RELATED LITERATURE AND STUDIES

This chapter reviews the existing approaches, the same algorithm, or models for object detection and classification. It is important to review if there are studies that share the same concept and approach. In addition, this chapter briefly discusses the paper recycling scheme, parameters of paper that were considered non-recyclable to include in the dataset, how YOLO detects and classifies an object, and related studies that use the same algorithm in line with waste management.

Recyclable Papers

According to the Department of Trade and Industry's Bureau of Philippine Standards (2024), The Philippine Development Plan, Ambisyon Natin 2040, the United Nations Sustainable Development Goals, and the Department of Trade and Industry's

Bureau of Philippine Standards (DTI-BPS) have all outlined national policies and international commitments on the environment, industry, innovation, and economy that the Philippines hopes to achieve. Recycling is essential for sustainable development since it reduces waste, conserves natural resources, and minimizes environmental damage. One of

the most important components of recycling is handling recyclable paper, which significantly adds to the trash stream. Understanding the various forms of recyclable paper and the conditions governing their recyclability is critical for effective waste management.

NOT FOR CIRCULATION

According to the study of Parayno, P. P., & Bustamante, M. G. (2006), Integration of Solid Waste Management Tools in Specific European and Asian Communities (ISTEAC) releases paper recycling schemes in the Philippines that review, assess, study, and propose solutions for the recycling of paper in the Philippines. In their study, they assess and show the annual paper mill capacity, which shows that different types of paper, newsprint, printing and writing, assorted, and cardboard, are the most produced paper annually. In this sector, ISTEAC's research also shows that the main types of paper used daily in the Philippines are white paper, newspapers, assorted, and cardboard. ISTEAC also shows the process of how Eco-aides employs the collection of recyclable paper, which indicates that junk shops accept and segregate assorted paper, newspapers, cartons, magazines, and selected white, and sell it to dealers, where dealers accept and segregate it again for pulping and milling the recyclable paper. Dealers segregate the wastepaper from white paper (with print), Old newspaper, Mixed/Assorted paper, and cardboard.

Parameters of Paper

The research of the study of Parayno, P. P., and Bustamante, M. G. (2006), the ISTEAC releases paper recycling schemes in the Philippines that review, assess, study, and propose solutions for the recycling of paper in the Philippines that showcase the accepted recyclable papers from the dealerships. Considering the recyclability of different scrap papers, there are guidelines released by the National Solid Waste Management Commission (2018) for identifying the quality standard for acceptance of the recyclability of scrap papers, in which the recyclable paper should not be mixed with different types of contaminants. Highlighting the idea of the presence of contaminants, the study of Tandon

NOT FOR CIRCULATION

Rita, Bhardwaj Nishi K., Mathur R.M., and Kulkarni A.G. (n.d.) shows the two types of contaminants, which are external and internal. Their study also highlighted that insoluble contaminants are considered more troublesome to papermaking operations, which they say need closer monitoring. Their study shows that dyes and pigments can distinguish insoluble contaminants. In the study of Yun Wang et al. (), contaminants are also composed of tape residue, adhesive residue, food residue, dyes, and pigments. The researchers consider these contaminants to be the common types the dealership faces.

According to the study of Wahyutama, Aria Bisma, and Mintae Hwang. (2022), focusing on multiple waste types, including paper, shows that the YOLO algorithm can recognize other waste types' of specific characteristics and variations. By concentrating on paper, the YOLO model can be trained more effectively, increasing its accuracy in identifying different paper types. They created their dataset by manually downloading it from a search engine, crawling it on the web, and downloading it from the dataset provider's website. This was demonstrated in the studies of Wahyutama, Aria Bisma, and Mintae Hwang. (2022), where modified YOLO models achieved high precision rates in detecting various paper types when tested under controlled conditions. Based on the study of Agbehadji, Abayomi, Nam, Milham, and Freeman (2022), their study consists of different types of waste images or photographs to serve as classes that is vital in determining whether the image or photograph of trash is classified correctly. They used the YOLO algorithm with Convolutional Neural Networks (CNN) for waste object detection and classification. They only use versions 3 and 4 of YOLO, and their database consists of eight classes of waste images and does not specify if it is recyclable or not. Another study by Francesco Bianconi, Luca Ceccarelli, Antonio Fernández, and Stefano A. Saetta (2014) justifies the

NOT FOR CIRCULATION

exclusive focus on paper for image processing. Early detection of impurities reduces the need for chemicals in later stages, benefiting the environment. A proper dataset can make detecting paper accurate, and determining the accuracy of impurity detection in paper makes it well suited for image processing techniques.

There is research about waste classification, but there is no research that classifies the recyclability of paper based on the standards of the dealership that this study focused on recyclability. Research on computer vision for segregating recyclable waste papers was conducted by Rahman, Hannan, Scavino, Hussain, and Basri (2011). The research uses a webcam and a different kind of algorithm but only identifies OCC (Old Corrugated Cardboard), ONP (old newspaper), and WP (White Paper) and does not classify if it is recyclable. Their research recommends that further work should focus on identifying more paper types. When it comes to impurities in paper, there is similar research by Bianconi, Ceccarelli, Fernández, and Saetta (2014), using machine vision methods to distinguish faulty paper patches (with impurities) from non-defective ones (without impurities), as well as other algorithms, but with just 11 paper samples and a concentrate on machine vision operations.

YOLO Algorithm

Object Detection

According to the creators, Redmon, Divvala, Girshick, and Farhadi of You Only Look Once: Unified, Real-Time Object Detection, the YOLO algorithm divides the input image into an S x S grid. Each grid cell is in charge of detecting an object if its center falls within its grid cell. They asserted that each grid cell predicts B bounding boxes and

NOT FOR CIRCULATION

confidence scores for those boxes. According to their findings, confidence scores reflect the model's level of certainty that the box contains an object and its estimate of the box's accuracy in terms of predictions. The confidence score is zero if the cell contains no objects; otherwise, it equals the intersection over the union (IOU) of the predicted box and the ground truth. Each bounding box has five predictions: x, y, w, h, and confidence. According to them, the box's width and height are anticipated for the whole image, while the (x,y) coordinates represent the box's center for the complete image. The confidence represents the IOU between the predicted and ground truth boxes. According to Wilson, Dr. Manusankar, and Dr. Prathibha's (2022) analysis of object recognition in the YOLO method, the YOLO model uses final weights to determine whether to detect bounding boxes, which are weighted depending on probabilities.

The study of Ahmed and Shanto (2024) found that the YOLO architecture converted every image into a grid and predicted bounding boxes and class probabilities for each cell simultaneously. This enabled the model to process photos in real-time, making it highly efficient for practical applications. Flores-Calero, Astudillo, Bustillos, Maza, Lita, Defaz, Ante, Zabala-Blanco, and Armingol (2024) discovered that the YOLO algorithm breaks an image into grid cells and finds objects inside them. They mentioned that YOLO's detection result is calculated utilizing the non-maximum suppression approach and a confidence rate threshold. According to Han, Cao, Zheng, Chen, Wang, and Fu (2023), YOLO uses a grid of size S x S on the input image, and in each grid cell, multiple bounding boxes and their corresponding class probabilities are predicted. They use YOLOv8 because of its high real-time performance, accuracy, and scalability.

NOT FOR CIRCULATION

In the study of Safaldin, Zaghden, and Mejdoub (2024) about using YOLOv8 to detect moving objects. They stated that the architecture of YOLO employs a Convolutional Neural Network (CNN) for efficient and instantaneous object detection. Within the system, images are fed in and carried through the robust framework, consisting of convolutional layers engineered to extract abstract features necessary for accurate object detection. They also said the same as other studies about the YOLO algorithm, starting with an input image covered in a grid cell, and each cell is assigned the task of detecting objects. YOLO predicts multiple bounding boxes and their corresponding confidence levels for every grid cell, which pinpoints the location and presence of objects.

According to the study of Zhao (2024), the YOLO approach treats target identification as a regression issue, and the network directly analyzes the input picture to output the type, location, and confidence level of the identified item. The YOLO object detection method splits the input picture into grids, with the center of the target item to be identified falling within one of the grids; that grid is in charge of identifying the target object.

Also, the study of Zailan, Nur & Mokhzaini, Azizan & Hasikin, Khairunnisa & Khairuddin, Anis, and Khairuddin, Uswah. (2022) applied object detection using YOLO, starting with applying logical SxS grids to a picture to interpret it. Then, the weighted feature sets are calculated to determine the likelihood of each cell region. A preliminary bounding box was generated based on the prediction probability ascertained by the trained model if the center of a potential item is contained inside one of the cells. The detection model utilizes numerous boxes to predict and extract a 3D tensor based on the defined number of classes. It calculates the bounding box for predictions using width and height

NOT FOR CIRCULATION

parameters and offsets from the cluster centroid. After the cell is offset from the upper part of the object detection with a small inference cost increment, the model processes further.

The study of Roy, Pronoy & Islam, Md. Riadul, and Rafi, Alam. (2024) focuses on deep learning models for their study, primarily using the YOLOv8 architecture for object detection. Their study started with data collection, which is required for training their model with a dataset of images, followed by data augmentation, which implemented rotation, flip, horizontal and vertical transformation, exposure adjustments, noise, blur, and mosaic patterns. Their study applied the detection before classification, which is vital for increasing the model's accuracy. They chose YOLOv8 among various CNN architectures for detection, as it is one of the fastest object detection models specifically designed to focus on constrained edge device deployment, reduce the classification time, and increase accuracy.

Object Classification

According to Redmon et al.'s (2016) study, "You Only Look Once: Unified, Real-Time Object Detection," the YOLO model for object classification states that each grid cell forecasts C conditional class probabilities. Their algorithm's probabilities depend on the grid cell containing an item. Their approach predicts a single set of class probabilities per grid cell, independent of the number of boxes. They multiply the conditional class probabilities and individual box confidence forecasts to obtain the class-specific confidence ratings for each box. The scores represent the likelihood of that class occurring in the box and how well the projected box matches the item.

NOT FOR CIRCULATION

In the study of Marium, Srinivasan, and Shetty (2020), it is stated that YOLO is a regression object detection algorithm that generates region proposal networks and classifies them simultaneously. The architecture style of YOLO includes regression for finding bounding boxes in images and confidence scores of object classes for each bounding box. YOLO's convolutional neural network (CNN) predicts the bounding boxes and class probabilities for all objects depicted in an image (Diwan et al., 2022). The YOLO algorithm is a one-stage algorithm and uses only one network to predict the bounding box coordinates and the classification probability, so it is swift (Huang et al., 2024). The CNN architecture that the YOLO uses is responsible for preprocessing the input image in the YOLO model to extract features, which helped to detect or classify objects (Roy, Pronoy et al.,2024). Their study used YOLOv8 because it is specifically designed to focus on constrained edge device deployment at high inference speed.

The study of Maity and Seba (2023) used Yolov8 for waste classification for several reasons. The YOLO (You Only Look Once) algorithm was selected and applied in this project. One main contributing element is YOLO's real-time object identification capabilities, which comply with the waste categorization system's specifications. The objective of their study is to efficiently and quickly classify waste products. YOLO is a good fit for this application as it can process photos in a single pass, enabling real-time detection. The YOLO algorithm also provides a decent trade-off between accuracy and speed. The YOLO technique in their research predicts bounding boxes and class probabilities for discovered items inside trash photos concurrently using a deep neural network architecture. The network learns the traits and features required for precise categorization by being trained on a labeled dataset of waste items. The trained YOLO

NOT FOR CIRCULATION

model scans the input photos, examines the content, and produces bounding boxes and matched class labels for waste objects recognized during inference. Their trash classification system provides real-time item identification by utilizing YOLO, which makes garbage segregation effective and automatic. The quickness of the algorithm guarantees that waste products may be sorted quickly and precisely, enhancing waste management procedures. The YOLO-based method is adaptable to various waste materials since it can effectively handle waste items of different sizes and forms. Integrating the YOLO algorithm in this project makes waste categorization effective, precise, and real-time, enhancing waste management procedures.

Object Detection and Classification

The study of Maity and Seba (2023) used YOLOv8 for waste detection and classification for several reasons; the YOLO (You Only Look Once) algorithm was selected and applied in this project. One main contributing element is YOLO's real-time object identification capabilities, which comply with the waste categorization system's specifications. The objective of their study is to efficiently and quickly classify waste products. YOLO is a good fit for this application as it can process photos in a single pass, enabling real-time detection. The YOLO algorithm also provides a decent trade-off between accuracy and speed. The YOLO technique in their research predicts bounding boxes and class probabilities for discovered items inside trash photos concurrently using a deep neural network architecture. The network learns the traits and features required for precise categorization by being trained on a labeled dataset of waste items. The trained YOLO model scans the input photos, examines the content, and produces bounding boxes and matched class labels for waste objects recognized during inference. Their trash

NOT FOR CIRCULATION

classification system provides real-time item identification by utilizing YOLO, which makes garbage segregation effective and automatic. The quickness of the algorithm guarantees that waste products may be sorted quickly and precisely, enhancing waste management procedures. The YOLO-based method is adaptable to various waste materials since it can effectively handle waste items of different sizes and forms. Integrating the YOLOv8 algorithm in this project makes waste categorization effective, precise, and real-time, enhancing waste management procedures.

The study of Shrestha, Shiva & Shrestha, Samman & Paudel, Prateek & Gurung, Sudarshan & Gaire, Sulav & Adhikari, and Smita. (2024) presents a novel deep learning- based waste classification system using the state-of-the-art object detection framework You Only Look Once v8 (YOLOv8) to address urgent environmental concerns. Their study highlighted that waste categorization is essential to efficient waste management because it allows different waste kinds to be separated for proper disposal, recycling, or composting. Their method uses YOLO v8 to identify and group waste materials into four classes: biodegradable, paper, plastic, and metal. Utilizing the improved accuracy and real-time processing of YOLO v8, their study offers a workable automated garbage-sorting solution. Convolutional neural networks (CNNs) are used to build the YOLO technique, allowing for real-time object recognition and classification. Their study model demonstrated high precision (94.12%) and recall (96.92%), indicating that waste items were thoroughly detected. Their model did well in recognizing items, with a mAP score of 98.25%. Furthermore, when tested with images with various factors such as angle, position, lighting, and resolution, the results show that the model can classify objects with even higher accuracy.

NOT FOR CIRCULATION

Synthesis

Recycling is critical in sustainable development, as it reduces waste, conserves resources, and minimizes environmental damage. In the Philippines, national policies and international commitments such as the Philippine Development Plan, Ambisyon Natin 2040, and the United Nations Sustainable Development Goals all emphasize the importance of recycling, particularly for recyclable papers, which comprise a large part of the waste stream. Effective waste management requires thorough examination of The Integration of Solid Waste Management Tools in Specific European and Asian Communities (ISTEAC). The integration of solid waste management tools in the Philippines has been examined in a paper on recycling schemes. ISTEAC's research identifies major types of paper used, including newsprint, writing, assorted papers, and cardboard, noting that junk shops collect and segregate various types, such as newsprint and cardboard, before selling to dealers for pulping and milling.
Different studies, such as those by Parayno and Busmente (2006), highlight quality standards for recyclable paper. Guidelines by the National Solid Waste Management Commission (2018) underscore that recyclable paper should be free from contaminants. Contaminants, classified into external (e.g., dyes and adhesives) and internal (e.g., tape residue), reduce the quality of recycled paper, as discussed by Tandon et al. Their findings emphasize the need to closely monitor insoluble contaminants, which can affect pulping and recycling.

The YOLO algorithm is widely used in real-time object detection, converting images into a grid and predicting bounding boxes and class probabilities for each cell.

NOT FOR CIRCULATION

YOLO operates by dividing an input image into a grid, with each cell predicting bounding boxes and class probabilities based on object presence. Pioneered by Redmon et al., YOLO achieves real-time detection by calculating bounding boxes and classifying objects in a single pass. Recent adaptations, including YOLOv8, further improve accuracy and performance, making it suitable for real-time waste sorting and classification tasks. YOLO's ability to generate bounding boxes and confidence scores for objects within each grid cell enhances its efficiency in applications requiring fast object recognition, such as waste management.

Studies have shown that YOLO can be effectively trained to detect various waste types, including different paper categories, by using large and diverse datasets. Notably, research by Wahyutama and Hwang (2022) demonstrated YOLO's high precision in detecting different paper types under controlled conditions. Further studies by Agbehadji et al. and Bianconi et al. suggest that YOLO's ability to distinguish between recyclable and non-recyclable waste can be enhanced with well-prepared datasets and image processing techniques to identify impurities early in the recycling process.

Research by Maity et al. has shown YOLOv8's effectiveness in waste classification systems, where real-time detection is essential for efficient waste sorting. The YOLO architecture, trained on labeled datasets, accurately identifies and categorizes waste, facilitating real-time item recognition. This makes YOLO a practical choice for recycling applications that require the rapid differentiation of recyclable from non-recyclable waste. The model's ability to predict bounding boxes and classify objects simultaneously enables efficient processing and aids in automating the waste segregation process.

NOT FOR CIRCULATION

In summary, while existing studies showcase YOLO's capabilities in detecting and classifying various waste types, further research is needed to create a classification model that meets the recycling standards specific to the Philippines. Incorporating these standards into a YOLO-based model for recyclable paper classification could greatly enhance local recycling processes, aligning with the national goals for sustainable development. By refining datasets and enhancing detection algorithms, the research aims to improve the model's ability to identify paper recyclability based on contamination levels and other dealership standards. However, existing studies show a need for further development to accurately classify recyclable versus non-recyclable paper, a gap this research seeks to address—a rough understanding of recyclable paper types and the conditions for their recyclability.

NOT FOR CIRCULATION

Chapter 3

METHODOLOGY

This section discusses the process of the suggested technique, the YOLOv8 algorithm, for recognizing and categorizing recyclable and non-recyclable paper. This section discusses data collection, the dataset, the model's workflow, and model assessment.

Project Design

The widespread use of paper in the Philippines poses a significant increase in paper waste, and due to the Philippines' poor paper waste management, paper waste recycling becomes a challenge. People have found it difficult to determine whether the paper is recyclable. The study proposes and aims to develop a model to detect and classify whether paper waste is recyclable.

This study aims to collect images of papers within the scope of this study, as well as papers with and without the factors that determine whether or not they are recyclable, and to preprocess and prepare the acquired dataset for model training. The model was developed using computer vision methods such as data annotation, data augmentation, object detection, and classification. Finally, the study seeks to use an algorithm regarded as a state-of-the-art computer vision model by Ultralytics (2024) called YOLOv8.YOLOv8 uses a Convolutional Neural Network (CNN) to extract features from the gathered dataset and to recognize papers in an image. YOLOv8 can also classify whether or not the paper is recyclable, depending on the parameters used in this study.

NOT FOR CIRCULATION

YOLOv8 supports various image formats such as BMP, DNG, JPEG, JPG, WEBP, MPO, PFM, PNG, and TIFF. Also, video formats such as ASF, AVI, GIF, M4V, MKV, MOV, MP4, MPEG, MPG, TS, WMV, and WEBM (Ultralytics, 2024). Large volumes of data from many sources should be supported and processed by the model for the training and validation sets. Data processing and computer vision methods, including data annotation, data augmentation, object detection, and classification, are included in the implementation of the model. Lastly, based on the collected dataset, the model should detect recyclable and non-recyclable paper waste.

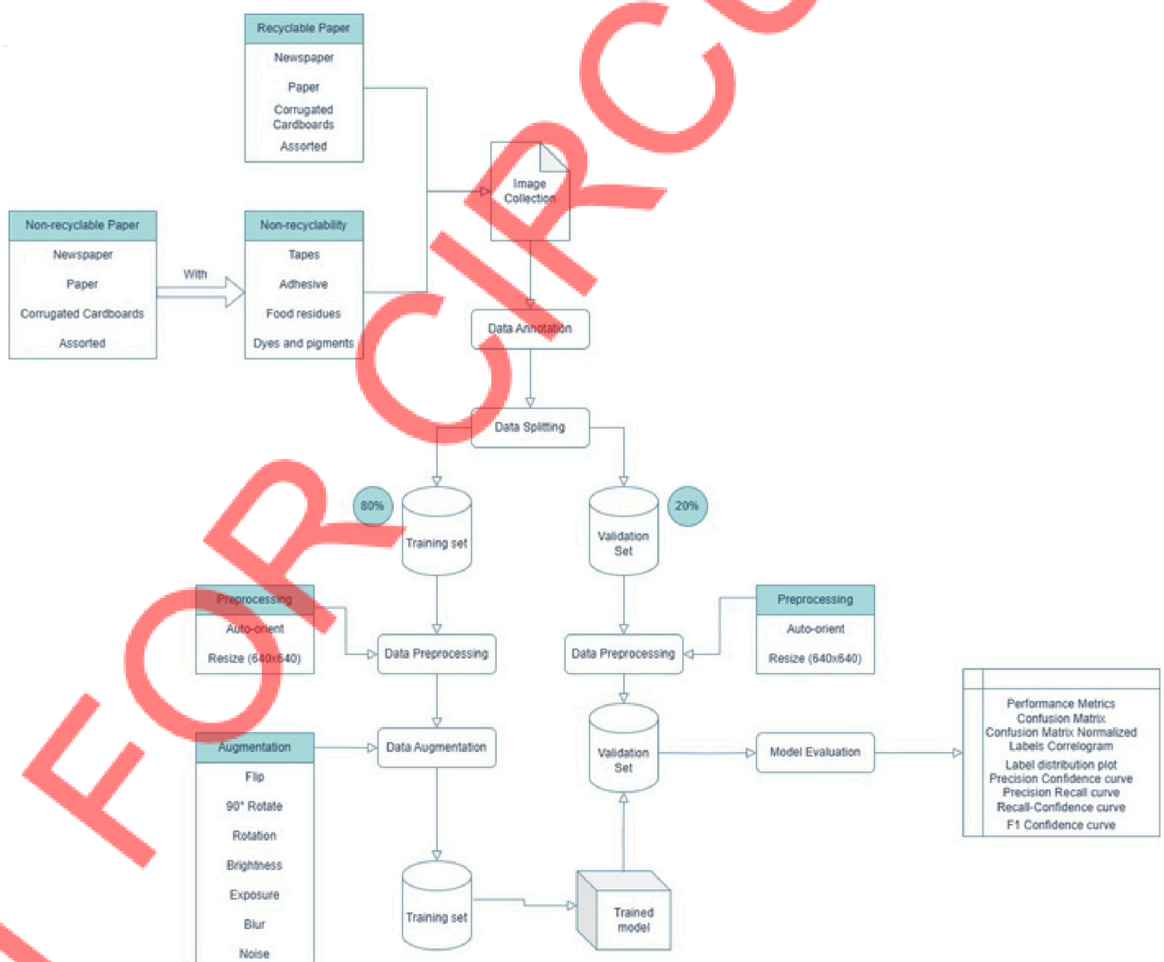


Figure 11. Training Process and Flow of YOLOv8 Model

NOT FOR CIRCULATION

Figure 11 shows the process of how the model flow of YOLOv8 works. The first stage is to gather images of paper that are recyclable and non-recyclable based on the parameters that this study used from different online sources on the internet, self-captured images, or AI-generated images. The next stage is to annotate each image in the collected data. By annotating the data, each image is manually labeled by a bounding box and assigned a class for classification. After annotating the images, they were split into 80% training and 20% validation sets. Preprocess the training and validation set of auto-orient, a feature in Roboflow Annotate that stores the images' pixels the same whether the image was oriented (Dwyer, 2024). The other preprocess is resizing the images by 640 x 640 to reduce training time, and the resolution required for training YOLOv8 is 640 x 640. Data augmentation of the training set: Augmentation is applied to generate augmented versions of each image in the dataset to diversify the conditions of an image and train the model. Roboflow Annotate has various augmentations, but the researchers only apply the appropriate augmentation, which in flip, 90° rotate for camera orientation, rotation for camera roll, brightness for lighting conditions, exposure to affect the lighted areas, blur for camera focus, noise for camera artifacts, mitigate adversarial attacks, and can help avoid overfitting (Nelson, 2020). After augmenting the training set, the training set was used to train the YOLOv8 model. The final stage results from the model's performance and determines the accuracy of object detection and classification of recyclable and non-recyclable paper with visualized metrics.

Data Collection, Annotation, Preprocessing, and Augmentation

The components created the model for detecting and classifying recyclable and non-recyclable paper. Based on this study's parameters, the researchers gathered images of

NOT FOR CIRCULATION

papers only included in the scope and non-recyclable papers for data acquisition. The data processing components of this study used the annotation, preprocessing, and augmentation tool, which is Roboflow Annotate, as it is supported by YOLOv8 and much easier to use than other annotation tools. The computer vision algorithm, which is YOLOv8, was implemented using Python, OpenCV, Ultralytics, and Roboflow libraries.

The allocated dataset contains 6,197 images (before data augmentation) in total. The count that covers the paper is 5,298 images covering eight classes, which are recyclable newspaper, paper, cardboard, assorted, and non-recyclable newspaper, paper, cardboard, and assorted, which consist of images with tape, adhesive, food residue, dyes, and pigments. There are 899 background images or null images that do not contain the desired object to train the model on what to detect and not detect. The images in the dataset are a mixture of online sources, self-captured, and AI-generated images. Most non-recyclable images with food residues are AI-generated images to diversify different kinds of possible food residues. The dataset also has images that contain multiple objects and different classes in an image.

Table 1. Dataset Composition

Paper	Newspaper	Cardboard	Assorted
White paper Envelope Opened Book	Newspaper	Appliance Box Paper box container	Book cover Magazine Calendar Colored Paper Brochure
Non-Recyclable			
Tape Adhesive Food residue Dyes and pigments	Tape Adhesive Food residue Dyes and pigments	Tape Adhesive Food residue Dyes and pigments	Tape Adhesive Food residue Dyes and pigments

NOT FOR CIRCULATION

Each image goes through a meticulous manual annotation to create a bounding box for the object and label it to its desired class. This process is performed using a website called Roboflow, which is a preprocessing, annotation, and augmentation tool. The specific steps include inspecting each image, creating a bounding box for the object, and labeling it to predefined eight classes.

