

Application of YOLOv8 Transfer Learning In Intelligent Garbage Sorting Machine

Nopphagaw Thongbai
Faculty of Information Technology and Management,
Department of Information Technology
King Mongkut's University of Technology North Bangkok,
Thailand
Nopphagaw.T@fitm.kmutnb.ac.th

Wivach Rungshawang
Faculty of Information Technology and Management,
Department of Information Technology
King Mongkut's University of Technology North Bangkok,
Thailand
Wivich.R@fitm.kmutnb.ac.th

Yupin Suppakhun
Faculty of Information Technology and Management,
Department of Information Technology
King Mongkut's University of Technology North Bangkok,
Thailand
Yupin.S@fitm.kmutnb.ac.th

Chanon Warisarn
College of Advanced Manufacturing Innovation,
King Mongkut's Institute of Technology Ladkrabang, Thailand
chanon.wa@kmitl.ac.th

Abstract— This paper proposes an idea to efficiently utilize waste products by integrating hardware and software for garbage sorting. The Smart Bin system aims to efficiently categorize and sort waste using advanced image processing and deep learning techniques. A metal sensor is used to identify metal items, while image processing is used to identify other items. In the image processing stage, images are enhanced with kernel filters before using YOLOv8, real-time object detection model, to predict labels and identify the location of garbage in the images. During the training process, the first 10 layers are frozen and then retrained using a custom image dataset supplemented with images captured from the machine itself. Edge detection in preprocessing further enhances garbage edge predictions and separates objects on the conveyor belt in the sorting machine. The results from the experiment demonstrate the system's capability to identify both the location and type of garbage. This concept is proposed to help solve pollution and waste problems and to inspire the further development of other new projects in the future.

Keywords—Waste identification, Smart recycling, Garbage Sorting Machine, YOLOv8, image Enhancement, Transfer learning

I. INTRODUCTION

The recognition and importance of the environmental management hierarchy, alongside the concept of the circular economy, have progressively increased over the past 10-20 years at both domestic and international levels. This new economic model focuses on recycling natural resources for maximum benefit. The loss of natural resources in quantitative and qualitative terms, along with the side effects of increased waste from dramatic consumption and climate change, have had global impacts. According to the latest information from the Pollution Control Department in Thailand, the total waste generated in 2022 was approximately 25.70 million tons. Of this, about 17.01 million tons were attributed to improperly disposed waste and accumulated unprocessed waste.

All countries are aware of the impact and focus on maximizing the benefits of effective natural resource management with the aim of improving and expanding the recycling of natural resources to create an environmentally friendly process. A simple example of this is the disposal of food packaging in supermarkets and convenience stores, such as milk cartons, plastic bottles, glass bottles, and soft drink cans after consumption. Improper disposal of these items

could lead to pollution and negatively impact the global environment.

Recycling is a process to reusing material by separating and reproducing or improving the quality of material with similar quality as new material or products. This not only reduces the garbage management budget but also saves space for collection and disposal of garbage while saving energy and natural resources in the process. Most importantly, the recycling process helps reduce the greenhouse effect. Therefore, this paper proposes the idea of a garbage sorting machine that utilizes image processing on a Raspberry Pi along with artificial intelligence to categorize materials efficiently.

Rani, Raman, and Ram (2022) reviewed the process involved in building a custom model for detecting and classifying multiple objects using single-stage and two-stage object detection [1]. In the training process, they used an image dataset from TrashNet and a custom dataset called Waste-Mart to detect multiple objects in sample images. The aim of the paper was to build a deep learning model for classifying solid waste tasks and improve the mean Average Precision (mAP) score for the custom dataset using the transfer learning technique. Transfer learning is used to transfer the weights of deep learning models from one domain to solve problems in other domains as well. The results from the experiment indicate that the two-stage detection achieved an mAP of 0.84, whereas the network achieved an mAP of 0.98 with the IoU threshold set at 0.5.

Wu, Zhang, Peng, Fan, Lu, and He (2023) reviewed the methodology and dataset of waste images.[2] The authors introduced the CNN and IWIR outlines, providing insights into the three main tasks of ML image processing: classification, object detection, and segmentation. Three task categories were addressed: (1) the identification of recyclable materials, (2) trash pollution detection, and (3) solid waste classification. Finally, they discussed the prospects of CNNs in this field. In conclusion, the article presents the processes involved in waste identification and sorting for effective waste management.

J. Redmon, S. Divvala, R. Girshick and A. Farhad (2015) introduced YOLO (You Only Look Once), a real-time object detection model[3]. It uses a single neural network for predicting both the class probabilities and bounding boxes of

objects in an image. In the recognition process, YOLO only requires one step to process the whole picture, making it faster and more efficient than other models. YOLO has evolved through several versions, each improving upon the previous ones in terms of speed, accuracy, and features. YOLO is widely used in various computer vision applications, including industrial processes, robotics, and monitoring tasks. Jocher, G., Chaurasiam A., & Qiu, J (2023) YOLO by Ultralytics (Version 8.0.0) implementation [4], Glenn-jocher, Laughing-q, and AyushExel (2023) Instance Segmentation [5], Jansi Rani S.V., Ranghu Raman V., Rahul Ram M., and Prithvi Raj A. (2022) Multi Object detection[6]

Ting-Wei Wu, Hua Zhang, Wei Peng, Fan Lu, and Pin-Jing He., Review about application for intelligent waste identification and recycling (2023)[7] Shengping Wen, Yue Yuan and Jingfu Chen (2023) proposed an idea for identifying waste plastics using deep learning in image classification and methodology to improve the performance of plastic detection in the sorting system [8]. They divided the labels of the image dataset into five categories. The training process involved a combination of image data from online resources and photography of plastics to provide a dataset of 6247 images. They annotated image labels and converted them into VOC format. During the training process, they randomly selected and merged the datasets into training and validation data with n-Fold cross-validation in a 9:1 ratio to evaluate the model for accuracy. YOLOX was implemented by adopting the “CSPDarknet” network as the backbone model.