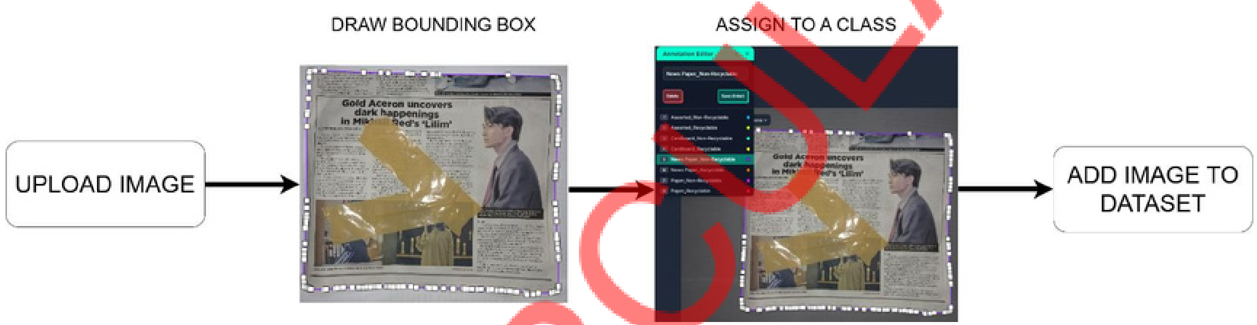


Figure 12. Dataset Annotation Process

After annotating, it was split into an 80% training set and a 20% validation set, 4,957 images for the training set and 1,240 images for the validation set. After splitting the dataset into training and validation sets, preprocessing is performed, which includes auto-orientation and resizing the resolution. Auto-orient contains metadata that dictates the orientation by which it should be displayed relative to how the pixels are arranged (Dwyer, 2024). Auto-orient is to correct the bounding box coordinates after it is augmented. Resizing images is a critical preprocessing step, which makes the training faster (Nelson, 2020). The default resizing of YOLOv8 is 640 x 640; resizing in the preprocessing step cuts down the time it takes for YOLOv8 to resize the image.

NOT FOR CIRCULATION

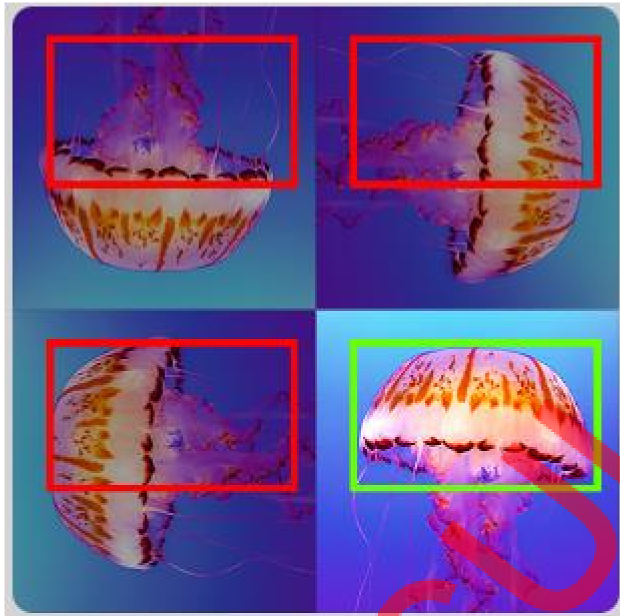


Figure 13. Auto-orient Example

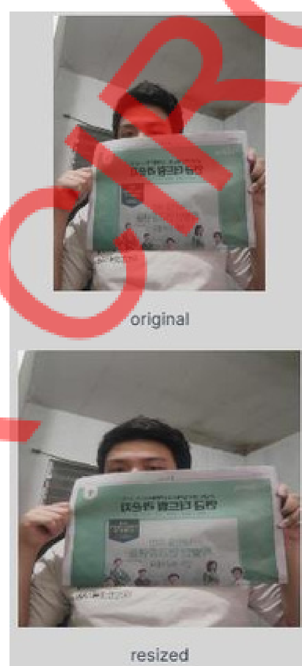


Figure 14. Resize Example

The dataset was augmented to further solidify the images for various conditions. The data augmentation process enriches the training dataset, providing a variety of images to improve the accuracy and generalization of the model in detecting and classifying objects (Syahrudin, E., Utami, E., and Hartanto, A. D., 2024). The model can be more

NOT FOR CIRCULATION

effective in real-world conditions. The augmentations included are flip: flip the image vertically and horizontally; 90° rotate: rotate an image to 90 or 180 degrees; rotation: randomly rotate to 15 or -15 degrees; exposure: adjust the gamma exposure of the image; brightness: adjust the image brightness, which includes all parts of the image; blur: adjust the blurriness of the image; and noise: to be more resilient camera artifacts.

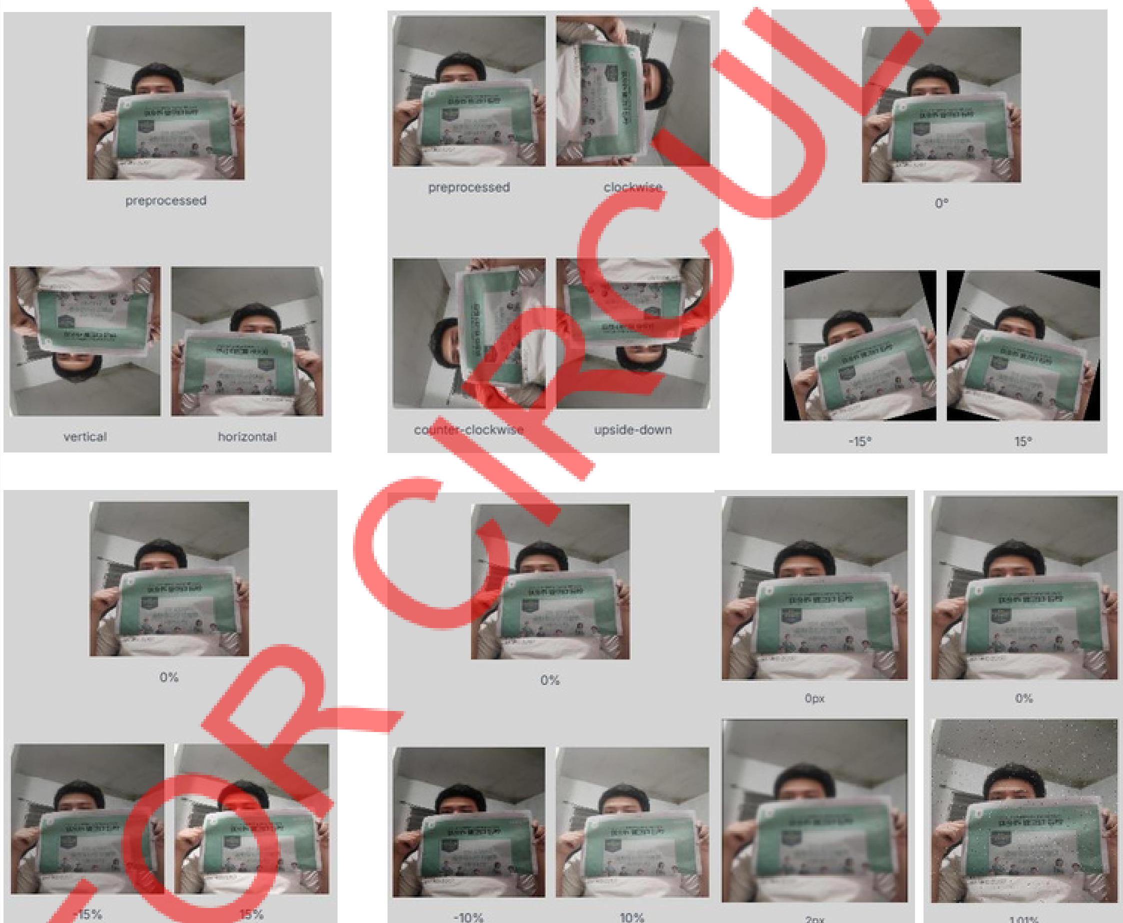


Figure 15. Image Augmentations

NOT FOR CIRCULATION

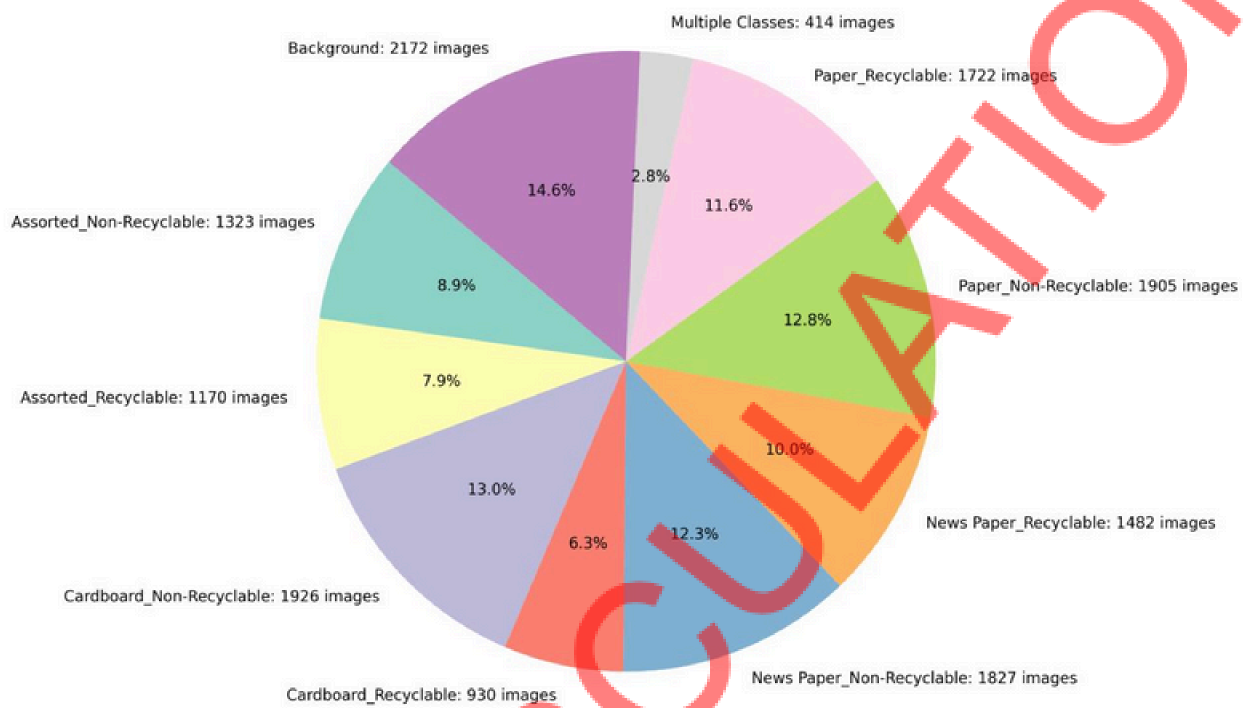


Figure 16. Distribution of Images in the Training Set

After the augmentation process, the images in the training set increased to 14,871 images. In Figure 16, it was visualized using the matplotlib library. The multiple classes slice was just added to represent images in the dataset with multiple classes in an image and were not part of the eight classes. The figure represents the number of images for each class and the background image in the training set.

NOT FOR CIRCULATION

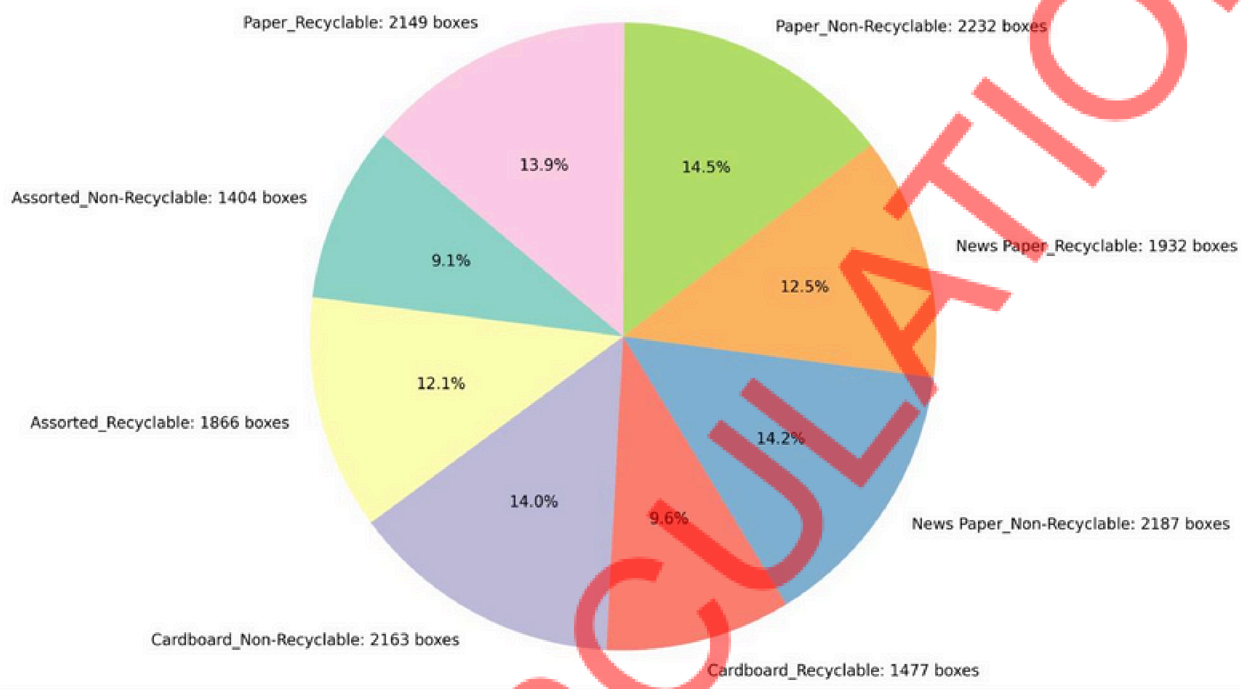


Figure 17. Distribution of Bounding Box per Class in the Training Set

Figure 17 visually represents the number of bounding boxes and labels in the training set. Apart from the number of images, each image can contain two or more bounding boxes of the same class or different classes, as shown in Figure 17, which is the count of all bounding boxes in the respective classes in the training set.

Model Training

Model training is the deciding step in developing a computer vision model. It teaches the YOLOv8 model to detect and classify objects in a frame correctly. Export the annotated, preprocessed, and augmented dataset, which in this study uses Roboflow. Export it in YAML format, for which Roboflow has a download code to download the dataset to the designated directory directly. To train a YOLOv8 model, install the Ultralytics library, which loads the YOLOv8 model. The study also installs a torch and

NOT FOR CIRCULATION

CUDA toolkit to utilize the system's GPU instead of CPU to train the model. Before the initial training, some hyperparameters need to be set to accommodate the hardware requirements. The study does not change hyperparameters to train YOLOv8 and uses the default values of YOLOv8. Why only these hyperparameters were changed will be discussed in Chapter 4.

Table 2. Hyperparameters that will be tuned for training

Hyperparameter	Description
data	Path to the exported dataset configuration file containing training and validation images and labels.
batch	The memory usage to train the model. The higher the GPU memory usage. The default is 100.
epoch	Total number of training epochs. Each epoch passes over the dataset and adjusts its weight to refine its detection and classification. The default is 100.
patience	Number of epochs to wait without improvement to initiate early stopping to avoid overfitting. The default is 100.
amp (Automatic Mixed Precision)	It speeds up training by reducing memory usage but has a minimal impact on accuracy. The default is True.

After the hyperparameter set, load the YOLOv8 model and train it on the exported dataset's YAML file. The training time depends on the hardware specification, the length of epochs, and the batch size.

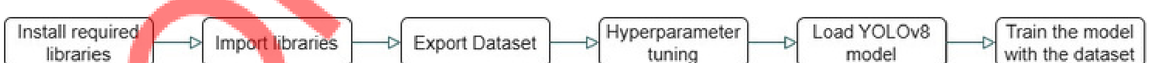


Figure 18. Training Implementation of the YOLOv8 Model

Project Development

The Python programming language was used in the entire development because it has a wide range of libraries intended for machine learning. The libraries and their functions are summarized in Table 3.

NOT FOR CIRCULATION

Table 3. Python Libraries

Library Version	Description
OpenCV 4.10.0	Used to capture frames from the webcam, draw bounding boxes Moreover, labels around detected objects in those frames display the real-time annotated video feed.
Roboflow 1.1.48	Used to access a specific project and dataset for the researchers YOLOv8 model, enabling the researchers to download the formatted dataset needed for training and inference.
Ultralytics 8.3.26	Used to load pre-trained YOLOv8 mode, specifically yolov8s.pt version enables the model to perform object detection, classification tasks, and training.
Torch 2.5.1+cu118	Used as the underlying framework for training the YOLOv8 model, leveraging its capabilities for tensor operations and automatic differentiation during the training process.
NVCC 12.6.77	NVidia CUDA Compiler used to compile and run CUDA code, which enables GPU acceleration for YOLOv8 model training and inference, enhancing performance by leveraging the parallel processing capabilities of NVIDIA GPUs.
Matplotlib 3.9.2	Used to visualize the training results, such as plotting loss curves or displaying detection results, which helps in analyzing the model's performance and making adjustments as needed during the development process

Table 4 shows the other software specifications necessary for the model training and system development, and Table 5 shows the hardware specifications needed.

Table 4. Software Specifications

	Model Training Requirements	System Development Requirements
Programming Language	Python 3.7 or higher.	Python 3.8 or higher.
Integrated Development Environment	Jupyter Lab 4.1.6 or Pytorch 1.10+	Jupyter Lab 4.1.6 or Pytorch 1.10+
Operating System	-	Windows 11 or later compatibility with the versions of CUDA and cuDNN intend to use.

NOT FOR CIRCULATION

Table 5. Hardware Specification

	Components	Descriptions
Processor	Multi-core processor intel i5/i7 or AMD Ryzen 3 minimum	Higher clock speed for better performance, especially for preprocessing tasks.
GPU	NVIDIA GPU with at least 4 GB of VRAM (e.g., GTX 1660) with CUDA 11.2+	Consider higher-end GPUs (e.g., RTX 2060, 30 series) for better performance. CUDA support is essential for efficient model inference.
Memory (RAM)	8GB+ RAM	8 GB or more, especially for larger models and batch processing.
Storage	50 GB+ free disk space (for dataset storage and model training)	Sufficient space to store model weights and datasets.
Camera	Basler Ace2 Basic	Recommendation for most computer vision applications, covering most use cases. This camera strikes an excellent balance between performance and cost, making it ideal for a wide range of tasks and efficient readings of frame rates.

Operation and Testing Procedure

The researchers set a testing environment during the real-time test to address the model's limitations in various environments.

Preliminary Test

The model was initially tested on a series of images, and the categories of papers and their non-recyclable counterparts were tested. The researchers provide a set of images and observe the model's performance in accurately detecting and classifying the paper with its correct class. The images were random and came from different online sources, as well as self-captured images of the researchers. The researchers also manually counted the correct predictions, created a confusion matrix, and computed the model's score results. The images that were tested consist of 5 images in each class and 10 null images or images that do not contain the eight classes, totaling 50 images.

NOT FOR CIRCULATION

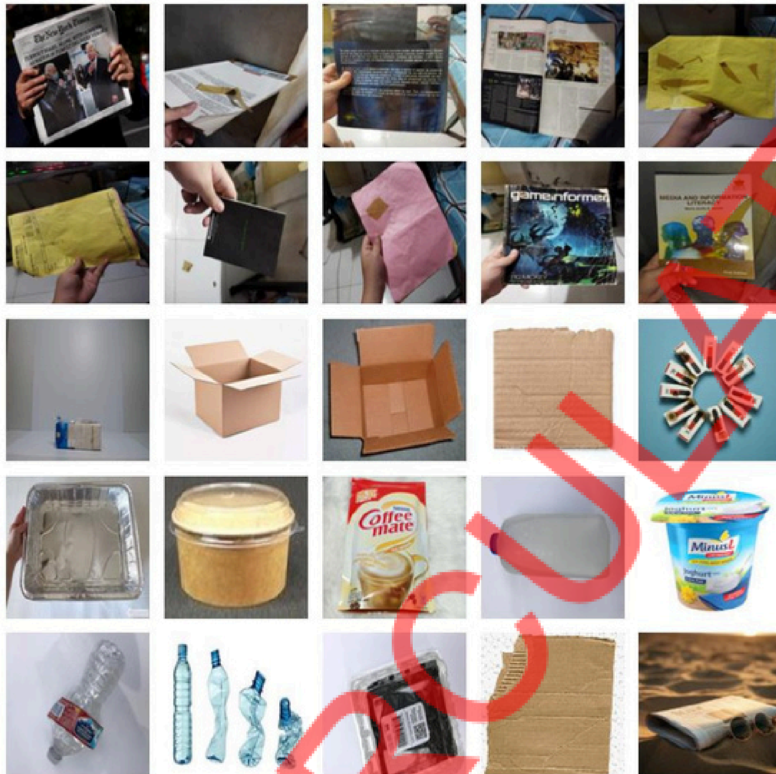


Figure 19. Sample Images to test the model

Real-time Test

Using a webcam/camera, the researchers utilized OpenCV libraries to test the YOLOv8 model in real time. The researchers used real examples of 4 categories of papers with and without contaminants. The classes for recyclable paper were Paper_Recyclable, News Paper_Recyclable, Cardboard_Recyclable, and Assorted_Recyclable, which consist of colored paper, covers of books, magazines, and calendars. The non-recyclable classes was Paper_non-Recyclable, News Paper_Recyclable, Cardboard_non-Recyclable, and Assorted_non-Recyclable, with contaminants.

The researcher provided the environment that was used for testing the model in real-time, in which the object was placed on a table with a black background. It limits the

NOT FOR CIRCULATION

faulty detection of the model, was not tested in other environments, and focuses on whether the model predicts and classifies the papers.

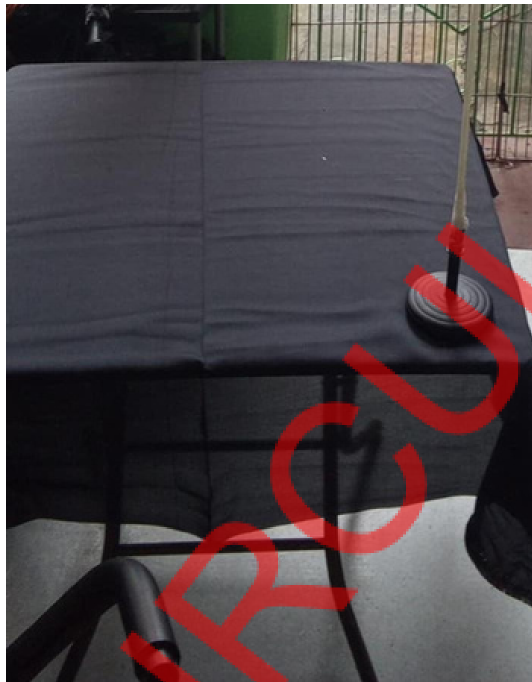


Figure 20. Testing Environment

1. Observation for the detection of the model:

   a. Testing the model's distance that can detect an object, the paper was placed on the table, and the researchers defined the distance by moving the camera far away from the detected object.

   b. During real-time detection testing, the frames per second (FPS) were displayed to observe the speed of detection of the model while running.

NOT FOR CIRCULATION

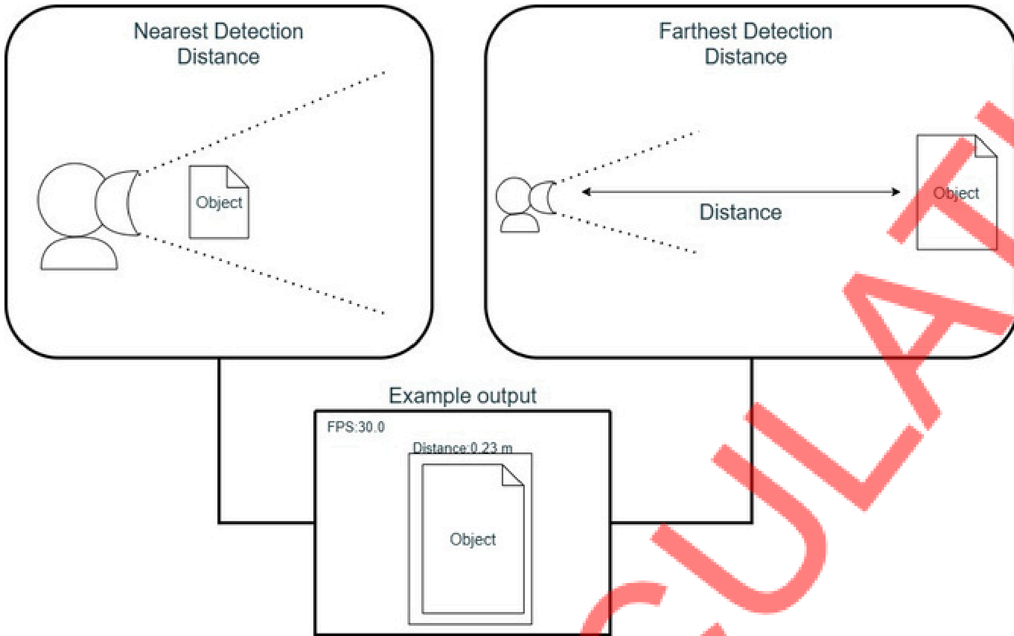


Figure 21. Testing Observation Procedure

2. Observation for detection and classification of the model:

The researchers used a webcam, as stated in the hardware specification, for real-time detection and classification of the model. The model has eight categories for classifying recyclable and non-recyclable paper, which are 'Paper_Recyclable,' 'Paper_Non-Recyclable,' 'News Paper_Recyclable,' News Paper_Non-Recyclable,' Assorted_Recyclable,' Assorted_Non-Recyclable,' 'Cardboard_Recyclable,' and 'Cardboard_Non-Recyclable. '

There were three (3) scenario tests, which are as follows:

1. Scenario A: One paper object will be detected and classified at a time; the object should be still and not moving for proper detection and classification
2. Scenario B: 2 or more objects will be detected and classified simultaneously; the object should be still and not moving for proper detection and classification.

3. Scenario C: 2 or more objects will be detected and classified at a time, but with random other stuff for proper detection of paper only.

The following is a step-by-step process of the Real-Time Test:

1. First, the researchers prepared objects within the scope of the YOLOv8 model's categories and objects that are not within the scope of the YOLOv8 model's categories. The researchers used OpenCV to prepare the
2. real-time detection and classification of the YOLOv8 model using the webcam. The researchers also implemented a process of saving the whole
3. frame of detected and classified images using OpenCV. The process of saving is where the researchers' prepared folders and subfolders to save
4. the images of different paper objects.

Evaluation Procedure

Initial Train and Testing Evaluation Procedure

This part shows and discusses the process of evaluating the model that was created in this study using the data collected and with the use of YOLOv8. The evaluations conducted on the model determined whether the model achieved high accuracy and precision in detecting whether the paper is recyclable. The real-time test was shown and discussed in this part to detect and classify the YOLOv8 model, using the images saved from the real-time test to create a confusion matrix.

NOT FOR CIRCULATION

After training, the model saves performance metrics, which was part of the evaluation procedure. Several metrics are used to evaluate the performance of the YOLOv8 model: confusion matrix, mAP score, precision, recall, and F1-score (Arvio, Kusuma, and Sangadji, 2024). These metrics can comprehensively reflect the model's performance in object detection tasks (Jia et al., 2024).

The training and validation loss is a must to observe for evaluation. Part of this loss is box loss and class loss. Box loss is the error between the predicted and ground truth bounding boxes, and class loss is the error between an object's predicted and actual class. (SJÖBERG & HYBERG, 2023). Box loss decreases to minimize the difference between the predicted and ground truth box coordinates. Class loss decreases to minimize the difference between the predicted and actual class. DFL loss decreases to minimize the imbalances in class distributions and focuses on hard-to-predict objects. Losses are necessary to train the model, which in training is to improve the model, while in validation, it is to evaluate the model on unseen data and measure the model's accuracy through the validation set.

One of the metrics for evaluation is the confusion matrix, which gives the following values:

1. True Positive: cases in which the model correctly predicts the positive class
2. True Negative: cases in which the model correctly predicts the negative class
3. False Positive: cases where the model incorrectly predicts the positive class
4. False Negatives: cases where the model mistakenly predicts the negative class

For further explanation of the cases, the True Positives are the number of times the model correctly predicts and classifies the object. True Negatives are times that the model

NOT FOR CIRCULATION

correctly does not detect an object where objects are not present. False Positives are the number of times the model classified the object incorrectly or detected it even though no object existed. Lastly, False Negatives are times when the model does not detect an object even if the object is present.

To evaluate the model, metrics of performance like accuracy, Recall, F1-score, and Mean Average Precision (mAP) are required, which represent the average value of accuracy across classes. YOLOv8 has a result for mAP50, which measures at a 0.50 threshold, focusing on the ability of the model to detect correctly, and mAP50-95 is the average precision across a range of thresholds, offering a comprehensive assessment of detection performance (Ultralytics, 2024). In YOLOv8, after training the last epoch, it displays the validation results of classes, each having precision, recall, mAP50, and mAP50-95. This also includes evaluating the model and how accurately each class was predicted.

According to Torres and Austen (2024), MAP (Mean Average Precision) is a widely used metric in YOLOv8 to assess the model's precision in detecting objects across different categories. They also stated that precision measures the accuracy of optimistic predictions, and recall measures the model's ability to capture all relevant instances. At the same time, the F1-score is the mean of precision and recall, considering false positives and negatives.

Precision is the ratio of actual optimistic predictions to total positive predictions made by the model. It displays the fraction of accurately predicted positive instances. However, recall is the ratio of true positive cases to true positive projections. This metric

NOT FOR CIRCULATION

assesses the model's ability to find each relevant occurrence in the dataset. The F1-score is a single metric that balances accuracy and recall by taking the harmonic mean of the two. It is beneficial when the distribution of classes is not consistent.

According to Rohit Kundu (2022), the F1 score serves as an alternative machine learning assessment statistic, focusing on class-specific performance instead of overall performance, as accuracy does. The F1 score, which combines two conflicting metrics—a model's accuracy and recall scores—has gained popularity in recent literature.

The F1 score is an important statistic for assessing the effectiveness of detection and classification models, particularly in the context of unbalanced datasets. It represents the harmonic mean of accuracy and recall, two additional important measures. Precision is calculated as the ratio of genuine positive forecasts to the total number of positive predictions generated by the model, demonstrating the accuracy of the predictions. Recall is the ratio of true positive predictions to total positive occurrences, demonstrating the model's capacity to detect all relevant cases. Combining precision and recall, the F1 score provides a single metric that balances the trade-off between these aspects, offering a comprehensive view of the model's performance. A high F1 score indicates the model has high precision and recall, making it useful for properly recognizing positive examples while limiting false positives.

In the study of Yaohui Hou. (2022), they also utilize precision, recall, and mean average precision (mAP) to calculate the analysis of the results of their model, which is the categorization of various solid wastes using YOLO. These performance metrics served as the evaluation procedure for the model's results.

NOT FOR CIRCULATION

Real-Time Test Evaluation Procedure

The researchers' real-time test recorded the paper objects that the model detected and classified, which were used by the researchers to plot a binary matrix to be used for evaluation of the detected object and a multi-class confusion matrix for extracting values to be applied to the multi-label confusion matrix for each classified object. A confusion matrix is utilized to visualize the results of a classification model (Singh et al., 2021). The binary, multi-class, and multi-label matrix has the following values:

1. True Positive (TP): the number of times the model correctly detected and classified a positive sample as positive.
2. False Negative (FN): the number of times the model incorrectly detected and classified a positive sample as negative.
3. True Negative (TN): the number of times the model correctly detected and classified a negative sample as negative.
4. False Positive (FP): the number of times the model incorrectly detected and classified a negative sample as positive.

Detection of YOLOv8 Model Evaluation Procedure

Table 6 shows the confusion matrix for object detection of the YOLOv8 model. Once the matrix received its values, the Accuracy, False Positive Paper Detection Rate, and False Negative Paper Detection Rate were computed.

NOT FOR CIRCULATION

Table 6. FNPDR and Accuracy

		Predicted Values	
		Predicted Detected Paper (+)	Predicted Not Detected Paper (-)
Actual Values	Actual Detected Paper (+)	True Positive (TP)	False Negative (FN)
	Actual Not Detected Paper (-)	False Positive (FP)	True Negative (TN)

False Negative Paper Detection Rate (FNPDR) is the number of times where the model missed the detection of a paper object as Not Detected Paper (FN) divided by the total number of Actual Detected Paper (FN + TP) (Grother et al. (2019); Wang & Deng (2021); Cheng et al. (2018)).

$$FNPDR(N, C) = \frac{FN}{FN + TP}$$

Where N is the number of Paper objects, and C is the confidence set by the researchers.

Detection and Classification of YOLOv8 Model

In the study of Markoulidakis, John & Rallis, Ioannis & Georgoulas, Ioannis. et al. (2021), which shows the multi-class confusion matrix method, highlighted that the metrics used for binary classification could not be applied to their full extent in the multi-class confusion matrix. They also show that the multi-class confusion matrix (see Table 7) has a different dimension, NxN, where N is the number of different class names, C0, C1, ..., CN. Their study also shows that using TP, TN, FP, and FN is not feasible in a multi-class confusion matrix. Alternatively, using the characterization shown in Figure 19 as an example, it is possible to do an analysis that focuses on a particular class. This method can

NOT FOR CIRCULATION

define a collection of metrics for every class. It is thus possible to produce measurements for the complete confusion matrix based on the appropriate mix of these metrics. The metrics specified for a multi-class confusion matrix are summarized in Table 8, with special attention paid to accuracy, recall, precision, and F1-score.

Table 7. Multi-class classification confusion matrix (Markoulidakis, John & Rallis, Ioannis & Georgoulas; Ioannis et al., 2021)

		Predicted Class			
		$C1$	$C2$	...	CN
Actual Class	$C1$	$C1,1$	FP	...	$C1,N$
		FN	TP	...	FN
	$C2$	...	...	...	...
	CN	$CN,1$	FP	...	CN,N
	...				

The researchers used the one vs. rest confusion matrix that retains the familiar structure of a multi-class confusion matrix and satisfies the requirements of a 2-dimensional confusion matrix for a multi-label confusion matrix (MLCM), which was labeled as Cn (C is the class name and n is the number of classes) for each matrix. Heydarian, Mohammadreza & Doyle, Thomas, and Samavi, Reza proposed this method. (2022). The one vs. rest confusion matrix considers all possible combinations of true and predicted classes that result in accurately extracting FN, FP, TP, and TN from the multi- class confusion matrix, which helped the metrics specified for a multi-class confusion matrix that are summarized in Table 8 be applied to the MLCM. With special attention paid to accuracy, recall, precision, and F1-score, the researchers can produce a measurement to evaluate each classification's accuracy, precision, recall, and F1-score.

NOT FOR CIRCULATION

$$(a)\ C0 \quad (b)\ C1 \quad (c)\ C2 \quad (d)\ Sum$$

$$\begin{bmatrix} 2 & 0 \\ 2 & 5 \end{bmatrix} \quad \begin{bmatrix} 1 & 3 \\ 3 & 2 \end{bmatrix} \quad \begin{bmatrix} 2 & 6 \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 5 & 9 \\ 5 & 8 \end{bmatrix}$$

Figure 22. Multi-Label Confusion Matrix

Table 8. Performance metrics for a multi-class confusion matrix (Markoulidakis, John & Rallis, Ioannis & Georgoulas; Ioannis et al., 2021)

Metric	Formula
Accuracy	$Acc(reduced) = \frac{\sum_{i}^{N} TP(C_i)}{\sum_{i}^{N} \sum_{j}^{N} C_{i,j}}$
Recall of Class *Ci*	$TPR(C_i) = \frac{TP(C_i)}{TP(C_i) + FP(C_i)}$
Precision Of Class *Ci*	$PPV(C_i) = \frac{TP(C_i)}{TP(C_i) + FP(C_i)}$
*F*1-Score of Class *Ci*	$F1(C1) = 2 \frac{TPR(C_i) \cdot PPV(C_i)}{TPR(C_i) + PPV(C_i)}$

Accuracy measures the overall correctness of the model by calculating the ratio of correctly predicted instances to the total number of instances. It quickly assesses how well the model detects paper objects among the other classes. The accuracy formulas below can be used to evaluate the model's overall accuracy and each class's accuracy within a multi- class confusion matrix. The researchers of this study included the accuracy formula for each class using the Adjabi et al. (2020) accuracy formula to evaluate the MMLM and FNPDR for a better understanding of each class's accuracy.

$$Class\ Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Overall\ Accuracy = \frac{\sum TP}{Total\ number\ of\ samples}$$

NOT FOR CIRCULATION

Chapter 4

RESULTS AND DISCUSSION

Project Description

This project aims to develop a model to detect and classify paper as recyclable. The model worked with various formats, including image files and videos, and with real-time detection through a camera. It also utilized YOLO's computer vision techniques to successfully detect and classify objects of interest.

Features:

1. Object Detection: The software is capable of detecting objects in images using You Only Look Once (YOLO) version 8. It can detect objects of interest, such as white paper, newspapers, books, magazines, brochures, calendars, and other predefined objects, and provides bounding box coordinates around the detected objects.
2. Object Classification: The software can classify the detected objects, whether they are recyclable or not, depending on the scope of the research. It also utilized YOLOv8 to recognize and classify objects based on visual features.

3. Data Augmentation: The images in the dataset to train the software use different augmentations to further train the model in different conditions, such as flip, rotation, brightness, exposure, and noise. It trains the model to be insensitive to subject and camera orientations and resilient to camera lighting, roll, and artifacts.

NOT FOR CIRCULATION

4. Real-time Processing: The software was designed to process images in real-time, making it suitable for applications that require fast and responsive object recognition. YOLOv8 is known for its speedy object detection and classification, which makes it suitable for real-time processing.

Project Structure

The flowchart presented in Figure 23 shows how the model was used to predict an image/video or webcam application. Predicting an image or video involves loading the trained model and inputting the image or video file directory for the model to find the file. The model goes through its trained YOLOv8 algorithm and then detects if the object in the image or video is present. If the object is present, then the model was put the bounding box and, at the same time, label its predicted class. The model only predicts once when inputting an image, as it is just one frame.

In contrast, when inputting a video, it takes time to process, depending on the length of the video, because it predicts every frame in the video. The YOLO library has a save function that saves the result of the predicted image or video, but there is no save function for the webcam application. To access the webcam and test the model, import the OpenCV library. The model predicts the same way in video on the webcam; it predicts every frame per second. To save the result using a webcam, use OpenCV's video writer class to record every frame.

NOT FOR CIRCULATION

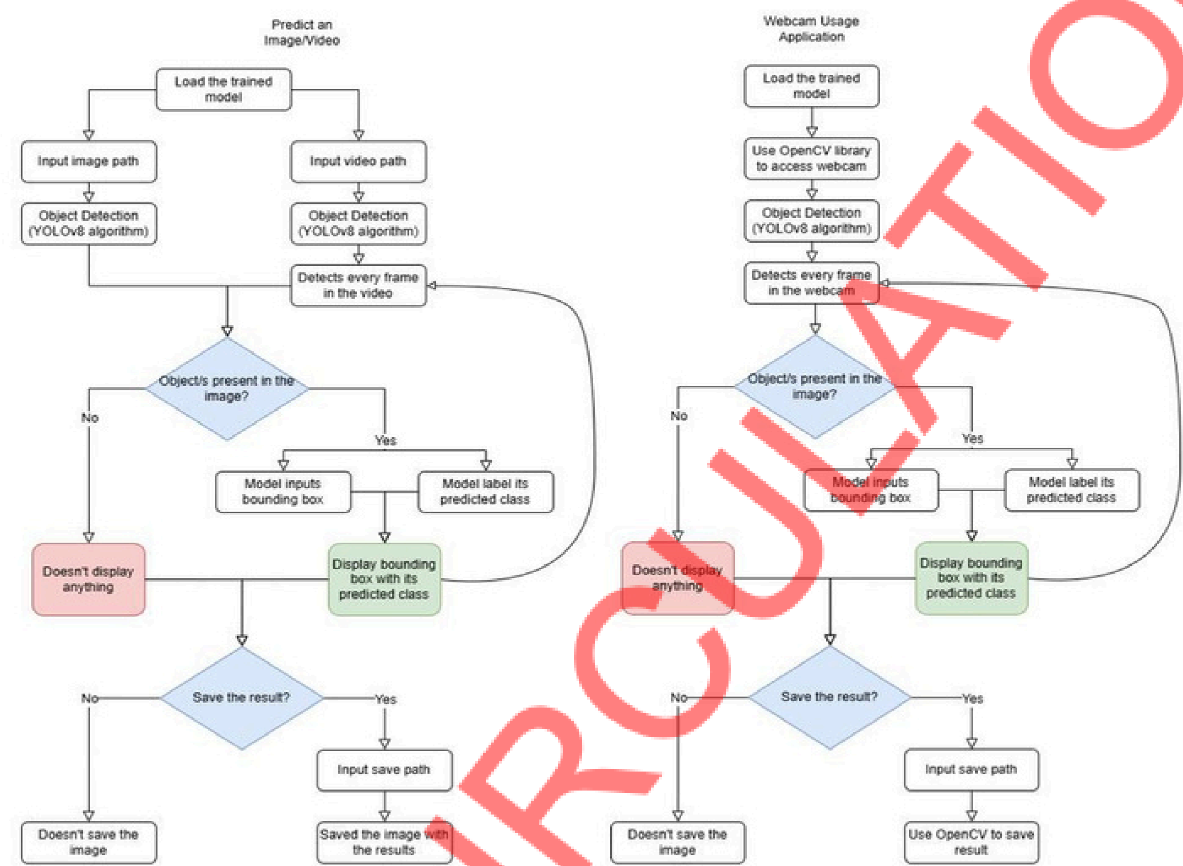


Figure 23. Flowchart Of Model's Detection Process

Project Capabilities and Limitations

This project aims to develop a model that can classify recyclable and non-recyclable materials based on image recognition techniques. By leveraging cutting-edge deep learning models, the project can identify various types of waste, such as paper, newspaper, assorted paper, and corrugated cardboard, aiding in the sorting process essential for effective waste management. The model's capability to classify materials can enhance recycling efforts, making waste processing more efficient and reducing waste sent to landfills. However, the project also faces limitations. The classification accuracy heavily depends on the quality of the training data, meaning that a diverse dataset representing all

NOT FOR CIRCULATION

relevant materials is crucial. Despite these challenges, this project has the potential to support significant sustainability efforts by promoting more effective waste sorting and recycling practices.

Capabilities of the Model

1. The model can detect paper objects based on the input image; multiple paper objects in an input image can also be detected.
2. The model can classify if the detected object is recyclable or non-recyclable; multiple paper objects can also be classified as recyclable or non-recyclable.
3. The model displays a bounding box and class label around the paper object.
4. The model can predict input images and videos with various supported formats.
5. The model can predict in real time using a webcam.

Limitations of the Model

1. The model cannot detect far or small objects.
2. The model cannot detect various conditions not included in the dataset.
3. The model returns the detected image in 640 x 640 resolution.
4. When testing the model on a webcam, the only size is 640 x 480, which is the default size of OpenCV.

Project Evaluation

This section discusses the training, validation, and test procedure results of the proposed YOLOv8 object detection and classification model.

NOT FOR CIRCULATION

YOLOv8 Model Development

The researchers set these hyperparameters to fully optimize the training time and limitations of this study. The data hyperparameter is changed to set the exported dataset for training the model. The batch is maximized to complement the memory usage capacity of the GPU set for this study. An epoch of 150 sufficed to train the model to accommodate the dataset. Patience is set to check for improvement and stop the training early to prevent overfitting after 20 epochs. Speeding up training, possibly at the cost of accuracy, is not considered in this study, so amp is false.

Table 9. Hyperparameters that were set for the YOLOv8 Model

Hyperparameter	Value
data	Dataset directory/data.yaml
batch	16
epochs	150
patience	20
amp	False

It took 141 epochs to train the YOLOv8 model before stopping early in JupyterLab with an NVIDIA GeForce RTX 4060 Ti GPU. The overall training time of 141 epochs is 16 hours, 15 minutes, and 23 seconds; the average training time per epoch is approximately 5 minutes.

NOT FOR CIRCULATION

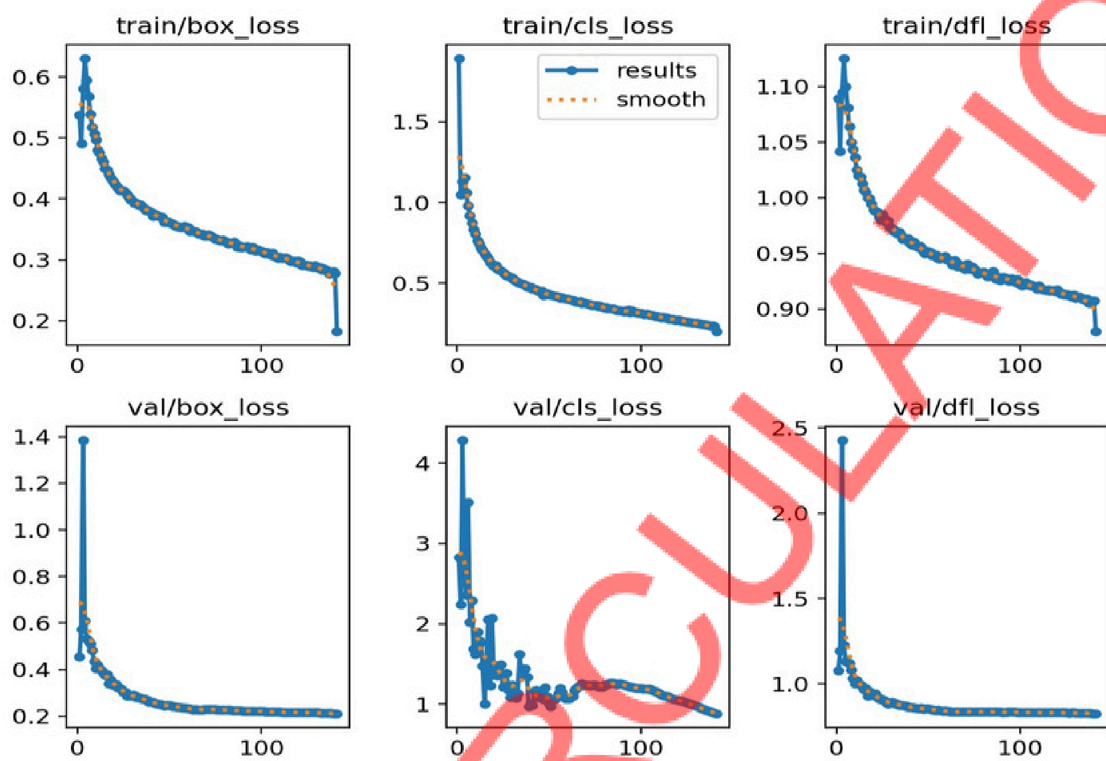


Figure 24. Training and Validation Loss Curves

Figure 24 shows that the losses decrease per epoch. Throughout 141 epochs, both training and validation losses, which are box, cls, and DFL, show a continuously lower altitude, which indicates that the model is learning smoothly from the training set and generalizing to the validation set. The train/box_loss decreases from around 0.6 to 0.1, which shows that the model was improved to put bounding boxes accurately. The train/cls_loss decreases from around 1.8 to a massive drop of 0.2, which indicates the model's aptitude to classify the papers with better accuracy. The train/dfl loss goes from around 1.12 to 0.84, which increases the model's confidence regarding its detections. The val/box_loss decreases from 1.4 to 0.2, meaning the model improved by bounding boxes to unseen data. The val/cls_loss, which has minor spikes of increasing and decreasing, still manages to decrease from around 4 to 0.9, meaning that the model can still inaccurately

NOT FOR CIRCULATION

classify the detected object. The val/dfl loss starts from around 1.14 and decreases to 0.6, which the model improved to predict hard-to-detect samples.

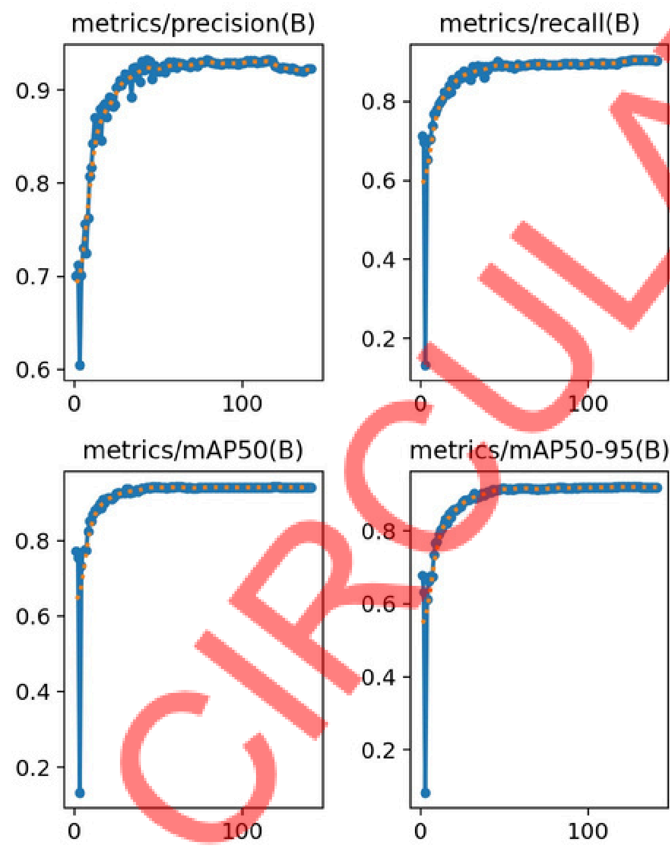


Figure 25. Performance Result of Training

Figure 25 shows the measurement of how accurate the model had become. The model achieved 92.4% in precision, which is the accuracy of how many detections were correct. In recall, the model achieved 90.2% accuracy in predicting all objects in an image. The mAP50 that the model achieved is 94.2%, and the mAP50-95 is 92.1%, which is a promising accuracy score. The researchers anticipated the accuracy would be lower given that the dataset used is self-created and contains images from multiple sources.

NOT FOR CIRCULATION

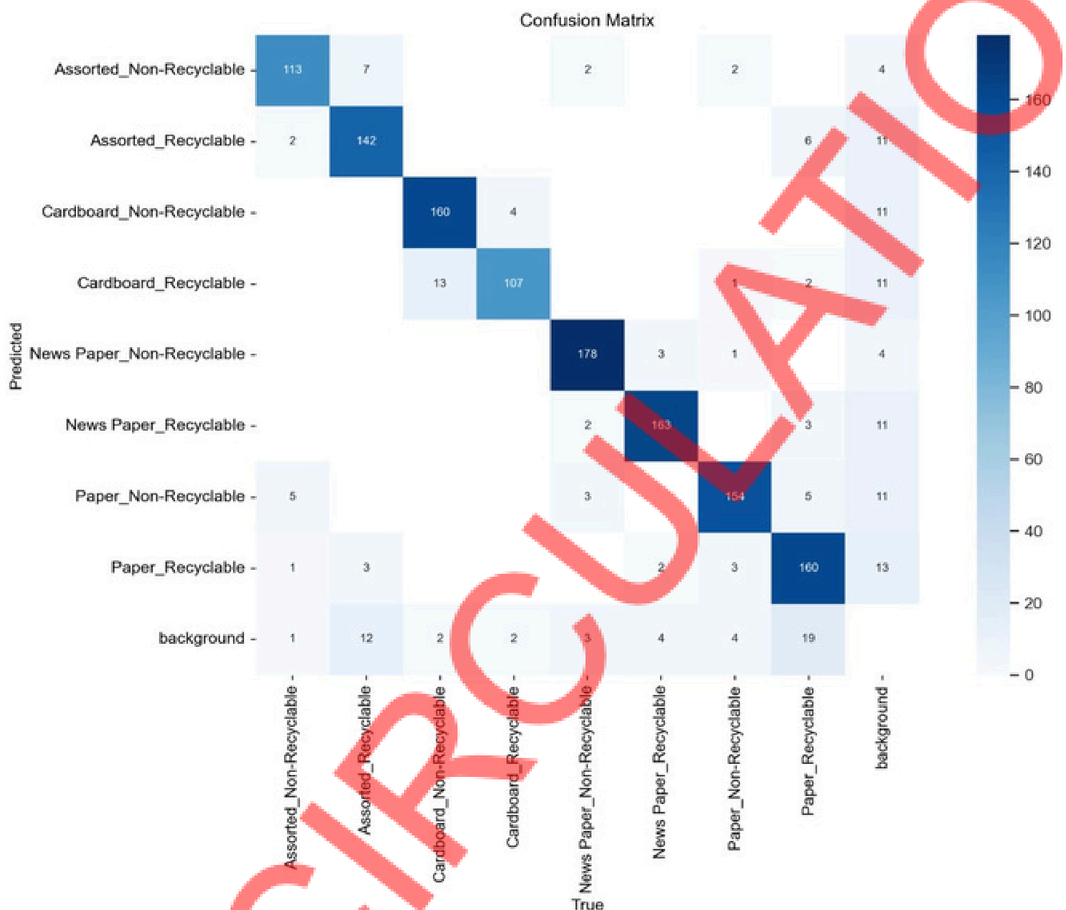


Figure 26. Confusion Matrix of Training Results

In Figure 26, two sides are predicted and true: the results of the model predictions on the validation set. The background class is the representation of false positives; based on the confusion matrix from the training and validation results of the model, the model has instances to initiate false positives that misclassified its classes, and even though there is no paper within the scope that is present, it still detects.

In the table below, the result of each class is based on the validation set. The lowest precision score is Cardboard_Recyclable, 88.2%, and the highest is Cardboard_Non- Recyclable. It may be due to an imbalanced dataset that non-recyclable cardboard is more included in the dataset. The lowest recall is Paper_Recyclable at 79%, and the highest is

NOT FOR CIRCULATION

Cardboard_Recyclable at 95.6%. The researchers observe that it may have been caused by the few instances of objects in the images in Cardboard_Recyclable, in contrast to the images in Paper_Recyclable, which have many objects in the image. The highest mAP50 of 96.8% and mAP50-95 of 96.2% is Cardboard_Non-Recyclable, possibly because it has the highest image count and the second highest annotated class in the dataset.

Table 10. Class Evaluation result

Class	Precision	Recall	mAP50	mAP50-95
Assorted_Non-Recyclable	88.4%	92.6%	94.1%	89.3%
Assorted_Recyclable	90.2%	81.7%	87.9%	85.2%
Cardboard_Non-Recyclable	98.2%	92.8%	96.8%	96.2%
Cardboard_Recyclable	88.2%	95.6%	96.6%	95.8%
News Paper_Non-Recyclable	97.8%	93.8%	95.6%	94.5%
News Paper_Recyclable	94.4%	93%	96.7%	92.7%
Paper_Non-Recyclable	90%	93.3%	95.9%	94.9%
Paper_Recyclable	92.4%	79%	89.8%	88%

Preliminary Test Evaluation Result

After the model predicts the set of images, the researchers manually review each predicted image and make a confusion matrix. Figure 27 shows how many correct detections and classifications the model predicted. The background is the determinant of the True Negative value, which the model should not detect without an object that is not part of any class.

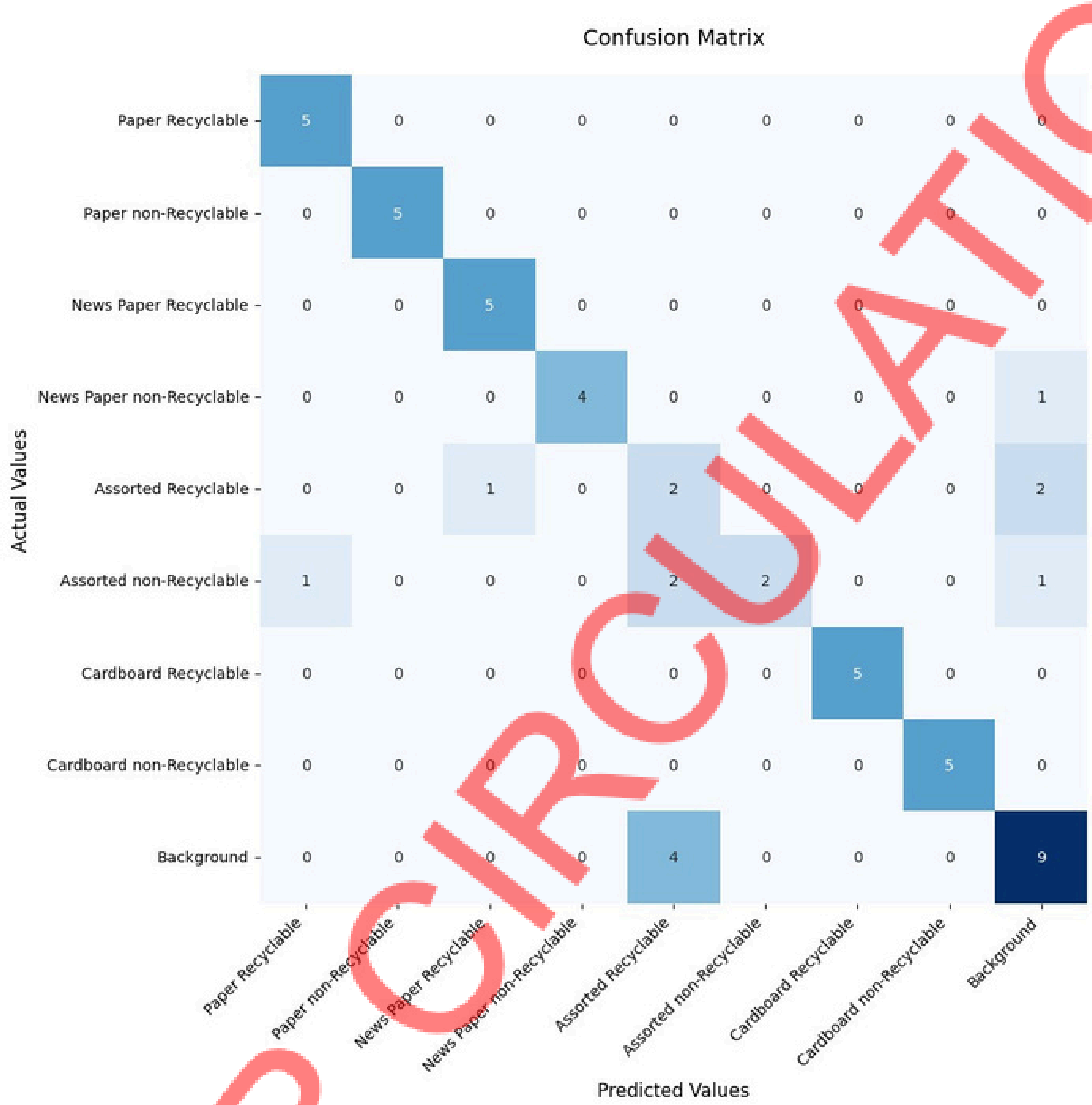


Figure 27. Confusion Matrix in Tested Images

Table 11 shows the confusion matrix values put into a table. The model falsely detected four objects in background images; all were misclassified as Assorted Recyclable. This explains why there were four total predictions in the background. Notably, a single background image was incorrectly detected 4 times, each classified as Assorted Recyclable. Additionally, there is an instance where the model detected an Assorted non- Recyclable object twice and misclassifies it as Assorted Recyclable and Paper Recyclable, resulting in 6 counts for that row instead of the expected 5. Images that were not detected

NOT FOR CIRCULATION

even by the object were News Paper non-Recyclable, two in Assorted_Recyclable, and Assorted_non-Recyclable. Out of 50 images, 33 were correctly detected and classified, while nine background images were correctly undetected, totaling 42 correct predictions.

Table 11. Confusion Matrix Values in Tested Images

Class	Total Predicted	True Positives	False Positives	False Negatives	True Negatives
Paper Recyclable	6	5	1	0	44
Paper non_Recyclable	5	5	0	0	45
News Paper Recyclable	6	5	1	0	44
News Paper non_Recyclable	4	4	0	1	45
Assorted Recyclable	8	2	6	3	39
Assorted non_Recyclable	2	2	0	4	44
Cardboard Recyclable	5	5	0	0	45
Cardboard non_Recyclable	5	5	0	0	45
Background	4	9	4	4	33
Total	45	42	12	9	384

Using the formulas for computing the metrics, the results were manually computed and came with the class score results for the test image set. Table 12 shows that the model struggles to detect and classify certain classes correctly, especially Assorted Recyclable and Assorted non-Recyclable, often confusing them with other classes or even detecting non-paper objects. Overall, the total accuracy of the model on the testing image set is 84%.

NOT FOR CIRCULATION

Table 12. Class Evaluation Results in Tested Images

Class	Accuracy	Precision	Recall	F1-score	Support
Paper Recyclable	98%	83.3%	100%	90.9%	5
Paper non-Recyclable	100%	100%	100%	100%	5
News Paper Recyclable	98%	83.3%	100%	90.9%	5
News Paper non-Recyclable	98%	100%	80%	88.89%	5
Assorted Recyclable	82%	25%	40%	30.8%	5
Assorted non-Recyclable	92%	100%	33.3%	50%	5
Cardboard Recyclable	100%	100%	100%	100%	5
Cardboard non-Recyclable	100%	100%	100%	100%	5
Overall Accuracy			84%		

Real-Time Model Evaluation

Based on the researchers' observation, the sizes of each paper object, like torn papers and crumpled papers, are detected by the model with a distance of 0.45m; poor lighting, unusual object shapes, and partially visible items can impact detection accuracy. The model was tested in different lighting conditions and poorly detected in dark surroundings or dim lights. Hence, the researchers decided to use a well-lit room for the real-time testing of the model. The researchers set up the distance for real-time detection and classification as 0.80m to 1m for the (3) three types of paper, and cardboard is detecte at the distance of 1.30m, which shows that our model is capable of detection with a proper distance with the object and the distance of the paper object to be detected varies on the size of the paper object. For Scenario A, the distances were adjusted in the three types of paper and cardboard. For Scenario B and C, the distance was set to 1m to cover multiple objects in the testing environment. Using OpenCV libraries, the minimum and maximum

NOT FOR CIRCULATION

frames per second (FPS) were determined, with 18 fps being the lowest FPS and 38 fps being the highest reached FPS while testing in real-time.

Scenario Test Result of the Model

Paper Object Detection

The researchers observed the paper object detection performance of the YOLOv8 model across the different scenarios that the researchers implemented. For Scenario A, 60 paper objects were tested and observed. Table 13 presents the number of times the paper object was detected by its respective class. It detected 51 out of 60 paper objects. In Scenario B, 55 paper objects were tested, and the model detected 47. Lastly, in Scenario C, 55 paper objects were alongside seven non-paper objects, with the model detecting 39 of them and correctly leaving all seven non-paper objects undetected. It proceeded to compute the FNPDR and accuracy in each scenario.

Table 13. Paper Object Detected in Scenario A, B, and C

Detected Objects as (Paper)	Scenario A	Scenario B	Scenario C
Paper Recyclable	8 6 8	8/9	8/11
Paper non-Recyclable	9 10	6/8 8	8/9
News Paper Recyclable	2/8	11 12	7/9
News Paper non-Recyclable	4/6	2/7 -	7/9
Assorted Recyclable	4/5	-	5/9
Assorted non-Recyclable			4/8 -
Cardboard Recyclable	51/6	47/5	- 7
Cardboard non-Recyclable			46/6
Not Paper	0	5	2
Total			

Scenario A.

Table 14 shows that not all the paper objects (85%) were detected in the first scenario, resulting in 15% of the FNPDR. The model does not detect 15% of the paper objects.

NOT FOR CIRCULATION

Table 14. FNPDR and Accuracy of YOLOv8 Model on Scenario A

		Predicted Values	
		Predicted Detected Paper (+)	Predicted Not Detected Paper (-)
Actual Values	Actual Detected Paper (+)	51	9
	Actual Not Detected Paper (-)	0	0

The computation to calculate the FNPDR and Accuracy of the YOLOv8 Model is given below:

FNPDR (60, 0.50) = FN/(FN + TP) = 9/ 9+51 = 9/60 = 0.15 or 15%

Accuracy = 51/60 = 0.85 or 85%

This scenario showcases that the YOLOv8 model can perform detection by a single object per detection.

Scenario B.

Table 15 shows that eight paper objects were not detected in the first scenario B, resulting in 14.54% of the FNPDR and an accuracy of 85.45%.

NOT FOR CIRCULATION

Table 15. FNPDR and Accuracy of YOLOv8 Model on Scenario B

		Predicted Values	
		Predicted Detected Paper (+)	Predicted Not Detected Paper (-)
Actual Values	Actual Detected Paper (+)	47	8
	Actual Not Detected Paper (-)	0	0

The computation to calculate the FNPDR and Accuracy of the YOLOv8 Model is given below:

FNPDR (55, 0.50) = FN/(FN + TP) = 8 / 8 + 47 = 8/55 = 0.1454 or 14.54%

Accuracy = 47/55= 0.8545 or 85.45%

The paper object maintained an accuracy of 85.45% detection during scenario B, where multiple paper objects were detected at a time. The researchers also observed that if the paper objects are contiguous to each other, the model detects the paper objects as one. However, if the paper objects are just near each other and not contiguous, they are being detected individually. The researchers also found out that the paper object detection distance varies with the size of the paper object. However, the paper object can still be detected without a maximum number of objects placed in the table if the paper objects can still fit in the table without being contiguous.

Scenario C.

Table 16 shows that 16 paper objects are not detected during scenario C, resulting in 29.09% FNPDR and an accuracy of 74.19% for the scenario of multiple paper objects detection and classification with non-paper objects.

NOT FOR CIRCULATION

Table 16. FNPDR and Accuracy of YOLOv8 Model on Scenario C

		Predicted Values	
		Predicted Detected Paper (+)	Predicted Not Detected Paper (-)
Actual Values	Actual Detected Paper (+)	39	16
	Actual Not Detected Paper (-)	0	7

The computation to calculate the FNPDR and Accuracy of the YOLOv8 Model is given below:

FNPDR (62, 0.50) = FN/(FN + TP) = 16/16 + 39 = 16/55 = 0.2909 or 29.09%

Accuracy = TP + TN / TP + FP + FN + TN = 39 + 7/39 + 0 + 16 + 7 = 46/62 = 0.7419 or 74.19%

The researchers observed that the accuracy of the paper object to detect is only 74.19% when the paper object is near a non-paper object, which shows that scenarios A and B have a maintaining accuracy of 85% when detecting paper objects with a distance of 0.80 m to 1 m for paper, newspapers, and assorted categories, and 1.30 m for corrugated boxes because the box should fit in the frame. Researchers also observed that corrugated cardboard cannot be included in scenarios A and B because the corrugated cardboard can only fit one (1) in the frame.

Paper Object Detection and Classification of the YOLOv8 Model

In this part, using the same YOLOv8 model, the researchers show the following table evaluation of the classified detected paper objects of each scenario created, shown in Tables 18, 20, and 22. The detected paper object was classified and tested in different and

NOT FOR CIRCULATION

random positions. Figures 28, 30, and 32 show the imbalances of each class of the multi-class confusion matrix (MCCM), which is why the researchers decided to use the multi-label confusion matrix (MLCM) for the once vs. rest evaluation, which evaluated each class's accuracy, precision, recall, and F1-score.

Scenario A.

The researchers of this study provided a total of 89 paper objects, as shown in Figure 28, for single-paper object detection and classification. The multi-class confusion matrix of the model consists of 8 classes.

NOT FOR CIRCULATION

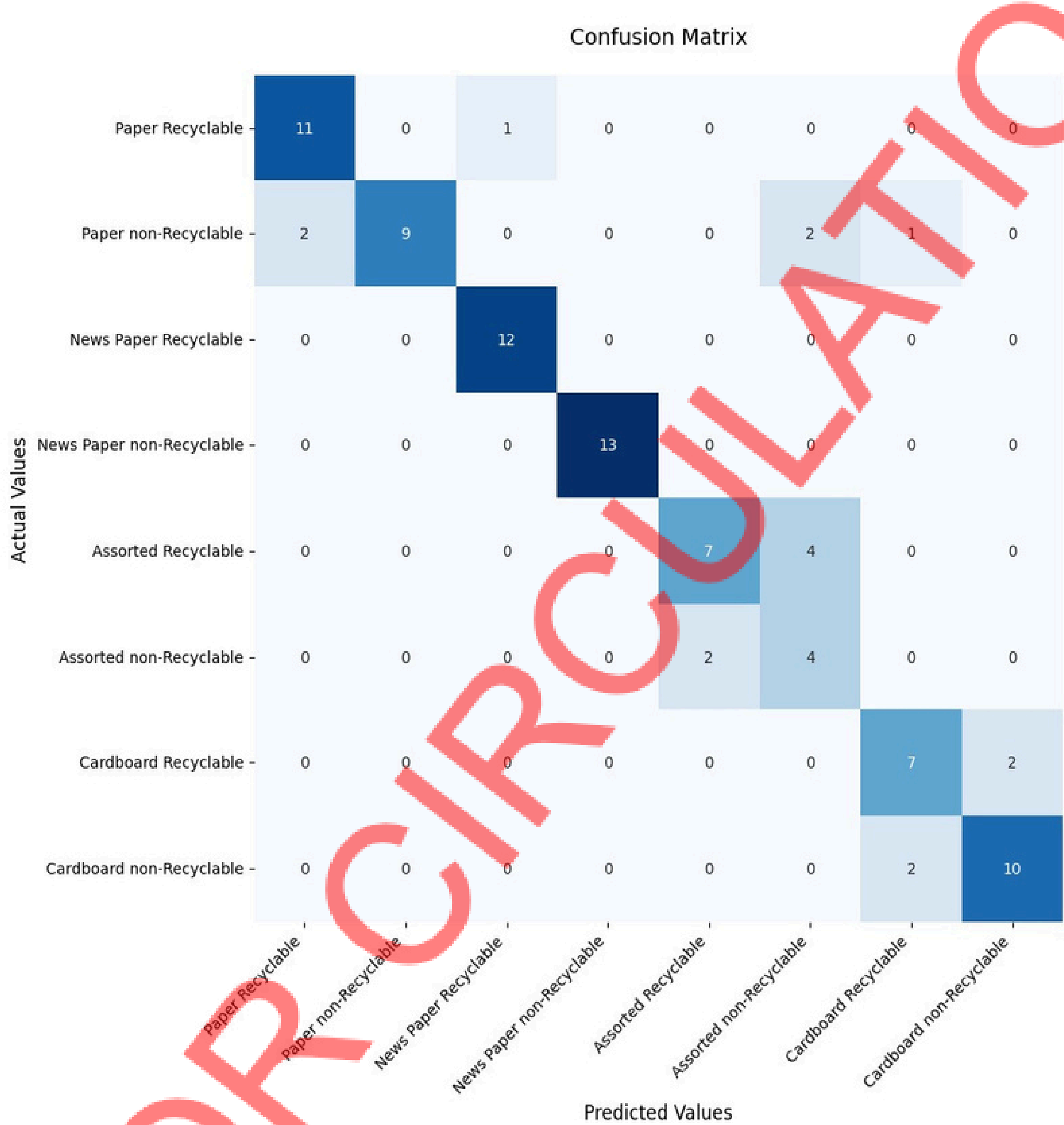


Figure 28. MCCM in Detection and Classification of Scenario A.

The researchers extracted the TP, TN, FP, and FN for the multi-label classification matrix shown in Figure 29 to evaluate accuracy, precision, recall, F1-score, and overall accuracy.

NOT FOR CIRCULATION

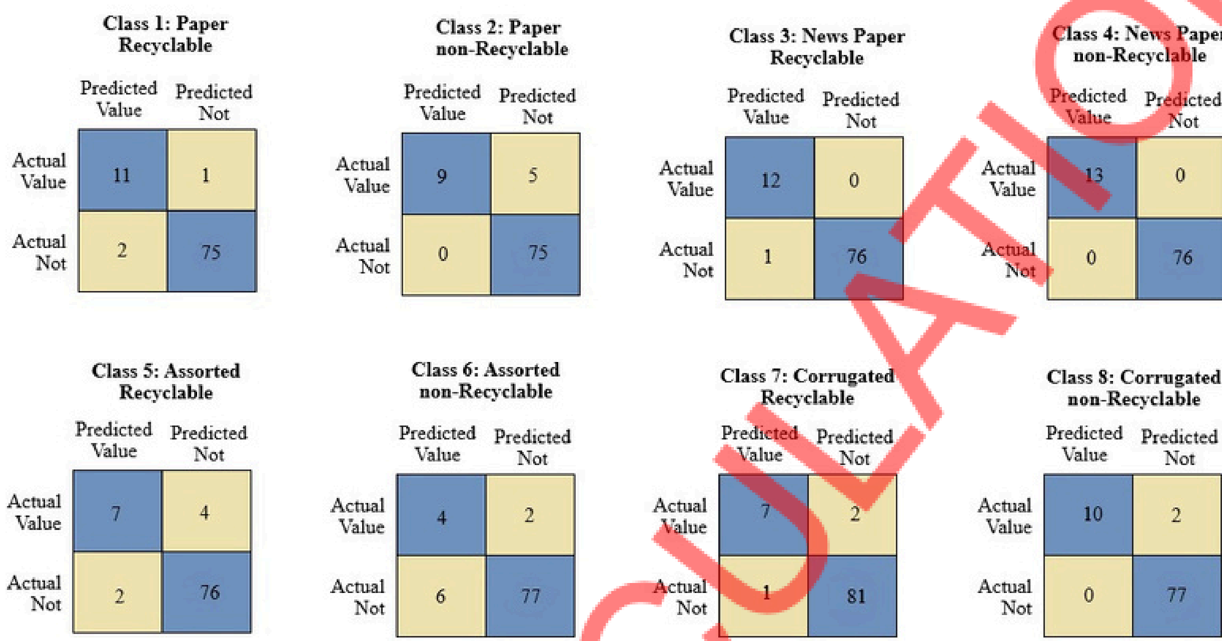


Figure 29. MLCM in Detection and Classification of Scenario A.

Table 17 shows the confusion matrix values of Scenario A. put into a table. The model misclassified an actual class as another class and another class as that class. Of 89 paper objects, 73 were correctly detected and classified, while 16 were misclassified.

Table 17. Confusion Matrix Values in Scenario A.

Class	Total Predicted	True Positives	False Positives	False Negatives	True Negatives
Paper Recyclable	13	11	2	1	75
Paper non_Recyclable	9	9	0	5	75
News Paper Recyclable	13	12	1	0	76
News Paper non_Recyclable	13	13	0	0	76
Assorted Recyclable	9	7	2	4	76
Assorted non_Recyclable	10	4	6	2	77
Cardboard Recyclable	10	7	3	2	77
Cardboard non_Recyclable	12	10	2	2	75
Total	89	73	16	16	606

NOT FOR CIRCULATION

Table 18 shows the real-time evaluation result of the YOLOv8 model for paper classification in scenario A across eight categories, which shows varied performance metrics, reflecting strengths and areas for improvement. The evaluation metrics for the YOLOv8 model in classifying recyclable and non-recyclable paper categories in scenario A are presented. The News Paper non-Recyclable achieved 100% in all metrics. At the same time, other classes were misclassified, particularly those with below 95% accuracy and subpar metric scores: Assorted Recyclable, Assorted non-Recyclable, Paper non-Recyclable, and Cardboard Recyclable. The lowest performing class was Assorted non-Recyclable, indicating that the model struggles to classify this category correctly. The model's overall accuracy in Scenario A was 82.02%, as it misclassified 16 objects out of 89.

Table 18. Class Evaluation Results in Scenario A.

Class	Accuracy	Precision	Recall	F1-score
Paper Recyclable	96.63%	84.62%	91.67%	88%
Paper non-Recyclable	94.38%	100%	64.29%	78.26%
News Paper Recyclable	98.88%	92.31%	100%	96%
News Paper non-Recyclable	100%	100%	100%	100%
Assorted Recyclable	93.26%	77.78%	63.64%	70.0%
Assorted non-Recyclable	91.01%	40%	66.67%	50%
Cardboard Recyclable	94.38%	70%	77.78%	73.68%
Cardboard non-Recyclable	95.51%	83.33%	83.33%	83.33%
Overall Accuracy		82.02%		

NOT FOR CIRCULATION

Scenario B.

The researchers of this study provided a total of 92 paper objects, as shown in Figure 30, for multiple paper object detection and classification. The multi-class confusion matrix of the model consists of only six classes due to the size of the cardboard, which did not fit the distance between the camera and the testing environment.

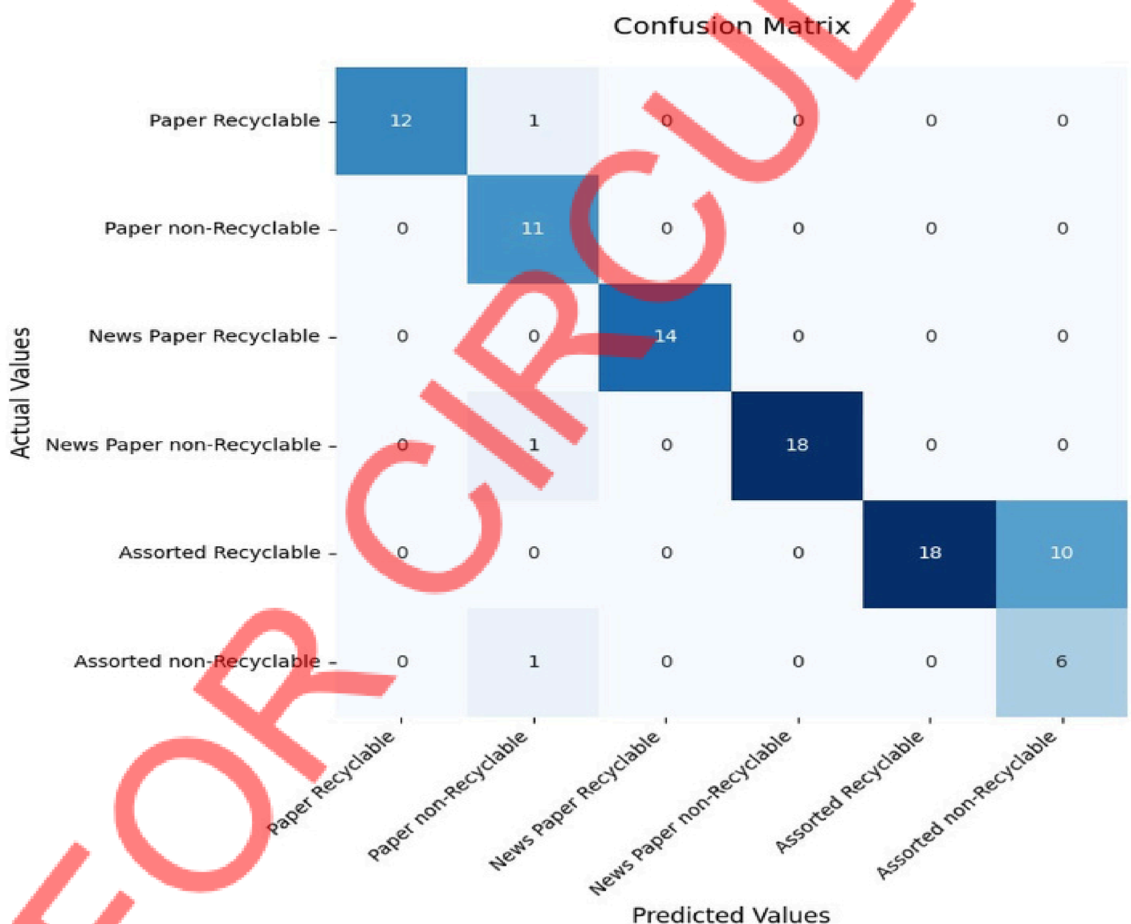


Figure 30. MCCM in Detection and Classification of Scenario B.

The researchers extracted the TP, TN, FP, and FN for the multi-label classification matrix in Figure 31 to evaluate accuracy, precision, recall, F1-score, and overall accuracy.

NOT FOR CIRCULATION

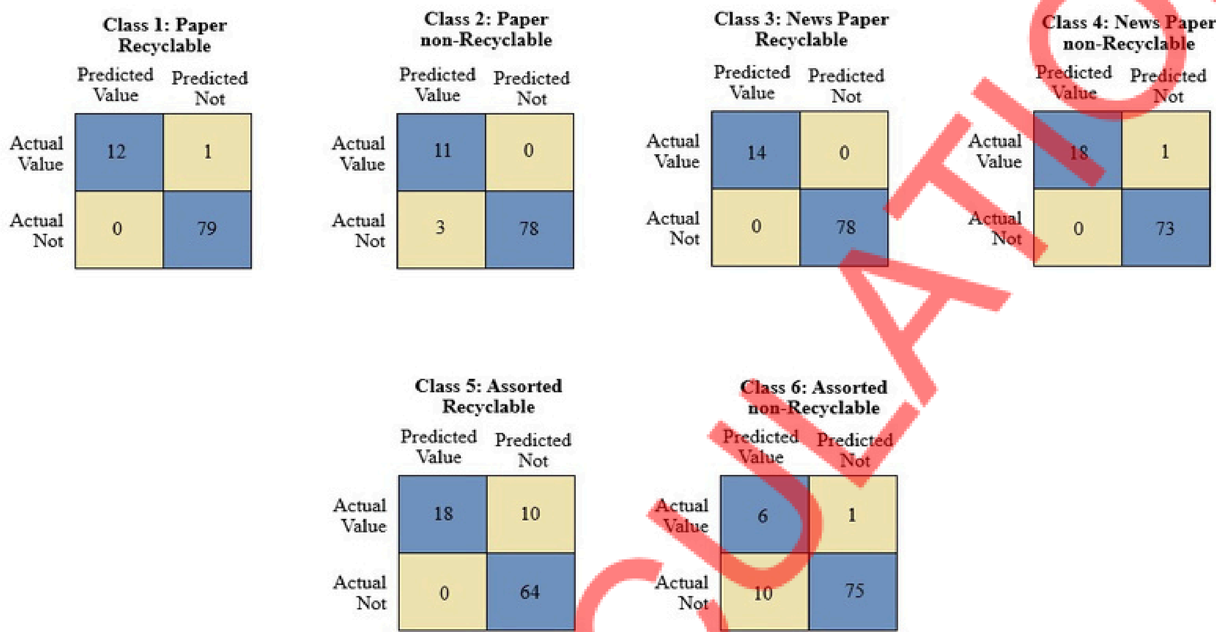


Figure 31. MLCM in Detection and Classification of Scenario B.

Table 19 shows the confusion matrix values of Scenario B. put into a table. The significant misclassification of the model was the classification of 10 Assorted Recyclable objects as Assorted non-recyclable, which shows that the model struggles in determining recyclable assorted paper. Out of 92 paper objects, 79 were correctly detected and classified, while 13 were misclassified.

Table 19. Confusion Matrix Values in Scenario B.

Class	Total Predicted	True Positives	False Positives	False Negatives	True Negatives
Paper Recyclable	12	12	0	1	79
Paper non_Recyclable	14	11	3	0	78
News Paper Recyclable	14	14	0	0	78
News Paper non_Recyclable	18	18	0	1	73
Assorted Recyclable	18	18	0	10	64
Assorted non_Recyclable	16	6	10	1	75
Total	92	79	13	13	447

NOT FOR CIRCULATION

Table 20 shows the real-time evaluation result of the YOLOv8 model for paper classification in scenario B across six categories, showing varied performance metrics and highlighting the specific class struggles. The evaluation metrics reflect the model's effectiveness when two or more paper objects are in the frame. The News Paper Recyclable class achieved 100% across all metrics, while the Assorted Recyclable and Assorted non-Recyclable classes achieved below 90% accuracy. The lowest performing class was Assorted non-Recyclable, which underperformed in Scenario A, indicating that the model struggles specifically in this class. The overall accuracy the model achieved in Scenario B was 85.87%, which is an increase in accuracy compared to Scenario A due to the exclusion of Cardboard Recyclable and Cardboard non-recyclable. The model still misclassified 13 out of 92 paper objects.

Table 20. Class Evaluation Results in Scenario B.

Class	Accuracy	Precision	Recall	F1-score
Paper Recyclable	98.91%	100%	92.3%	96%
Paper non-Recyclable	96.74%	78.57%	100%	87.5%
News Paper Recyclable	100%	100%	100%	100%
News Paper non-Recyclable	98.91%	100%	94.74%	97.3 %
Assorted Recyclable	89.13%	100%	64.29%	78.26%
Assorted non-Recyclable	88.04%	37.5%	85.71%	52.17%
Overall Accuracy		85.87%		

NOT FOR CIRCULATION

Scenario C.

The researchers of this study provided an overall of 95 detected and classified papers with few non-paper objects alongside, as shown in Figure 32, for multiple paper object detection and classification. The multi-class confusion matrix of the model consists of 6 classes.

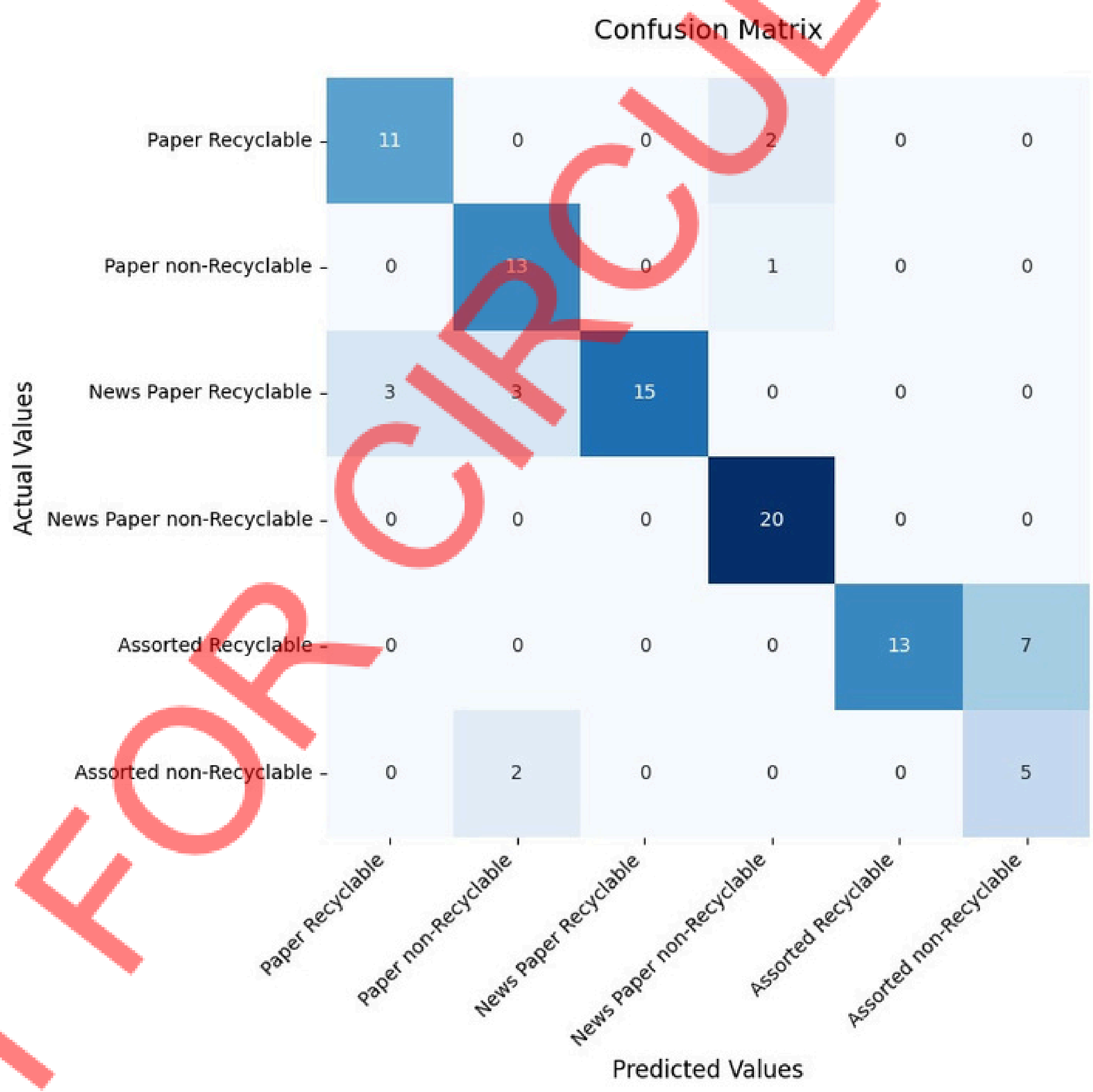


Figure 32. MCCM in Detection and Classification of Scenario C.

NOT FOR CIRCULATION

The researchers extracted the TP, TN, FP, and FN for the multi-label classification matrix shown in Figure 33. This evaluates accuracy, precision, recall, F1-score, and overall accuracy.

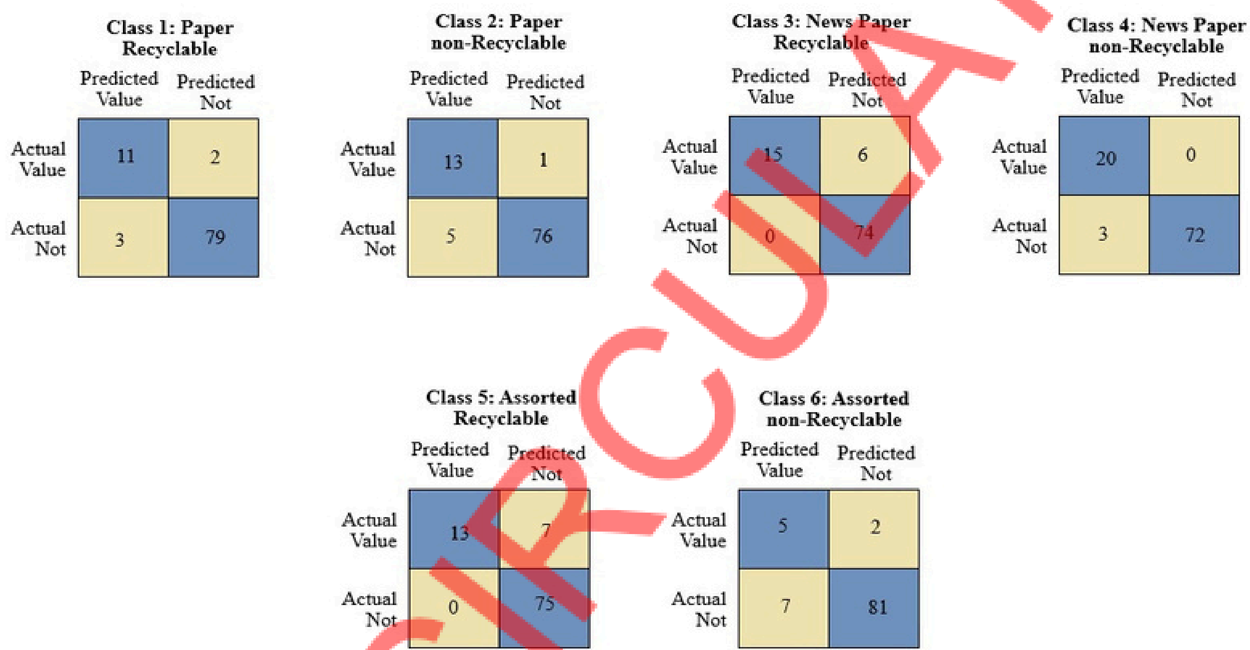


Figure 33. MLCM in Detection and Classification of Scenario C.

Table 21. Confusion Matrix Values in Scenario C.

Class	Total Predicted	True Positives	False Positives	False Negatives	True Negatives
Paper Recyclable	14	11	3	2	79
Paper non_Recyclable	18	13	5	1	76
News Paper Recyclable	15	15	0	6	74
News Paper non_Recyclable	23	20	3	0	72
Assorted Recyclable	13	13	0	7	75
Assorted non_Recyclable	12	5	7	2	81
Total	95	77	18	18	457

NOT FOR CIRCULATION

Table 21 shows the confusion matrix values of Scenario B. put into a table. Testing the model with two or more paper objects alongside a few non-paper objects tends to misclassify several classes. Out of 95 paper objects, 77 were correctly detected and classified, while 18 were misclassified.

Table 22. Class Evaluation Results in Scenario C.

Class	Accuracy	Precision	Recall	F1-score
Paper Recyclable	94.74%	78.57%	84.62%	81.43%
Paper non-Recyclable	93.68%	72.22%	92.86%	81.25%
News Paper Recyclable	93.68%	100%	71.43%	83.33%
News Paper non-Recyclable	96.84%	86.96%	100%	92.86%
Assorted Recyclable	92.63%	100%	65%	78.79%
Assorted non-Recyclable	90.53%	41.67%	71.43%	52.63%
Overall Accuracy	81.05%			

Table 22 shows the real-time evaluation result of the YOLOv8 model for paper classification in scenario C across six categories, showing varied performance metrics, and measures the model's capability of detecting and classifying two or more paper objects with non-paper objects in a frame. Neither of the classes achieved 100% accuracy. However, the highest performing class across all metrics was the News Paper non-Recyclable class, while the Assorted Recyclable class achieved 92.63% and the Assorted non-Recyclable class achieved 90.53% accuracy. The lowest performing class was Assorted non-Recyclable, which also underperformed in Scenario A and B, indicating further improvement was needed, mainly due to the complexity of this class. The overall accuracy the model achieved in Scenario C was the lowest out of the three scenarios, which is

NOT FOR CIRCULATION

81.05%, which might be due to the inclusion of non-paper objects; the model's capability to detect and classify correctly, lessens. The model still misclassified 18 out of 95 paper objects.

NOT FOR CIRCULATION

Chapter 5

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

This chapter presents the results obtained, summaries, and conclusions. This chapter concludes with the researchers' suggestions for further enhancing the model and resolving the issues raised by the project evaluation.

Summary of Findings

The YOLOv8 model can be trained even through a custom dataset, which tests the reliability and effectiveness of the dataset. Having a well-defined dataset dictates YOLOv8 accuracy in object detection and classification. Proper annotation and augmentation are key to further enhancing the dataset.

Based on the collected, examined, and evaluated data, the summary of the results is summed up as follows:

1. YOLOv8 Model Development

   The study addresses the challenge of identifying recyclable and non-recyclable paper waste in the Philippines, using YOLOv8, a state-of-the-art computer vision model, to improve waste management. The methodology section laid the foundation for this study, which included gathering pictures of recyclable and non-recyclable paper. AI-generated photos, online scraping, and self-captured photographs were all compromised in the collection. Bounding boxes and categorization labels were applied to these photos after careful annotation. Data

NOT FOR CIRCULATION

preprocessing, such as auto orientation and resizing to 640x640 pixels, was performed to standardize the dataset for model training.

Additionally, data augmentation techniques like flipping, rotation, brightness adjustment, and noise addition were applied to enhance the robustness of the training dataset. The YOLOv8 model was trained using this preprocessed dataset, with 80% of the images allocated for training and 20% for validation. The implementation leveraged YOLOv8's CNN architecture to extract and classify features from the images effectively. Metrics such as accuracy and loss were visualized to evaluate the model's performance detecting and classifying recyclable and non-recyclable paper waste.

The allocated dataset contains 6,197 images before data augmentation in total. Among these, 5,298 images cover eight classes: recyclable newspaper, paper, cardboard, assorted, and non-recyclable newspaper, paper, cardboard, and assorted. These images include papers with tape, adhesive, food residue, dyes, and pigments. Eight hundred ninety-nine background images do not contain the desired object, allowing the model to learn what to detect and what not to detect. After augmentation, the dataset increased to 14,871 images in the training set, visualized with the matplotlib library. The YOLOv8 model was trained using this preprocessed dataset, with 80% of the images (14,871) allocated for training and 20% (1,240) for validation, which shows that the dataset comprises 16,111 images after the augmentation. The implementation leveraged YOLOv8's CNN architecture to extract and classify features from the images effectively. Metrics such as accuracy and loss were visualized to evaluate the model's performance

NOT FOR CIRCULATION

detecting and classifying recyclable and non-recyclable paper waste. The process flow included meticulous manual annotation using Roboflow Annotate, a preprocessing and augmentation tool compatible with YOLOv8, making it more efficient than other annotation tools.

Model training is the deciding step in developing a computer vision model. It teaches the YOLOv8 model to detect and classify objects in a frame correctly. The annotated, preprocessed, and augmented dataset was exported using Roboflow in YAML format, which contains the configuration for training and validation images and labels. Roboflow provided a convenient download code to save the dataset to the designated directory. To train the YOLOv8 model, the Ultralytics library was installed to load the YOLOv8 framework. The torch and CUDA toolkit were installed to leverage the system's GPU for faster training instead of using the CPU. The training involved setting several key hyperparameters (data, batch, epoch, patience, and amp). The study utilized YOLOv8's default hyperparameters, with changes to the default values justified and discussed in Chapter 4. After setting the hyperparameters, the YOLOv8 model was loaded, and training commenced using the exported YAML dataset. Training time depended on hardware specifications, the number of epochs, and batch size.

2. Model Evaluation Result

The training and validation losses consistently decreased over each epoch and stopped at 141. Across 141 epochs, the training box loss decreases from around 0.6 to 0.1, classification loss from 1.8 to 0.2, and DFL loss from 1.12 to 0.84.

NOT FOR CIRCULATION

Similarly, the validation box loss decreases around 1.4 to 0.2, classification loss from 4 to 0.9, and DFL loss from 1.14 to 0.6. The consistent decrease across all losses indicates that the model indeed learns and has no immediate signs of overfitting or underfitting; however, the high initial validation class loss suggests that class imbalance occurs (e.g., some classes are underrepresented) or the model took longer to learn classifying than localizing the object.

After the model's training, the model achieved an overall 92.4% in precision, 90.2% in recall, and 92.1% in mean average precision (mAP), which is a promising metric score, indicating a good performance in object detection and classification. However, the class evaluation results reveal that some classes have low to suboptimal performance. Notably, classes such as Assorted_Recyclable and Paper_Recyclable achieved low recall scores (81.7% and 79%), but with high precision, the model tends to miss true positives for these classes, but when it does, it is usually correct.

Additionally, Assorted_Non-Recyclable, Paper_Non-Recyclable, and Cardboard_Recyclable exhibited slightly lower precision than other classes but were still acceptable. The reduced precision in Assorted_Non-Recyclable and Cardboard_Recyclable may be likely due to imbalanced bounding box distribution and visual complexity of an image to their counterparts. Similarly, the lower precision for Paper_Non-Recyclable might be due to its recyclable counterpart's visual similarity and complexity. Even with the model's high overall evaluation metrics, it still tends to misclassify and occasionally fails to detect certain classes.

NOT FOR CIRCULATION

3. Preliminary Test Evaluation Result

The model was tested on 50 images, which included five images for each class and 10 null images provided by the researchers. The overall accuracy that the model achieved in the test set is 84%. The low accuracy is due to poor performance on the Assorted Recyclable and Assorted non-Recyclable classes. The model struggled to detect and classify images correctly in these classes, and the possible cause is the complexity of this type of paper.

The class Assorted Recyclable has the lowest performance across all metrics, with 82% accuracy, 25% precision, 40% recall, and an F1-score of 30.8%. This poor result is due to several detection and classification issues: two objects from this class were not detected, two from other classes were misclassified, and four non-paper objects were incorrectly detected and classified as this class. This means that the model has a high false positive instance and tends to miss objects of this class.

The Assorted non-Recyclable class had the second lowest performance with 92% accuracy, 100% precision, 33.3% recall, and an F1-score of 50%. The model detected only 2 out of 5 objects in this class and did not detect one. The issue is more on the model failing to detect belonging to this class, but when it does, it classifies with this class with high confidence.

4. Real-Time Evaluation Result

NOT FOR CIRCULATION

Before the real-time testing, the model was tested for its detection capability in varying distances and lighting conditions. The distance was adjusted in Scenario A, while in Scenario B and C, it was set to 1m from the camera to fit multiple objects in the frame. The researchers conducted the test in a well-lit room because the model performed poorly in dark or dim lighting conditions.

4.1. Detection of YOLOv8 Model Result

In Scenario A, the model was evaluated for its capability to detect a single paper object in the frame at a time. It detects 51 out of 60 paper objects, demonstrating its performance using FNPDR. This shows that the model did not detect 15% of the paper objects, resulting in an accuracy of 85% for single-object paper detection.

In Scenario B, the model was evaluated for its capability to detect multiple paper objects in the frame. It detects 47 out of 55 paper objects, demonstrating its performance using FNPDR. This shows that the model did not detect 14.54% of the paper objects, resulting in an accuracy of 85.45% for multiple-object paper detection.

In Scenario C, the model was evaluated for its capability to detect multiple paper objects with non-paper objects in the frame. It detects 39 out of 55 paper objects, while seven non-paper objects were not detected, demonstrating its performance using FNPDR. This shows that 29.09% of the paper objects were not detected by the model, resulting in an accuracy of 74.19% for multiple-object paper detection with non-paper objects.

NOT FOR CIRCULATION

Researchers observed that paper object detection accuracy drops to 74.19% when near non-paper objects. Scenarios A and B maintained an accuracy of 85% when detecting paper, newspaper, and assorted categories at distances of 0.80 m to 1 m. At the same time, corrugated boxes required a distance of 1.30 m to fit entirely within the frame. Additionally, corrugated cardboard was excluded from scenarios B and C, as only one box could fit in the frame at a time.

4.2. Detection and Classification of YOLOv8 Model Results

In Scenario A, the model achieved an overall accuracy of 82.02%. The factor that makes it a low accuracy is attributed to certain classes with below 95% accuracy and subpar metric scores: Assorted Recyclable, Assorted non-Recyclable, Paper non-Recyclable, and Cardboard Recyclable. Among these, the two lowest performing classes were Assorted Recyclable and Assorted non-Recyclable with below 80% metric scores. Notably, the News Paper non-Recyclable achieved 100% in all metrics, while its counterpart, News Paper Recyclable, achieved near-perfect accuracy of 98.88% with only one paper object misclassified. The model struggled to correctly classify assorted papers with and without contaminants, which may be due to the complexity and diversity of paper types included in this class.

In Scenario B, the model achieved an overall accuracy of 85.87%, which is an increase, but it may be due to the exclusion of the two cardboard

NOT FOR CIRCULATION

classes. The three classes with below 85% scores in the metrics were Paper non-Recyclable, Assorted Recyclable, and Assorted non-Recyclable. Among these, the two lowest performing classes remained the same with Scenario A: Assorted Recyclable and Assorted non-Recyclable with below 90% accuracy and metric scores. Surprisingly, Assorted Recyclable achieved 100% precision, but many objects belonging to it were misclassified, while no other objects were misclassified as this class. Notably, the News Paper Recyclable achieved 100% in all metrics, while its counterpart, News Paper non-Recyclable, achieved a near-perfect accuracy of 98.91% with only one paper object misclassified. The model struggles to correctly classify assorted papers with and without contaminants, even when tested on different paper objects.

In Scenario C, the model achieved an overall accuracy of 81.05%, which is a significant decrease compared to other scenarios and can be due to some classes' difficulty classifying. At the same time, non-paper objects were present, and new sets of objects were tested. Almost all classes had below 85% scores in the metrics except the News Paper non-Recyclable class. Among these classes, Assorted Recyclable and Assorted non-Recyclable remained the same as the two lowest-performing classes. Similarly, Assorted Recyclable achieved 100% precision but still misclassified some of the objects belonging to its class, while no other objects were classified as this class. Despite being the highest performing class, the News Paper non-Recyclable did not achieve 100% in all metrics.

NOT FOR CIRCULATION

All the objects belonging to that class were correctly classified, but other objects were misclassified as this class. With the presence of non-paper objects, the model faces a challenge in correctly classifying all of its classes, especially the two assorted classes.

Conclusion

One of the study's objectives was to gather, preprocess, and create a data set of recyclable and non-recyclable paper images, which the researchers achieved. The gathering phase comprised four distinct paper classes: Paper, Newspaper, Assorted paper, and Corrugated cardboard, following the standard of collection from the dealership of recyclable paper. The preprocessing phase was used to create the differences between recyclable and non-recyclable paper by considering the presence of contaminants as suggested by the National Solid Waste Management (NSWMC) and supported by previous studies. This process involved annotating, augmenting, and splitting the images to create the dataset.

The YOLOv8 model was then trained using the created dataset. Necessary libraries and software were installed, and the dataset was exported into YAML format for model training. The hyperparameter was also set to meet the hardware specification of the device used in the study.

After training, the model produced the performance metrics and class evaluation results. The metrics were recall, precision, and mean Average Precision (mAP) evaluated at the best-performing epoch. The loss curves were constructed to monitor each epoch's performance. The researchers also observed and tested the model's performance in a set of

NOT FOR CIRCULATION

images. Then, the researchers constructed a confusion matrix to visualize the result and manually calculate the model's precision, recall, F1 score, and accuracy.

Real-time testing was implemented across different scenarios to evaluate the model's performance in object paper detection and classification. The paper detection performance was observed, and performance metrics and results were calculated using the FNPDR to determine the model's missed detection of a paper object and the accuracy of the overall detection of the model. The model's paper object classification performance was tested and measured using MCCM and extracting values of TP, TN, FP, and FN for MLCM to calculate accuracy, precision, recall, F1-score, and overall classification accuracy.

The study revealed that the YOLOv8 model did not generalize properly to most classes, particularly classes with complexity, and comprised five distinct paper types, such as Assorted Recyclable and Assorted non-Recyclable. These classes consistently showed lower accuracy and metric scores than the rest across all testing procedures. It highlights the model's difficulty in handling the complexity of the paper types, along with added contamination in the paper types inside these classes. The model's performance was constrained by methodological factors such as data imbalance, lack of model tuning, and comparative evaluation with other algorithms. These limitations prevented the model from achieving consistent classification accuracy across all classes and better generalization.

Despite these limitations, the model showed strong performance in less complex classes with only one paper type, such as News Paper Recyclable and News Paper non-Recyclable achieving high metric scores across all testing procedures. This indicates that

NOT FOR CIRCULATION

it can operate in real time and has potential practical applications in paper recyclability sorting. Furthermore, this research contributes to computer vision and AI by applying YOLOv8 to real-time recyclable and non-recyclable paper detection and classification. This is an area not previously explored in existing studies during the time of this study. This study provides a strong foundation and groundwork with targeted improvements in the model's limitations for future advancements in this field.

Recommendations

The researchers considered the following recommendations for future studies.

1. Expanded and Improved Quality Dataset

Having a more refined dataset can result in better detection and classification of the algorithm. Gather images with more diversity and categories of recyclable and non-recyclable paper contaminants. In particular, ensure that complex classes like Assorted Recyclable and Assorted non-Recyclable include a balanced number and high-quality samples of their subtypes (e.g., book covers, magazines, calendars, colored paper, and brochures). It is also possible to split the subtypes into different classes. Maintain equal class amounts to avoid imbalance that can lead to poor model performance for underrepresented classes. Collect images with different angles, lighting conditions, and backgrounds to increase diversity and apply additional augmentations to help the model generalize better in real-world scenarios.

NOT FOR CIRCULATION

2. Model Tuning and Optimization

Tuning and optimization of the model might lead to more promising performance. Adjusting hyperparameters and modifying the model's architecture, such as fine-tuning the Neck and Head sections, adding more layers, or testing different loss functions. These modifications may improve the results of complex classes without using entirely different architectures.

3. Comparative Algorithm Evaluation

To evaluate whether YOLOv8 is the most suitable algorithm for this task, future studies should compare its performance with other object detection and classification models. This comparative evaluation would provide a fair assessment of the model's strengths, function, and limitations. This is not to discredit YOLOv8 but to determine which algorithm performs best under the complexities of paper recyclability classification. It would provide insight into how different algorithms handle challenging classes and may lead to additional solutions.

4. Scenario-Specific Testing and Evaluation

Future studies should test the model in real-world conditions and various test scenarios. It can be various paper shapes, similarly looking paper objects and contaminants, overlapping paper types, crowded backgrounds, and many more conditions. It can contribute to assessing the model's limitations and capability, which can then be improved.

NOT FOR CIRCULATION

REFERENCES

Adjabi, I., Ouahabi, A., Benzaoui, A., & Taleb-Ahmed, A. (2020). Past, Present, and Future of Face Recognition: A Review. Electronics, 9(8), 1188. https://doi.org/10.3390/electronics9081188

Agbehadji, I., Abayomi, T., Nam, B. K. H., Millham, R., & Freeman, E. (2022). Nature-inspired search method and custom waste object detection and classification model for smart waste bin. Sensors, 22(16), 6176. https://doi.org/10.3390/s22166176

Ahmed, Z., & Sadat Shanto, S. (2024). Performance analysis of YOLO architectures for surgical waste detection in post-covid-19 medical waste management. Malaysian Journal of Science and Advanced Technology, 1–9. https://doi.org/10.56532/mjsat.v4i1.232

Altıkat, Aysun & Gülbe, Alper & Altikat, Sefa. (2021). Intelligent solid waste classification using deep convolutional neural networks. International Journal of Environmental Science and Technology. 19. 10.1007/s13762-021-03179-4.

Arvio, Y., Kusuma, D. T., & BM Sangadji, I. (2024). Inorganic Waste Detection Application Using Smart Computing Technology with YOLOv8 Method. Sinkron : Jurnal Dan Penelitian Teknik Informatika, 8(4), 2389-2396. https://doi.org/10.33395/sinkron.v8i4.14117

Bai, Ruihan & Shen, Feng & Wang, Mingkang & Lu, Jiahui & Zhang, Zhiping. (2023). Improving Detection Capabilities of YOLOv8-n for Small Objects in Remote Sensing Imagery: Towards Better Precision with Simplified Model Complexity. 10.21203/rs.3.rs-3085871/v1.

NOT FOR CIRCULATION

Bianconi, F., Ceccarelli, L., Fernández, A., & Saetta, S. A. (2014). A sequential machine vision procedure for assessing paper impurities. Computers in Industry, 65(2), 325–332. https://doi.org/10.1016/j.compind.2013.12.001

Chen, G., Chen, P., Shi, Y., Hsieh, C.-Y., Liao, B., & Zhang, S. (2019, May 15). Rethinking the usage of batch normalization and dropout in the training of Deep Neural Networks. arXiv.org. https://arxiv.org/abs/1905.05928

Chen, Y., Wang, S., Lin, L., Cui, Z., & Zong, Y. (2024). Computer vision and Deep Learning Transforming Image Recognition and beyond. International Journal of Computer Science and Information Technology, 2(1), 45–51. https://doi.org/10.62051/ijcsit.v2n1.06

Dawn Wilson, Dr. Manusankar C, & Dr. Prathibha P H. (2022). Analytical Study on Object Detection using Yolo Algorithm. International Journal of Innovative Science and Research Technology, 7(8), 587–589. https://doi.org/10.5281/zenodo.7036535

Department of Trade and Industry Bureau of Philippine Standards. (2024, May 14). Standards for recovered paper towards environmental sustainability and Paper Industry Revitalization | Department of Trade and Industry Philippines. https://www.dti.gov.ph/archives/news-archives/standards-for-recovered-paper/

Diwan, T., Anirudh, G., & Tembhurne, J. V. (2022, August 8). Object detection using yolo: Challenges, architectural successors, datasets and applications - multimedia tools and applications. SpringerLink. https://link.springer.com/article/10.1007/s11042-022-13644-y

NOT FOR CIRCULATION

Dwyer, B. (2024, September 10). When should I auto-orient my images?. Roboflow Blog. https://blog.roboflow.com/exif-auto-orientation/

Flores-Calero, M., Astudillo, C. A., Guevara, D., Maza, J., Lita, B. S., Defaz, B., Ante, J. S., Zabala-Blanco, D., & Armingol Moreno, J. M. (2024). Traffic sign detection and recognition using Yolo Object Detection Algorithm: A systematic review. Mathematics, 12(2), 297. https://doi.org/10.3390/math12020297

Gong, B., Ergu, D., Cai, Y., & Ma, B. (2020). A Method for Wheat Head Detection Based on Yolov4. https://doi.org/10.21203/rs.3.rs-86158/v1

Han, T., Cao, T., Zheng, Y., Chen, L., Wang, Y., & Fu, B. (2023). Improving the detection and positioning of camouflaged objects in yolov8. Electronics, 12(20), 4213. https://doi.org/10.3390/electronics12204213

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr.2016.90

Heydarian, Mohammadreza & Doyle, Thomas & Samavi, Reza. (2022). MLCM: Multi-Label Confusion Matrix. IEEE Access. 10. 10.1109/ACCESS.2022.3151048.

Hidayatullah, P., Syakrani, N., Sholahuddin, M. R., Gelar, T., & Tubagus, R. (2025, April 8). Yolov8 to Yolo11: A comprehensive architecture in-depth comparative review. arXiv.org. https://arxiv.org/abs/2501.13400

Hou, Yaohui. (2022). Detection of Domestic Waste Based on YOLO. Frontiers in Computing and Intelligent Systems. 2. 126-128. 10.54097/fcis.v2i1.3182. https://www.researchgate.net/publication/366264641_Detection_of_Domestic_Waste_Based_on_YOLO

NOT FOR CIRCULATION

Huang, H., Wang, B., Xiao, J., & Zhu, T. (2024). Improved small-object detection using yolov8: A comparative study. Applied and Computational Engineering, 41(1), 80–88. https://doi.org/10.54254/2755-2721/41/20230714

Huang, R., Pedoeem, J., & Chen, C. (2018). Yolo-Lite: A real-time object detection algorithm optimized for Non-GPU Computers. 2018 IEEE International Conference on Big Data (Big Data). https://doi.org/10.1109/bigdata.2018.8621865

Iosip, Alina & Dobon, Antonio & Hortal, Mercedes & Bobu, Elena. (2012). The influence of contaminants in the environmental impact of recovered paper: A life cycle assessment perspective. The International Journal of Life Cycle Assessment. 17. 10.1007/s11367-012-0430-y.

Jia, C., Wang, D., Liu, J., & Deng, W. (2024). Performance optimization and application research of YOLOV8 model in object detection. Academic Journal of Science and Technology, 10(1), 325–329. https://doi.org/10.54097/p9w3ax47

Jocher, G. (2023a, July 21). How does Yolov8 Anchor-free detection work? · issue #3362 · Ultralytics/ultralytics. GitHub. https://github.com/ultralytics/ultralytics/issues/3362

Jocher, G. (2023b, May 19). Where and when non-maximum suppression(nms) happens? · issue #2700 · Ultralytics/ultralytics. GitHub. https://github.com/ultralytics/ultralytics/issues/2700

Jocher, G. (2024a, February 23). Why used Silu in yolov8 architecture? · ultralytics · discussion #8382. GitHub. https://github.com/orgs/ultralytics/discussions/8382

NOT FOR CIRCULATION

Jocher, G. (2024b, March 5). Yolov8 - Bottleneck Block · issue #8659 · Ultralytics/ultralytics. GitHub. https://github.com/ultralytics/ultralytics/issues/8659

Keita, Z. (2022, September 28). Yolo Object Detection explained: A beginner's guide. DataCamp. https://www.datacamp.com/blog/yolo-object-detection-explained

Lavanya, G., & Pande, S. D. (2023). Enhancing real-time object detection with Yolo algorithm. EAI Endorsed Transactions on Internet of Things, 10. https://doi.org/10.4108/eetiot.4541

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436–444. https://doi.org/10.1038/nature14539

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. Computer Vision – ECCV 2016, 21–37. https://doi.org/10.1007/978-3-319-46448-0_2

Maity, S. (2023). Yolo (you only look once) algorithm-based Automatic Waste Classification System. JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES, 18(8). https://doi.org/10.26782/jmcms.2023.08.00003

Malge, P. (2024, March 23). Live object detection and image segmentation with yolov8. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2024/03/live-object-detection-and-image-segmentation-with-yolov8/

Marium, A., Srinivasan, D.G., & Shetty, S.A. (2020). Literature Survey on Object Detection using YOLO.

NOT FOR CIRCULATION

Markoulidakis, John & Rallis, Ioannis & Georgoulas, Ioannis & Kopsiaftis, George & Doulamis, Anastasios & Doulamis, Nikolaos. (2021). Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem. 9. 81. 10.3390/technologies9040081.

Mirkhan, A. (2023). Yolo Algorithm: Real-time object detection from A to Z. kili. https://kili-technology.com/data-labeling/machine-learning/yolo-algorithm-real-time-object-detection-from-a-to-z

Mukherjee, S. (2023, November 3). Yolov8: Pioneering breakthroughs in Object Detection Technology. Paperspace Blog. https://blog.paperspace.com/yolov8-a-revolutionary-advancement-in-object-detection-2/#how-does-yolo-work

National Solid Waste Commission. (2018). Volume III: Guidelines - solid waste management division.https://nswmc.emb.gov.ph/wp-content/uploads/2018/03/Final-Report-Volume-III.pdf

Nelson, Joseph (2020, March 9). Why to Add Noise to Images for Machine Learning. Roboflow Blog: https://blog.roboflow.com/why-to-add-noise-to-images-for-machine-learning/

Pan, S., Wang, N., Lin, Y., & Tang, J. (2024). Based On YOLOV8 Intelligent Trash Can Garbage Classification Detection Algorithm. Deleted Journal, 2(1), 28–32. https://doi.org/10.54097/67mska34

Parayno, P. P., & Bustamante, M. G. (2006). Paper Recycling Schemes in the Philippines. Environmental Studies Institute, Miriam College. https://www.wadef.com/projects/isteac/StudyReport__Paper_Recycling_Research_Philippines.Work_Results.pdf

NOT FOR CIRCULATION

Pedro, J. (2023, December 11). Detailed explanation of yolov8 architecture-part 1. Medium. https://medium.com/@juanpedro.bc22/detailed-explanation-of-yolov8-architecture-part-1-6da9296b954e

Rahim, Leena & Abidin, Nor & Aminuddin, Raihah & Abu Samah, Khyrina Airin Fariza & Ibrahim, Asma & Yusoh, Syarifah & Mohd Nasir, Siti. (2024). YOLOv5 Model-Based Real-Time Recyclable Waste Detection and Classification System. 10.1007/978-3-031-53824-7_5.

Rahman, M. O., Hussain, A., Scavino, E., Basri, H., & Hannan, M. A. (2011). Intelligent Computer Vision System for segregating recyclable waste papers. Expert Systems with Applications, 38(8), 10398–10407. https://doi.org/10.1016/j.eswa.2011.02.112

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr.2016.91

Rohit Kundu. (December 16, 2022). F1 score in Machine Learning: Intro & Calculation. V7. https://www.v7labs.com/blog/f1-score-guide

Roy, P. K., Islam, Md. R., & Rafi, Md. J. A. (2024). Non-biodegradable plastic waste detection and classification using Deep Learning: Bangladeshi Environmental scenario. 2024 6th International Conference on Electrical Engineering and Information &Amp; Communication Technology (ICEEICT).

Roy, P. K., Islam, Md. R., & Rafi, Md. J. A. (2024). Non-biodegradable plastic waste detection and classification using Deep Learning: Bangladeshi

NOT FOR CIRCULATION

Environmental scenario. 2024 6th International Conference on Electrical Engineering and Information &Amp; Communication Technology (ICEEICT). https://doi.org/10.1109/iceeict62016.2024.10534312

Safaldin, M., Zaghden, N., & Mejdoub, M. (2024). An improved Yolov8 to detect moving objects. IEEE Access, 12, 59782–59806. https://doi.org/10.1109/access.2024.3393835

Shelke, S. M., Pathak, I. S., Sangai, A. P., Lunge, D. V., Shahale, K. A., & Vyawahare, H. R. (2023). A review paper on Computer Vision. International Journal of Advanced Research in Science, Communication and Technology, 673–677. https://doi.org/10.48175/ijarsct-8901

Schumacher, D. (n.d.). CSPDARKNET53. https://serp.ai/cspdarknet53/

Shamsan, A. (2024, August 28). [7] the silu activation function unlocking neural network potential.pptx. SlideShare. https://www.slideshare.net/slideshow/7-the-silu-activation-function-unlocking-neural-network-potential-pptx/271381753?utm_source=chatgpt.com#6

Shrestha, Shiva & Shrestha, Samman & Paudel, Prateek & Gurung, Sudarshan & Gaire, Sulav & Adhikari, Smita. (2024). Deep Learning for Waste Management: Leveraging YOLO for Accurate Waste Classification. Volume: 15.

Shroff, M., Desai, A., & Garg, D. (2023). Yolov8-based waste detection system for recycling plants: A deep learning approach. 2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS). https://doi.org/10.1109/icssas57918.2023.10331643

Singh, A. (2024, February 20). Selecting the right bounding box using Non-Max suppression (with implementation). Analytics Vidhya.

NOT FOR CIRCULATION

https://www.analyticsvidhya.com/blog/2020/08/selecting-the-right-bounding-box-using-non-max-suppression-with-implementation/

SJÖBERG, A., & HYBERG, agJ. (2023). Investigation Regarding the Performance of YOLOv8 in Pedestrian Detection.

Srivastava, S. R., Meena, Y. K., & Singh, G. (2021). The landscape of soft computing applications for Terrorism Analysis: A Review. Applied Soft Computing, 113, 107977. https://doi.org/10.1016/j.asoc.2021.107977

Sung, Yien-Wen & Li, Jie & Lee, Chao-Yang & Fang, Qingjun. (2025). Improvement of Yolov8 Object Detection Based on Lightweight Neck Model for Complex Images. Image Analysis and Stereology. 69-86. 10.5566/ias.3514.

Syahrudin, E., Utami, E., & Hartanto, A. D. (2024). Augmentation for Accuracy Improvement of YOLOv8 in Blind Navigation System. Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi), 8(4), 579 - 588. https://doi.org/10.29207/resti.v8i4.5931

Szeliski, R. (2011). Computer vision. Texts in Computer Science. https://doi.org/10.1007/978-1-84882-935-0

Tandon, R. & Bhardwaj, N.K. & Mathur, R.M. & Kulkarni, A.G.. (n.d.). Contaminants in recycled paper and their removal. 13. 29-38.

Team, L.O. (2024, March 12). Computer Vision developers consider when selecting a camera . Living Optics. https://www.livingoptics.com/what-computer-vision-developers-should-consider-when-selecting-a-camera/

Tervén, J., Córdova-Esparza, D.-M., & Romero-González, J.-A. (2023). A comprehensive review of YOLO architectures in Computer Vision: From YOLOV1 to

NOT FOR CIRCULATION

Yolov8 and Yolo-Nas. Machine Learning and Knowledge Extraction, 5(4), 1680–1716. https://doi.org/10.3390/make5040083

Timilsina, A. (2024, March 17). Yolov8 architecture explained! Medium. https://abintimilsina.medium.com/yolov8-architecture-explained-a5e90a560ce5

Torres, J., & Austen, J. T. J. (2024, August 31). Yolov8 metrics: Unveiling key insights: Yolov8 guide. YOLOv8. Retrieved November 02, 2024 from https://yolov8.org/unveiling-the-secrets-of-yolov8-metrics/

Ultralytics. (2023a). ultralytics/ultralytics: YOLOv8 – State-of-the-art real-time object detection. GitHub. https://github.com/ultralytics/ultralytics

Ultralytics. (2023b, November 12). Ultralytics YOLOv5 Architecture. Ultralytics YOLO Docs. https://docs.ultralytics.com/yolov5/tutorials/architecture_description/ (Accessed April 21, 2025)

Ultralytics. (2024, April 17). Home. Ultralytics YOLOv8 Docs. https://docs.ultralytics.com/

Ultralytics. (2025, April 13). Silu activation function explained. https://www.ultralytics.com/glossary/silu-sigmoid-linear-unit

Vaj, T. (2024, June 29). How YAML file helps Yolo. Medium. https://vtiya.medium.com/how-yaml-file-helps-yolo-a07a5877e7e5

Volume III: Guidelines - solid waste management division. (2018). https://nswmc.emb.gov.ph/wp-content/uploads/2018/03/Final-Report-Volume-III.pdf

NOT FOR CIRCULATION

Wahyutama, Aria Bisma, and Mintae Hwang. 2022. "YOLO-Based Object Detection for Separate Collection of Recyclables and Capacity Monitoring of Trash Bins" Electronics 11, no. 9: 1323. https://doi.org/10.3390/electronics11091323

Wang, Y., Marcello, C., Sawant, N., Salam, A., Abubakr, S., Qi, D., & Li, K. (2022). Identification and characterization of sticky contaminants in multiple recycled paper grades. Cellulose, 30(3), 1957–1970. https://doi.org/10.1007/s10570-022-05006-6

Wang, Minghao. (2023). Comparative Analysis of YOLOv8 Architectural Variations Based on Detection of Cow Stall Labels. 10.13140/RG.2.2.16044.03206.

Wang, Y., & Deng, T. (2024). Enhancing elderly care: Efficient and reliable real-time fall detection algorithm. DIGITAL HEALTH, 10. https://doi.org/10.1177/20552076241233690

Wu, Zhangnan, Yajun Chen, Bo Zhao, Xiaobing Kang, and Yuanyuan Ding. 2021. "Review of Weed Detection Methods Based on Computer Vision" Sensors 21, no. 11: 367. https://doi.org/10.3390/s21113647

Xiao, Bingjie & Nguyen, Minh & Yan, Weiqi. (2023). Fruit ripeness identification using YOLOv8 model. Multimedia Tools and Applications. 83. 1-18. 10.1007/s11042-023-16570-9.

Zailan, N. A., Azizan, M. M., Hasikin, K., Mohd Khairuddin, A. S., & Khaipanruddin, U. (2022). An automated solid waste detection using the optimized YOLO model for riverine management. Frontiers in Public Health, 10. https://doi.org/10.3389/fpubh.2022.907280

NOT FOR CIRCULATION

Zhang, H., & Hong, X. (2019). Recent progresses on object detection: A brief review. Multimedia Tools and Applications, 78(19), 27809–27847. https://doi.org/10.1007/s11042-019-07898-2

Zhang, X., & Tian, Y. (2024, January). Traffic sign detection algorithm based on improved ... Traffic Sign Detection Algorithm Based on Improved YOLOv8s. https://www.engineeringletters.com/issues_v32/issue_1/EL_32_1_19.pdf

Zhang, Z. (2023). Drone-yolo: An efficient neural network method for target detection in drone images. Drones, 7(8), 526. https://doi.org/10.3390/drones7080526

Zhao, X. (2024). Deep learning based visual perception and decision-making technology for Autonomous Vehicles. Applied and Computational Engineering, 33(1), 191–200. https://doi.org/10.54254/2755-2721/33/20230265

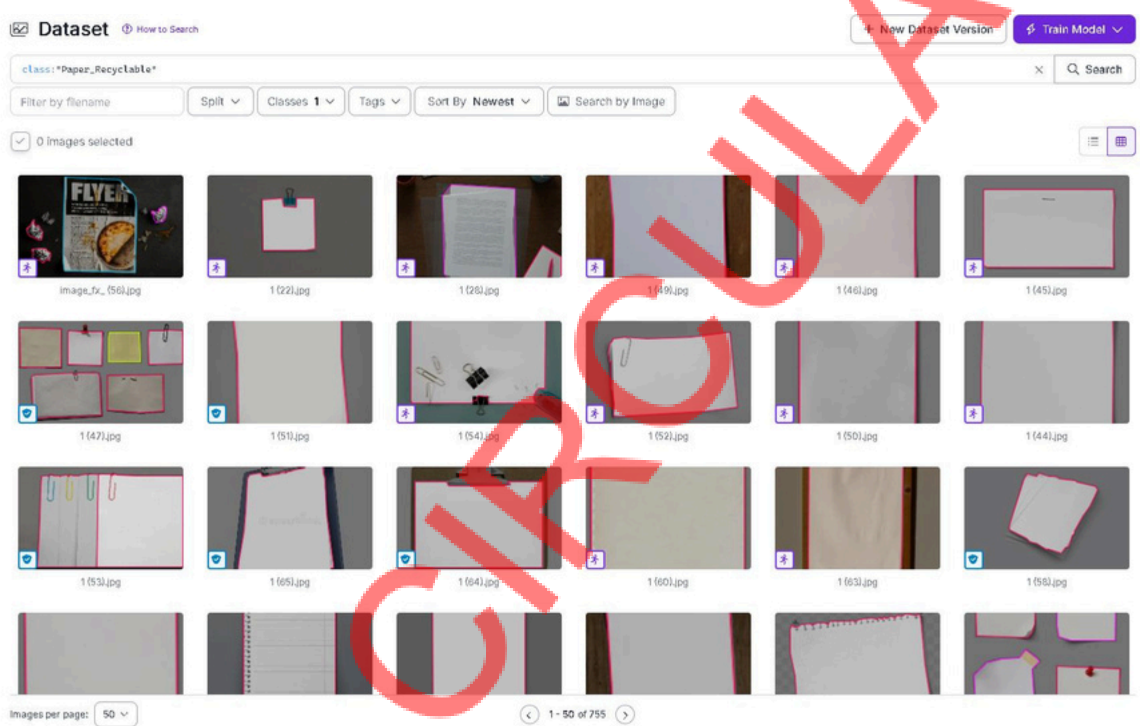
Zhou, X., Chen, W., & Wei, X. (2024). Improved field obstacle detection algorithm based on yolov8. Agriculture, 14(12), 2263. https://doi.org/10.3390/agriculture14122263

NOT FOR CIRCULATION

APPENDICES

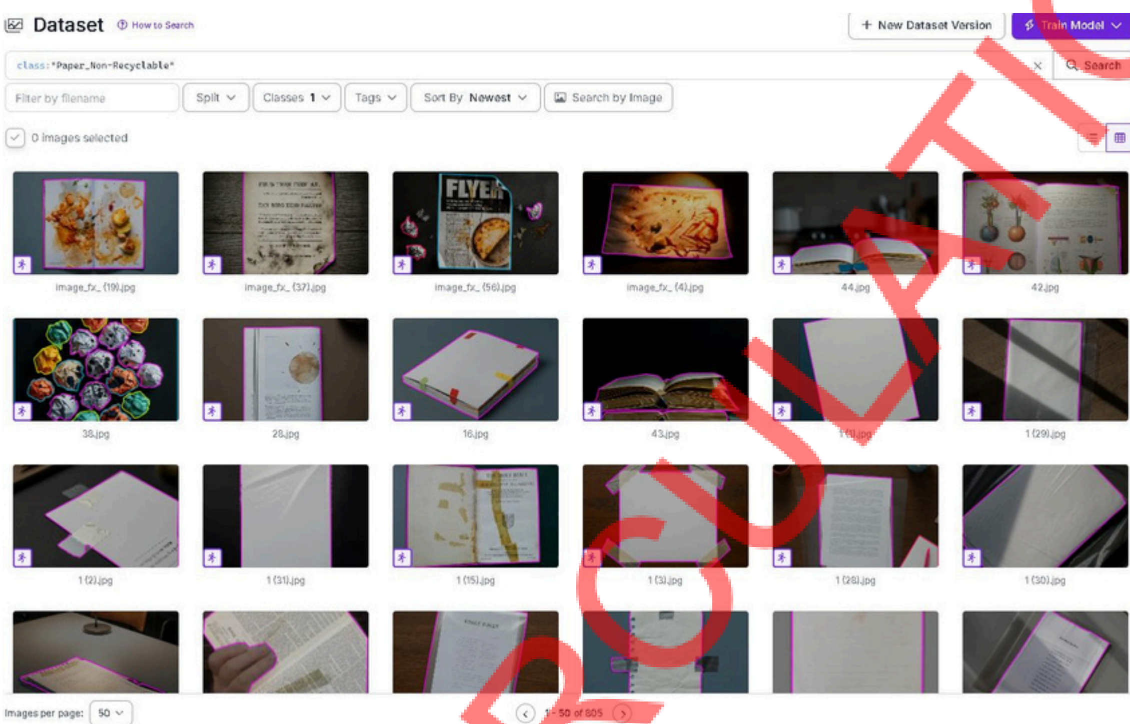
APPENDIX A: SAMPLE IMAGE OF DATASET

Class: Paper_Recyclable

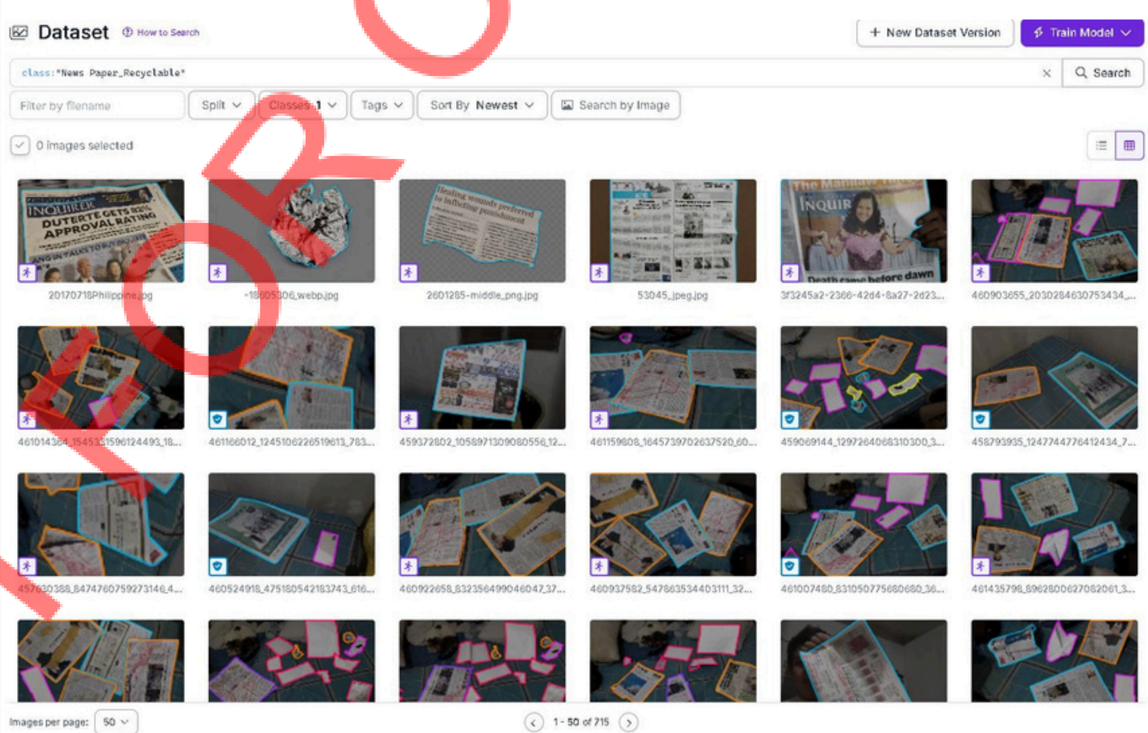


NOT FOR CIRCULATION

Class:Paper_Non-Recyclable

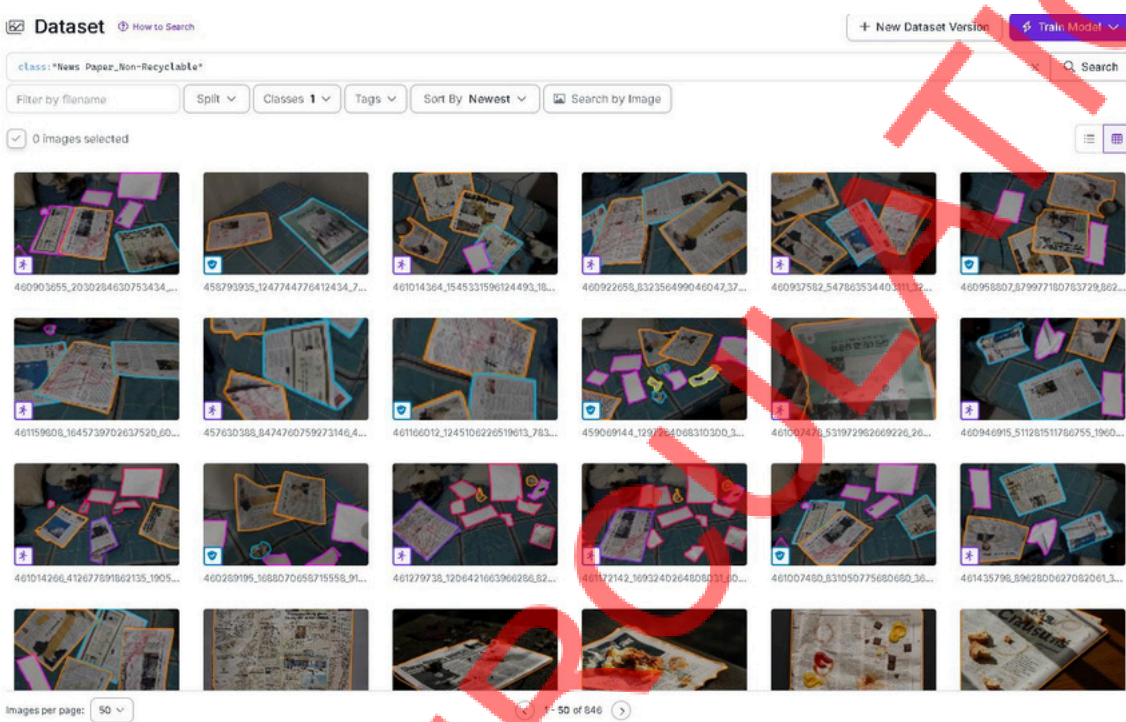


Class: News Paper_Recyclable

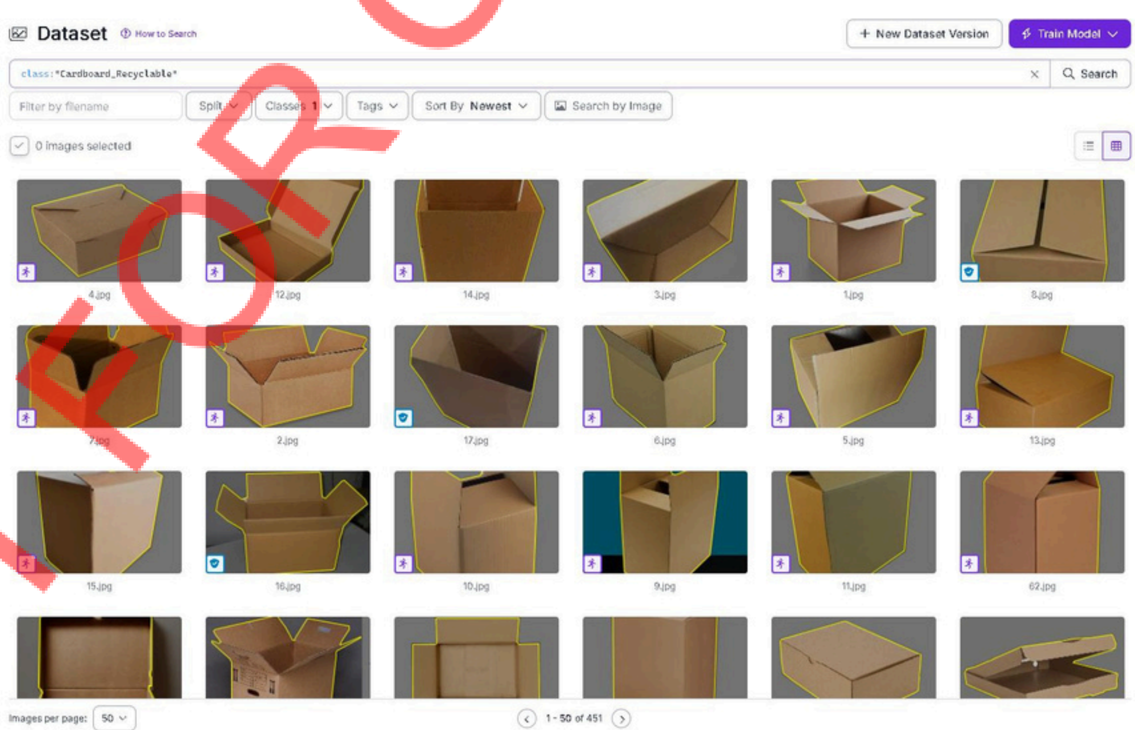


NOT FOR CIRCULATION

Class: News Paper_Non-Recyclable

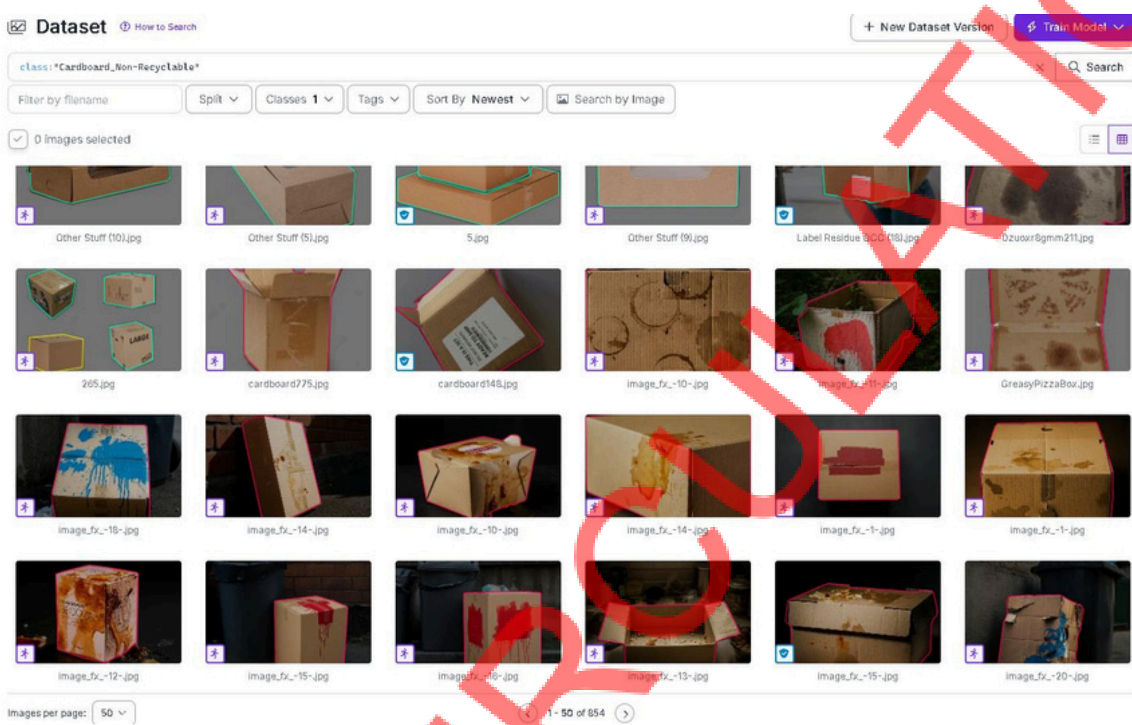


Class: Cardboard_Recyclable

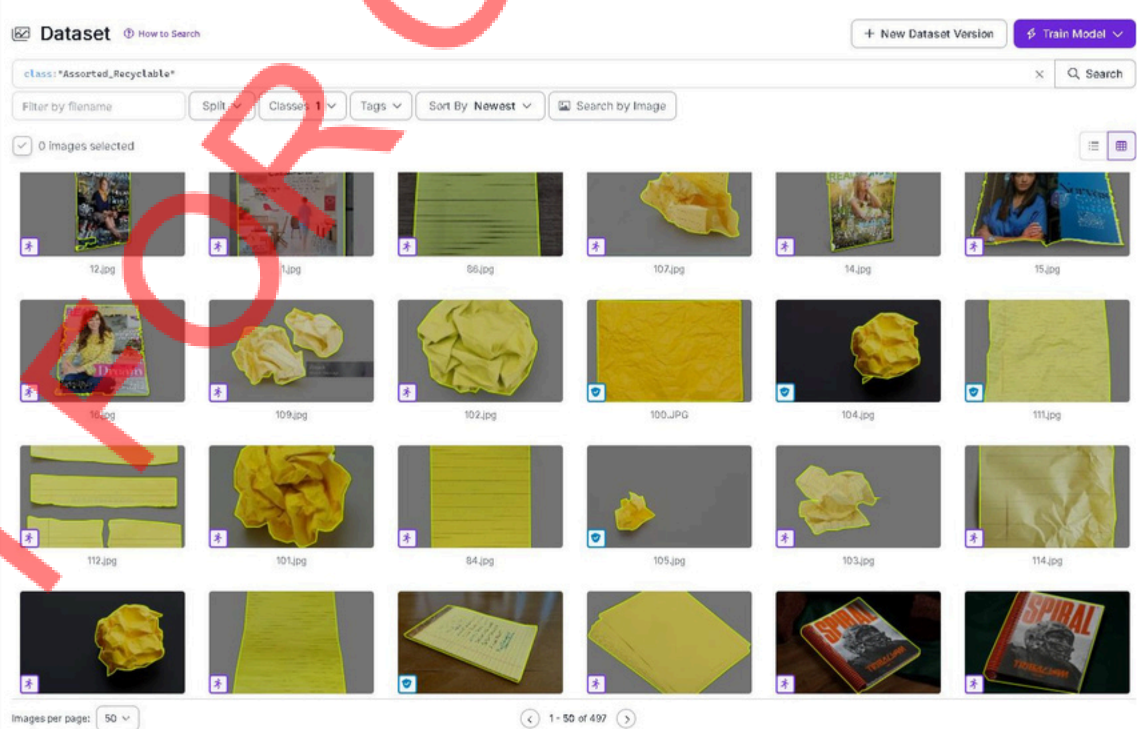


NOT FOR CIRCULATION

Class: Cardboard_Non-Recyclable

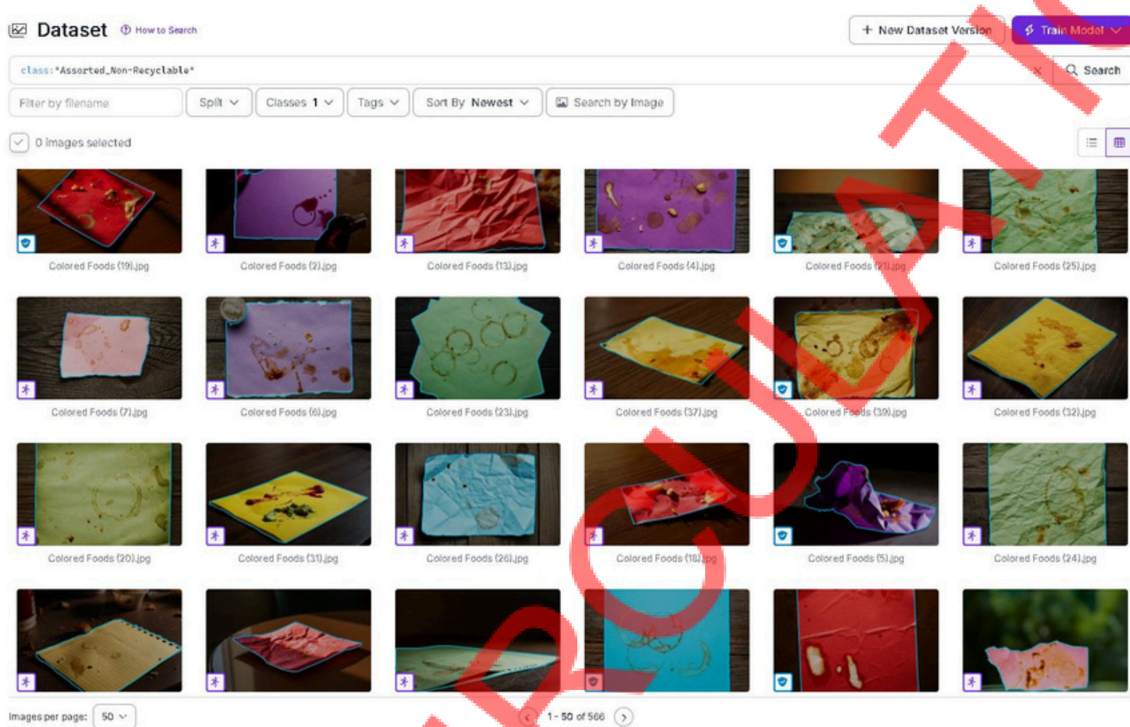


Class: Assorted_Recyclable

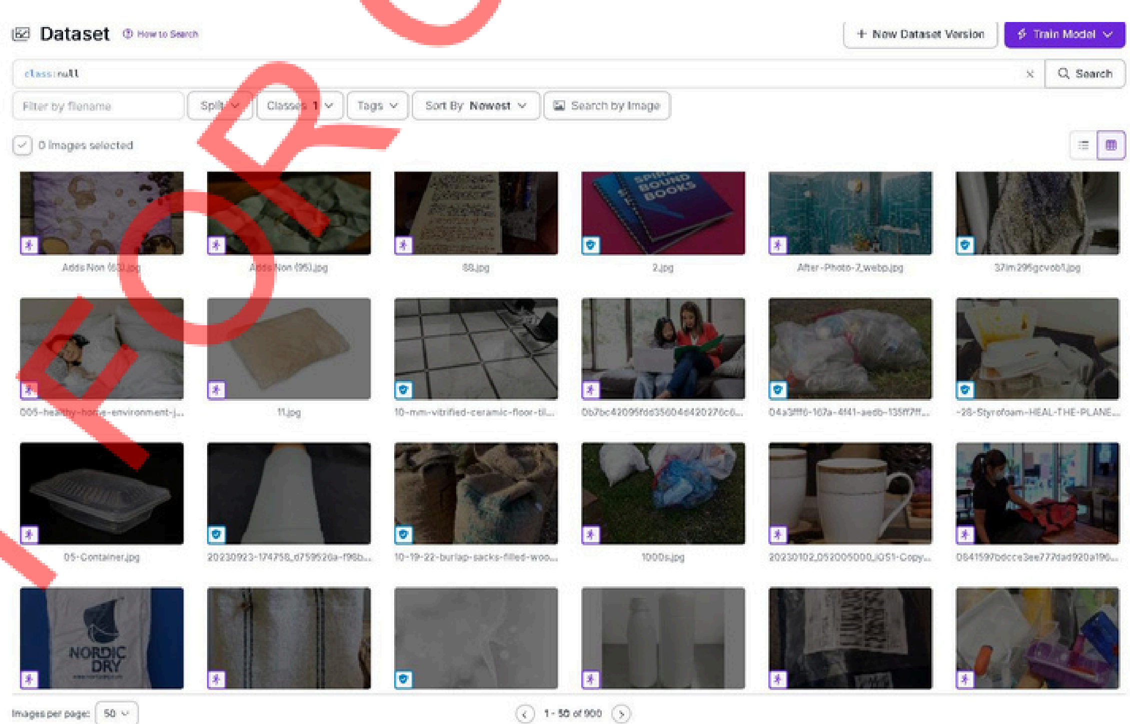


NOT FOR CIRCULATION

Class: Assorted_Non-Recyclable



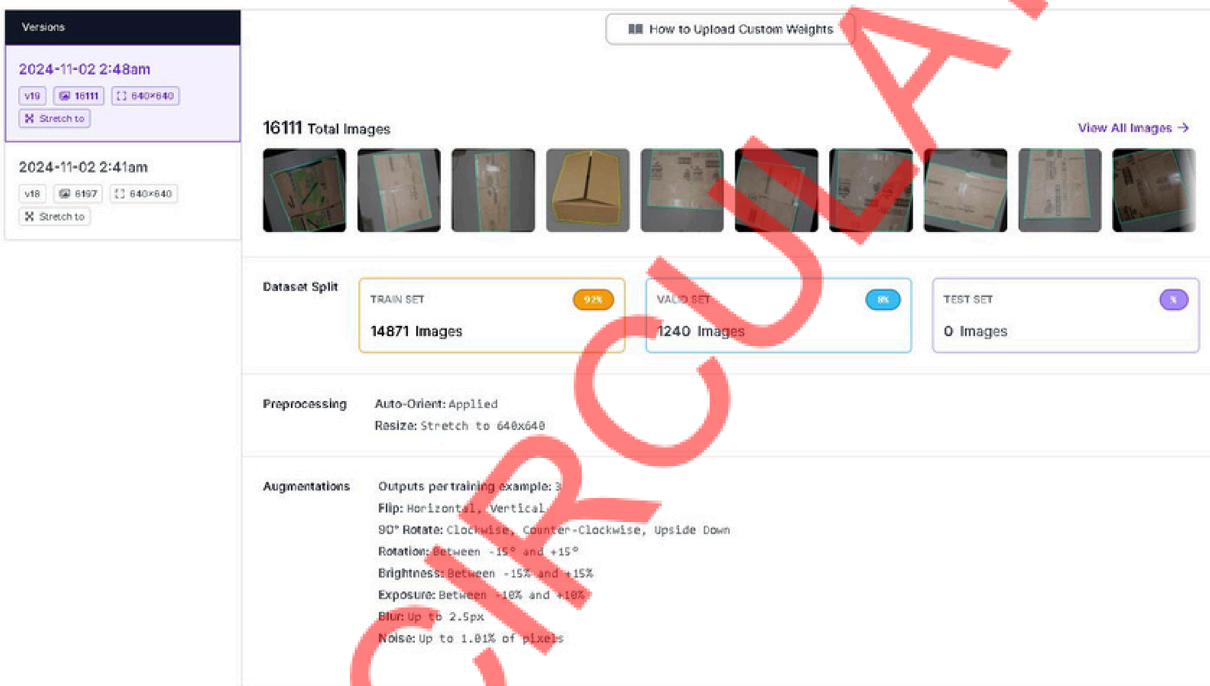
Class: Null/Background



NOT FOR CIRCULATION

APPENDIX B: SCREENSHOT OF PREPROCESS, AUGMENTATIONS, AND ANNOTATION OF IMAGES

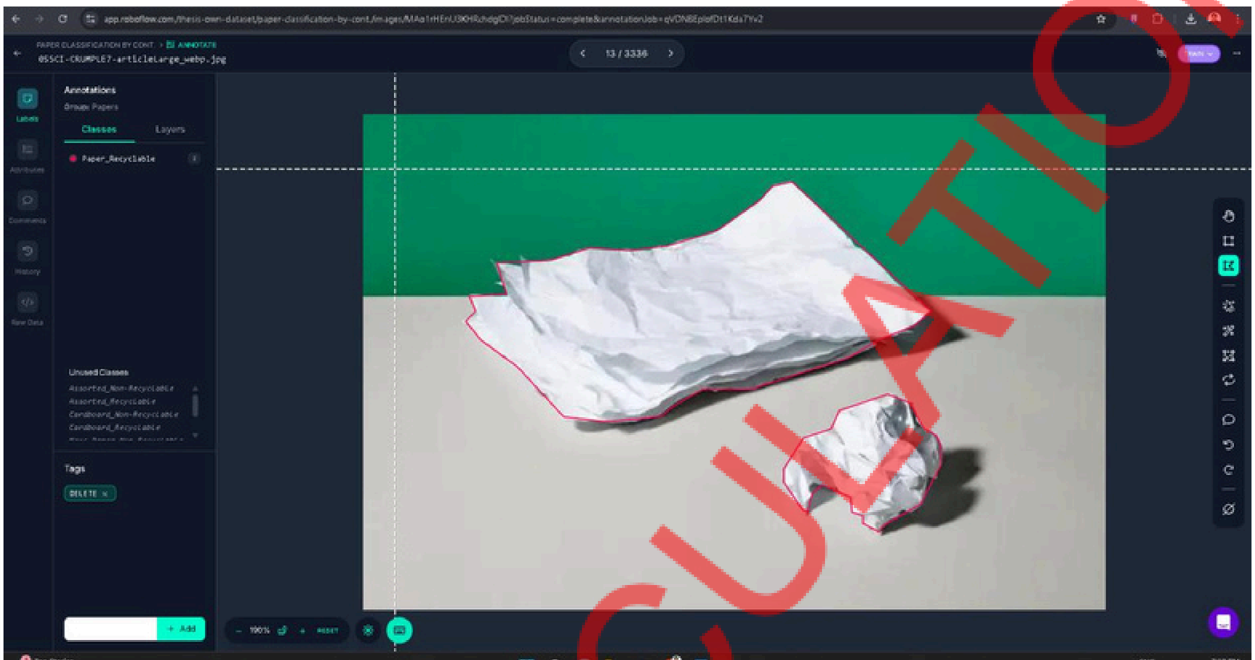
PREPROCESS AND AUGMENTATION

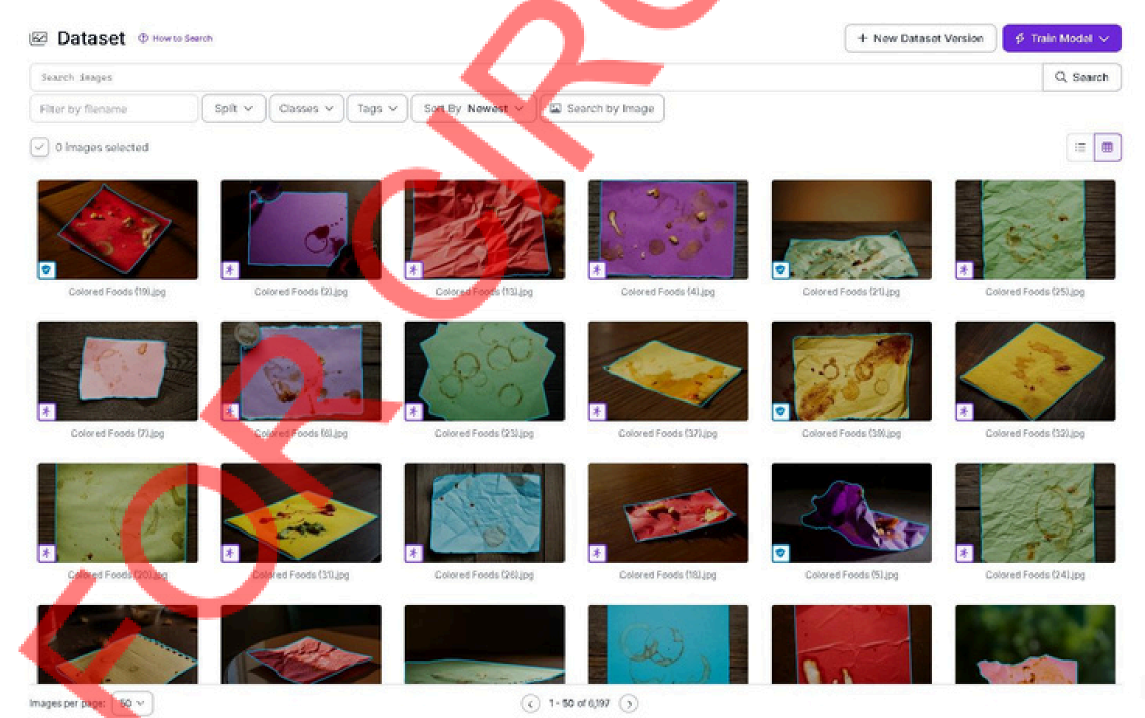


ANNOTATION

During the annotation, the researchers implemented Smart Polygon (AI Labeling) feature by Roboflow which allows the researchers to freeform draw annotations for more precise shapes.

NOT FOR CIRCULATION





NOT FOR CIRCULATION

APPENDIX C: SOURCE CODE

Initialization

```
import os
HOME = os.getcwd()
print(HOME)
```

```
D:\PaperCYolov8s.1a\PaperCYolov8s.1
```

This is the code used to determine and display the current working directory (CWD) of the Python script or Jupyter Notebook.

```
[2]: from IPython import display
     display.clear_output

     import ultralytics
     ultralytics.checks()
     print(ultralytics.__version__)
```

The code imports necessary modules for managing Jupyter notebook outputs *(display)*, initializes the Ultralytics YOLO library for object detection tasks, checks system compatibility, and prints the library version.

```
from ultralytics import YOLO
from IPython.display import display, Image
model = YOLO(f'D:/PaperCYolov8s.1a/PaperCYolov8s.1/yolov8s.pt')
```

This code imports the YOLO class for loading a YOLOv8 model, sets up image display in Jupyter using IPython.display, and loads a pre-trained YOLOv8s model from the specified file path for object detection tasks.

NOT FOR CIRCULATION

```
[6]: import torch

print(torch.__version__)
print(torch.cuda.is_available())
```

This code is used to check the version of PyTorch installed and verify if CUDA (for GPU acceleration) is available in the system.

Download Dataset from Roboflow

```
[4]: !pip install roboflow

from roboflow import Roboflow
rf = Roboflow(api_key="[redacted]")
project = rf.workspace("thesis-own-dataset").project("paper-classification-by-cont.")
version = project.version(19)
dataset = version.download("yolov8")
```

This script installs the Roboflow Python library, connects to a Roboflow project using an API key, accesses a specific dataset version from the "thesis-own-dataset" workspace, and downloads it in YOLOv8 format for training or inference.

Checking GPU Availability for PyTorch Operations

```
import torch

# Check if a GPU is available
if torch.cuda.is_available():
    print(f"GPU is available. GPU name: {torch.cuda.get_device_name(0)}")
else:
    print("GPU is not available. Using CPU instead.")
```

NOT FOR CIRCULATION

This script checks if a GPU is available for PyTorch operations; if available, it prints the GPU's name, otherwise, it indicates that computations used the CPU.

Training YOLOv8s Model

```
]: from ultralytics import YOLO
import torch

model.train(
    data="D:/PaperCYolov8s.1a/PaperCYolov8s.1/Paper-Classification-by-Cont.-19/data.yaml",
    amp = False,
    batch=16,
    epochs=150,            # Number of epochs
    patience=20

)
      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
      9/150      7.52G     0.5067     0.8338      1.043         22        640: 100%|██████████| 930/930 [04:37<00:00,
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|██████████| 39/39 [02:20
                   all       1240       1294      0.807      0.771      0.852      0.767

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
     10/150      7.57G     0.4972     0.8062      1.036         26        640: 100%|██████████| 930/930 [04:51<00:00,
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|██████████| 39/39 [02:31
                   all       1240       1294      0.817      0.782      0.851       0.77

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
     11/150      7.56G     0.4801      0.766      1.025         14        640: 100%|██████████| 930/930 [06:15<00:00,
```

This code trains a YOLOv8 model using the specified dataset and configuration, with parameters for batch size, number of epochs, and early stopping (patience) criteria.

```
model=YOLO("D:/PaperCYolov8s.1a/PaperCYolov8s.1/runs/detect/train/weights/last.pt")
model.train(resume=True, epochs=150)
```

This code resumes training of a previously saved YOLOv8 model from the specified checkpoint and continues for 150 more epochs.

NOT FOR CIRCULATION

Evaluating YOLOv8 Model Performance on Validation Dataset

```
from ultralytics import YOLO

model = YOLO("D:/PaperCYolov8s.1a/PaperCYolov8s.1/runs/detect/train/weights/best.pt")

results = model.val(data="D:/PaperCYolov8s.1a/PaperCYolov8s.1/Paper-Classification-by-Cont.-19/data.yaml")

print(results)
print(results.box.map)
```

This script evaluates a trained YOLOv8 model using a validation dataset, then prints the evaluation results and the mapped box values which likely related to detection metrics such as plots and bounding box information.

NOT FOR CIRCULATION

Real-time Object Classification and Saving Detected Objects Using YOLOv8

```
import cv2
import os
from ultralytics import YOLO

# Load the model
model = YOLO("D:/PaperCYolov8s.1a/PaperCYolov8s.1/runs/detect/train/weights/best.pt")

# Create a directory to save detected objects if it doesn't exist
output_dir = "detected_objects"
if not os.path.exists(output_dir):
    os.makedirs(output_dir)

# Access the webcam
cap = cv2.VideoCapture(0)

frame_count = 0  # Initialize a frame counter for unique image names

while True:
    ret, frame = cap.read()  # Read the webcam frame
    if not ret:
        break

    # Perform inference on the current frame
    results = model.predict(frame, conf=0.5)

    # Iterate over the results
    for result in results:
        for box in result.boxes:
            # Get the bounding box coordinates
            x1, y1, x2, y2 = map(int, box.xyxy[0])  # Convert coordinates to integers
            class_id = int(box.cls[0])  # Get class index
            label = model.names[class_id]  # Get class label

            # Draw the bounding box
            cv2.rectangle(frame, (x1, y1), (x2, y2), (255, 0, 0), 2)
            # Put the class name on the frame
            cv2.putText(frame, label, (x1, y1 - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255, 255), 2)

            # Crop and save the detected object
            detected_object = frame[y1:y2, x1:x2]
            file_name = f"{label}_{frame_count}.jpg"
            cv2.imwrite(os.path.join(output_dir, file_name), detected_object)
            frame_count += 1  # Increment the frame count for unique naming

    # Display the frame with annotations
    cv2.imshow("Webcam YOLOv8", frame)

    # Exit on pressing 'q'
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break  # Break the loop

# Release the webcam and close OpenCV windows
cap.release()
cv2.destroyAllWindows()
```

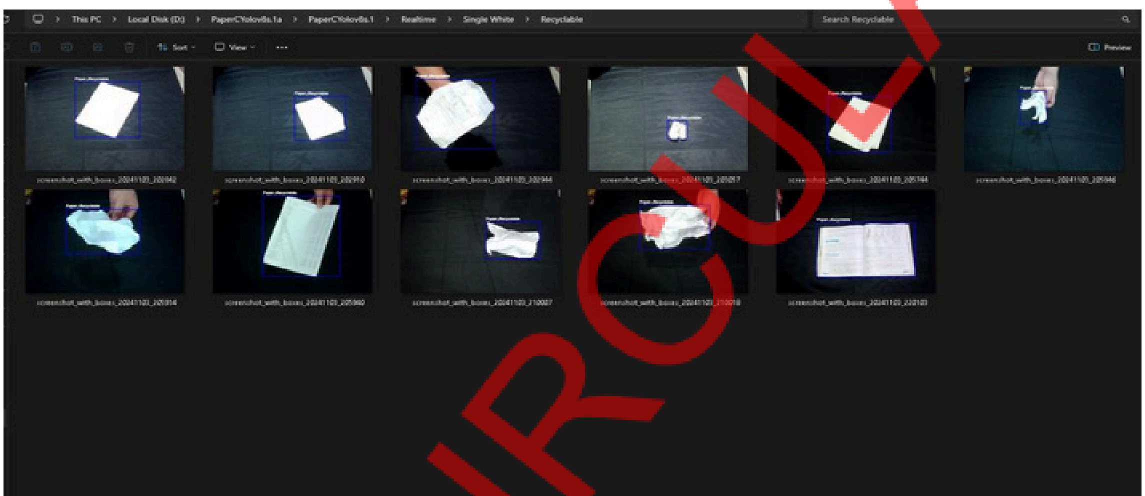
This code uses a YOLOv8 model to perform real-time object detection and classification on webcam frames, drawing bounding boxes around detected objects, displaying the annotated frame, and saving cropped images of the detected objects in a specified directory.

NOT FOR CIRCULATION

APPENDIX D: SAMPLE SAVED DETECTED IMAGES FROM REAL-TIME TESTING PROCEDURE OF THE YOLOV8 MODEL

Scenario A

Class: Paper Recyclable

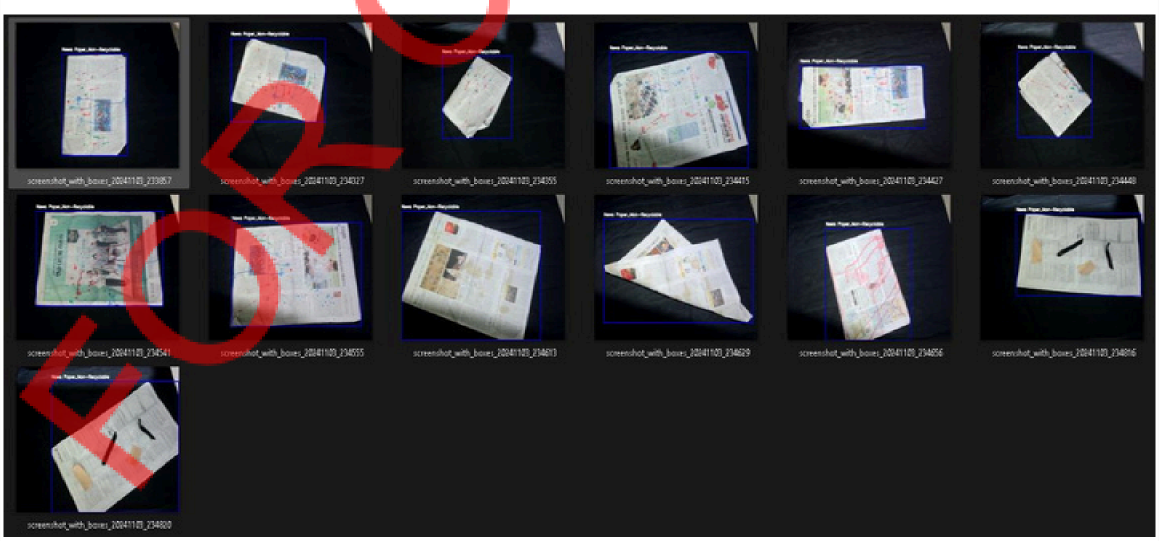


Class: Paper non-Recyclable

NOT FOR CIRCULATION

Class: Newspaper Recyclable

Class: Newspaper non-Recyclable



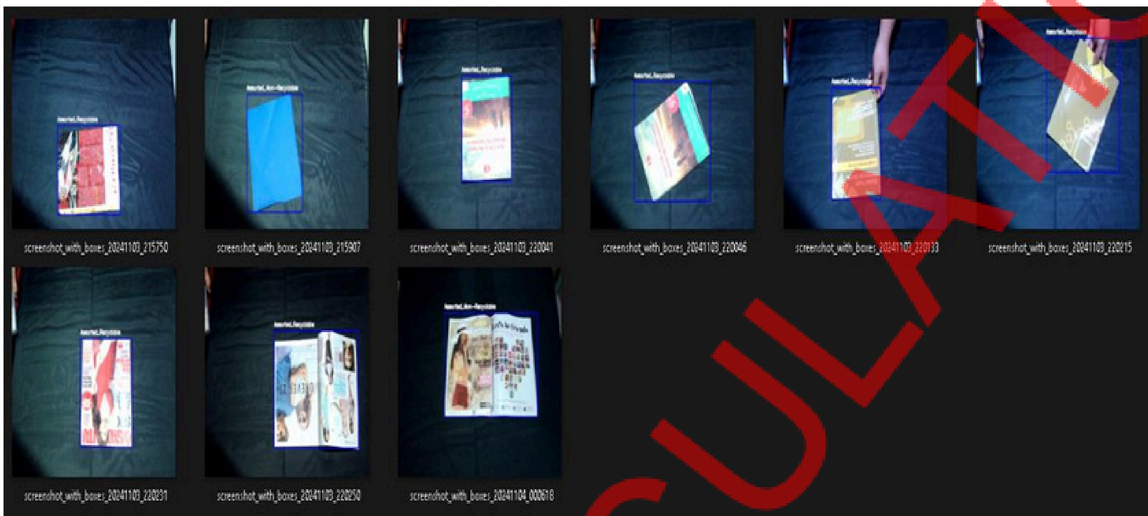
NOT FOR CIRCULATION

Class: Cardboard Recyclable:

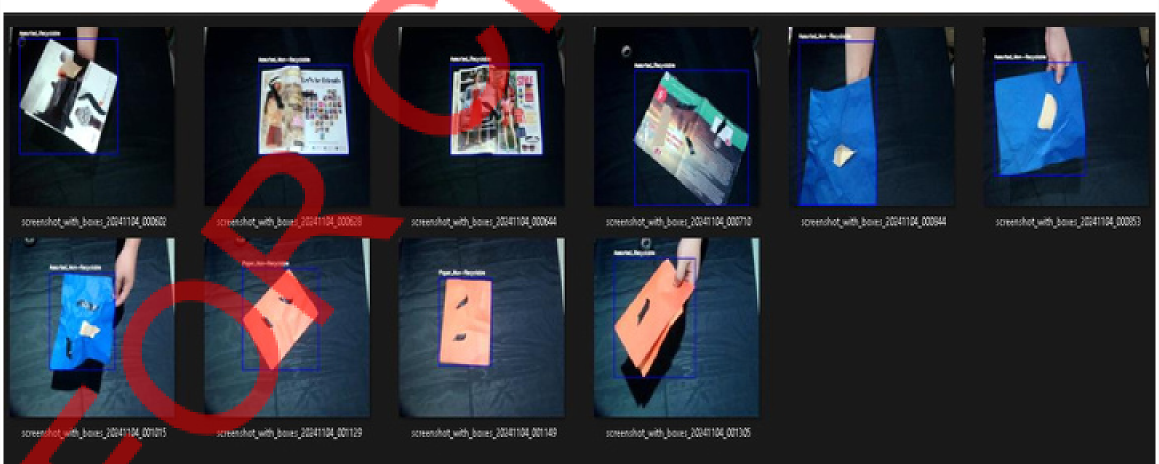
Class: Cardboard non-Recyclable

NOT FOR CIRCULATION

Class: Assorted Recyclable



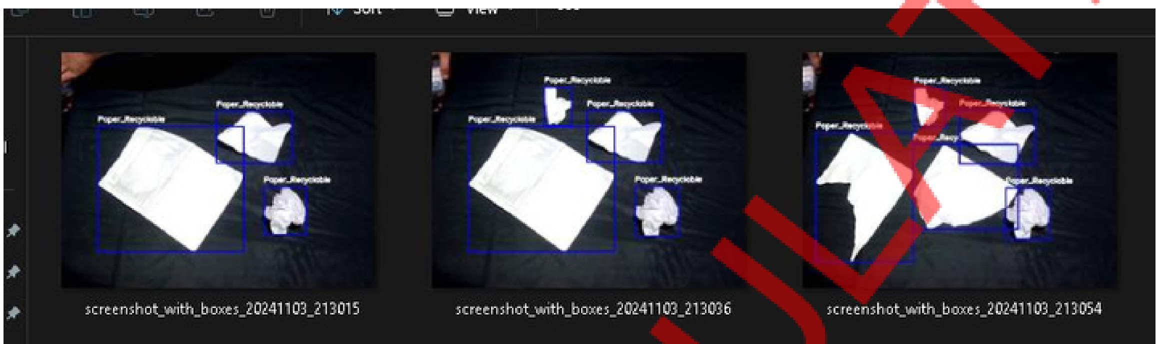
Class: Assorted non-Recyclable



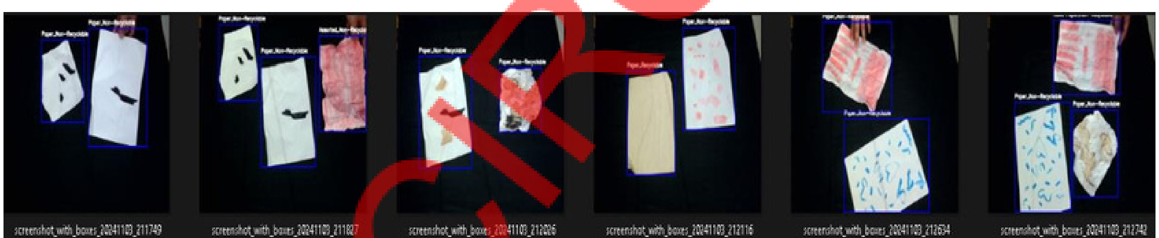
NOT FOR CIRCULATION

Scenario B

Class: Paper Recyclable



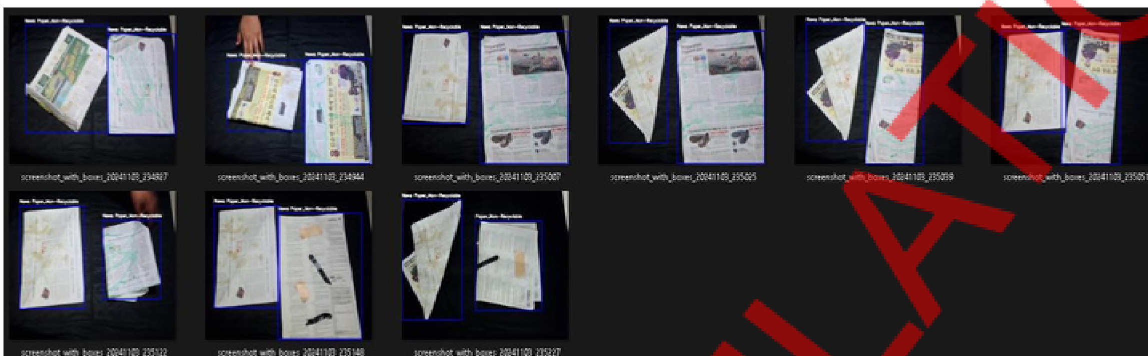
Class: Paper non-Recyclable



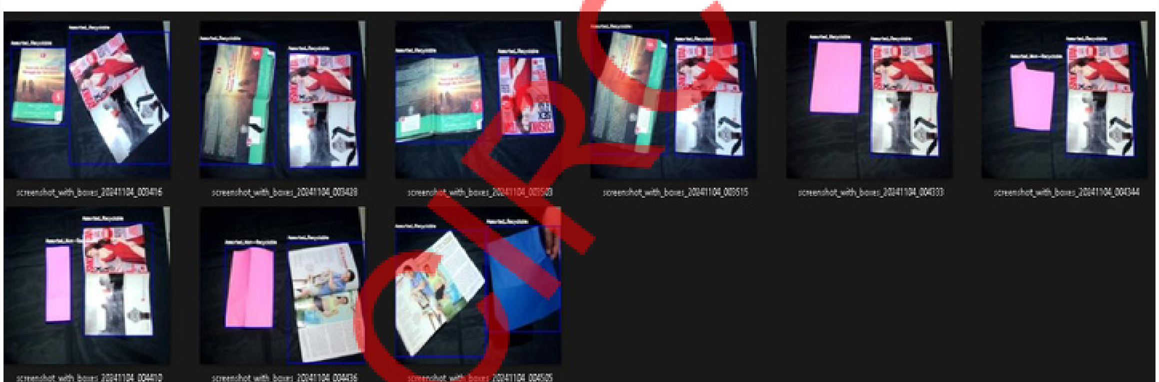
Class: Newspaper Recyclable

NOT FOR CIRCULATION

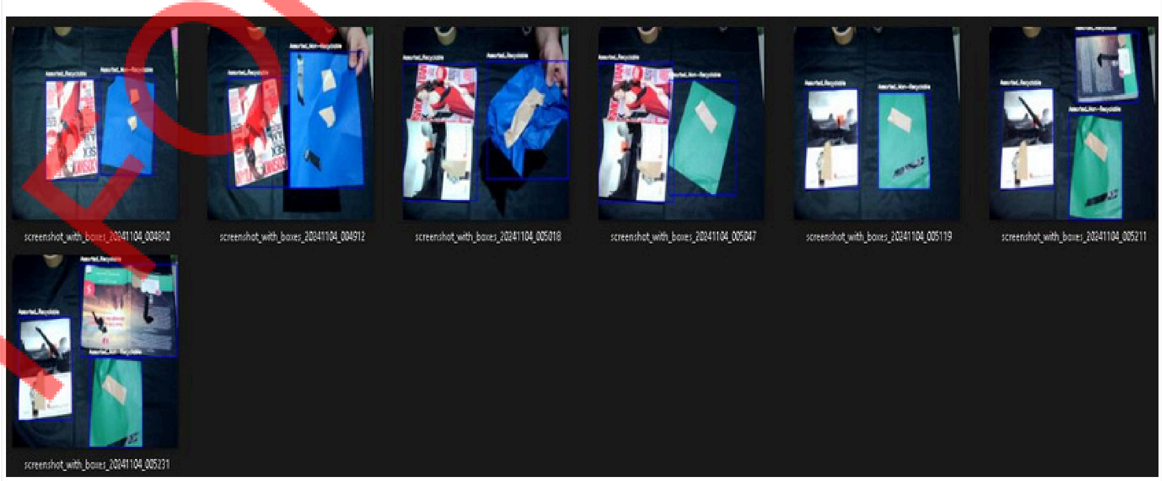
Class: Newspaper non-Recyclable



Class: Assorted Recyclable



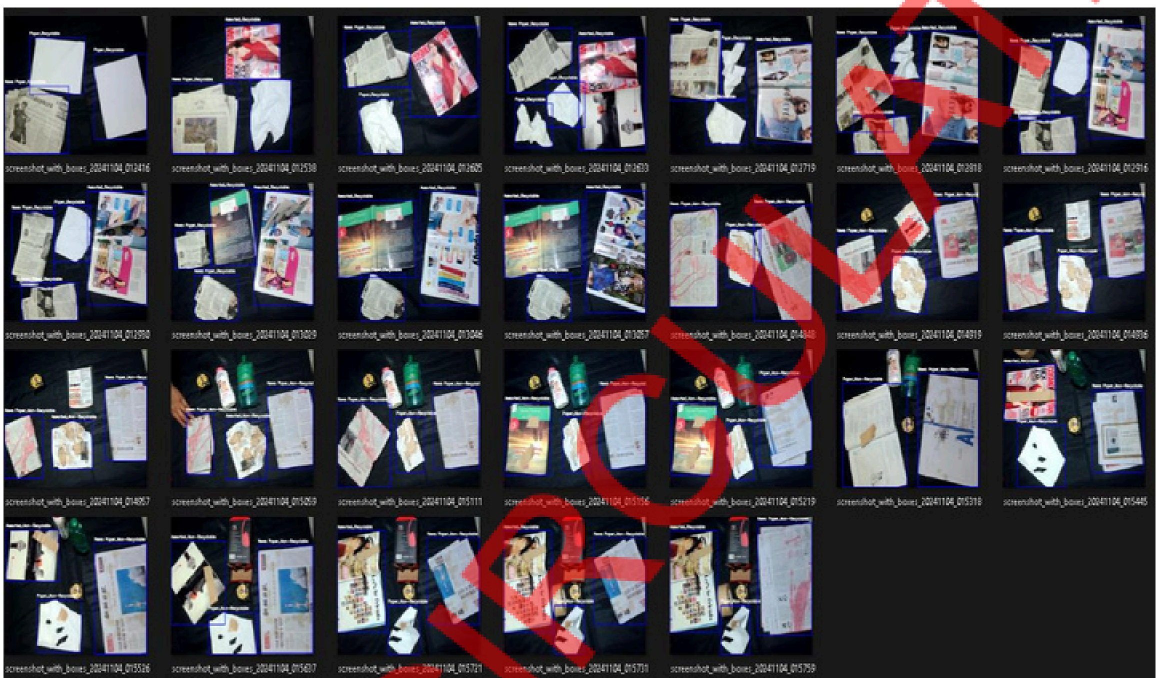
Class: Assorted non-Recyclable



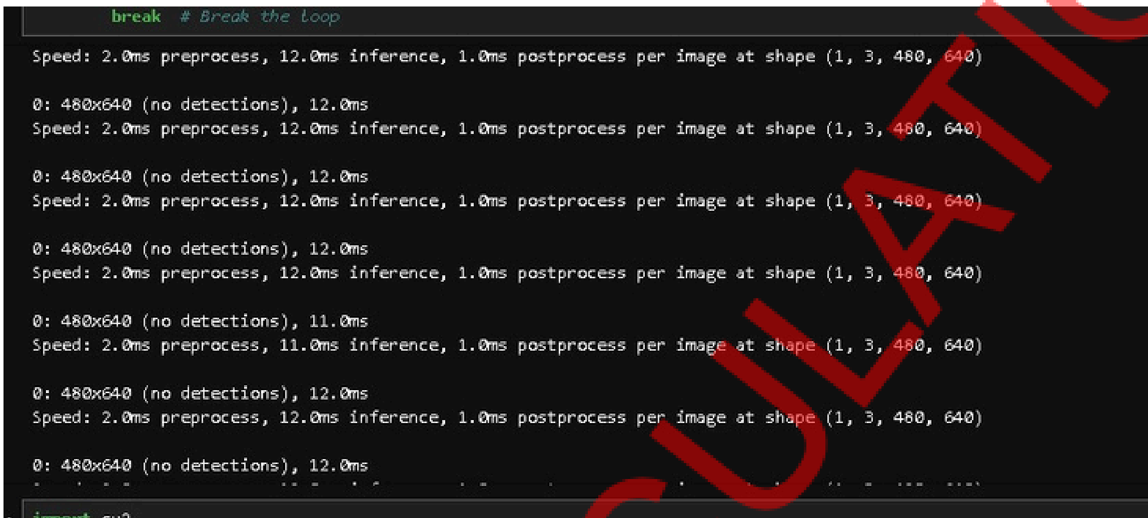
NOT FOR CIRCULATION

Scenario C

Class: All Classes



Frame Per Second During Real-Time Detection



```
        break  # Break the loop
Speed: 2.0ms preprocess, 12.0ms inference, 1.0ms postprocess per image at shape (1, 3, 480, 640)

0: 480x640 (no detections), 12.0ms
Speed: 2.0ms preprocess, 12.0ms inference, 1.0ms postprocess per image at shape (1, 3, 480, 640)

0: 480x640 (no detections), 12.0ms
Speed: 2.0ms preprocess, 12.0ms inference, 1.0ms postprocess per image at shape (1, 3, 480, 640)

0: 480x640 (no detections), 12.0ms
Speed: 2.0ms preprocess, 12.0ms inference, 1.0ms postprocess per image at shape (1, 3, 480, 640)

0: 480x640 (no detections), 11.0ms
Speed: 2.0ms preprocess, 11.0ms inference, 1.0ms postprocess per image at shape (1, 3, 480, 640)

0: 480x640 (no detections), 12.0ms
Speed: 2.0ms preprocess, 12.0ms inference, 1.0ms postprocess per image at shape (1, 3, 480, 640)

0: 480x640 (no detections), 12.0ms
```

The FPS (frames per second) of the Yolov8 model ranges from a minimum of approximately 13.1 FPS, calculated using the highest time per frame of 76.3ms, to a maximum of around 90.9 FPS, derived from the lowest time per frame of 11.0ms. This range highlights the variability in detection speed, depending on the time required to process each frame.

NOT FOR CIRCULATION

APPENDIX E: CERTIFICATE OF LANGUAGE EDITING

CERTIFICATE OF LANGUAGE EDITING

This is to certify that the research study entitled,

"PAPERDOYLO: A YOLOv8 - BASED SYSTEM FOR IDENTIFYING RECYCLABLE PAPER WASTE " conducted by **Gomez, Christian Sydney Earl R.** and **De Guzman, Serge Trever B.** had been subjected to Language Editing at National University- Laguna by the undersigned.

Alliana Miranda Ablan

NOT FOR CIRCULATION

CURRICULUM VITAE

CHRISTIAN SYDNEY EARL R. GOMEZ

gomez.christiansydney@gmail.com | 09182943831

PERSONAL INFORMATION

Date of Birth:	May 15, 2003
Age:	21
Place of Birth:	Calamba, Laguna
Height:	180 cm
Religion:	Catholic
Sex:	Male
Civil Status:	Single

EDUCATIONAL ATTAINMENT

Bachelor's Degree: 2021-Present

National University (NU Laguna)

Bachelor of Science in Computer Science

Senior High School: 2019-2021

Liceo De Los Baños

Accountancy, Business, and Management

NOT FOR CIRCULATION

SERGE TREVER B. DE GUZMAN

sergetrever@gmail.com | 09858042024

PERSONAL INFORMATION

Date of Birth:	June 22, 2002
Age:	22
Place of Birth:	Cabuyao, Laguna
Height:	170 cm
Religion:	Christian
Sex:	Male
Civil Status:	Single

EDUCATIONAL ATTAINMENT

Bachelor's Degree: 2021-Present

National University (NU Laguna)

Bachelor of Science in Computer Science

Senior High School: 2018-2020

Calamba Doctors' College

Science, Technology, Engineering, and Mathematics

NOT FOR CIRCULATION