II. SMART BIN GARBAGE SORTING MACHINE

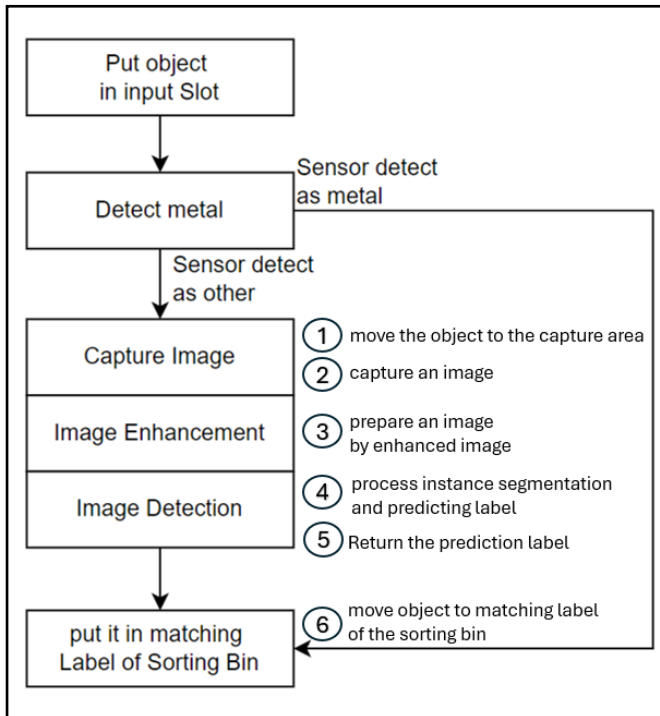


Fig. 1. The full process of the Smart Bin Garbage Sorting Machine

The full process of the Smart Bin Garbage Sorting Machine is explained in Fig. 1. The process starts by placing an object into the input slot. The object is then fed into the metal detection area, where the process initiates using a sensor. If the sensor detects an object as metal, the Smart Bin skips the image capture and instance segmentation prediction processes and places the object directly into the metal bin.

If the sensor detects an object as another, the Smart Bin will process the following steps: (1) move the object to the capture area; (2) capture an image; (3) prepare an image; (4) process instance segmentation and predicting labels; (5) when the prediction process is finished, it will return the prediction label results; and (6) move an object to the matching label of the sorting bin.

A. Prototype of the Smart Bin Garbage Sorting Machine

The Smart Bin system aims to efficiently categorize and sort waste using advanced image processing and deep learning techniques.

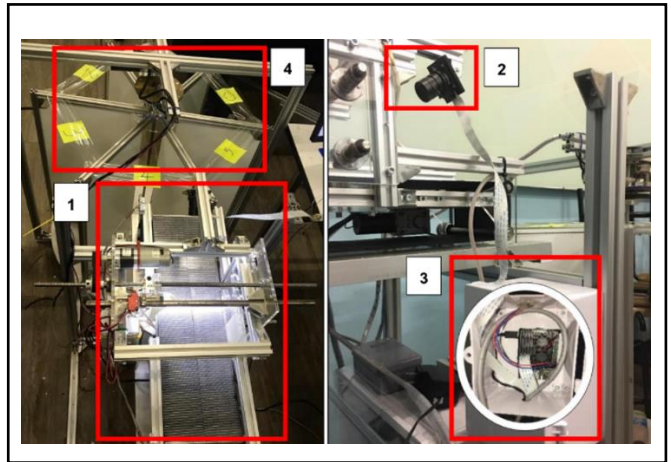


Fig. 2. Prototype of the Smart Bin Garbage Sorting Machine

Fig. 2 shows the main components of the Smart Bin prototype: (1) A conveyor belt transports object from the input unit to the capture area. (2) A vision camera in the capture area captures object image. (3) Raspberry Pi 4 Model B serves as a multipurpose controller, controlling camera, preparing images, predicting instance segmentation, labeling objects, and sending label index to trigger ESP32 motor controller. (4) ESP32 motor controller is used to control sorting bin operation by: (a) convert label index to command to control. (b) send command to rotation the sort bin. (c) move object from capture area. The conveyor belt will transport objects to the sorting bin.

B. Object Detection in Smart Bin

The Smart Bin uses Transfer Learning of YOLOv8 to predict type of garbage object. YOLOv8 is an algorithm for real-time object detection. YOLO is the state-of-the-art model for the classification, detection and segmentation tasks in computer vision.

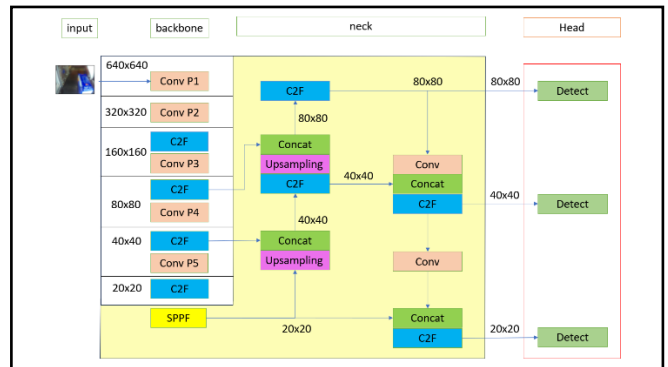


Fig. 3. YOLOv8 architecture

In Fig. 3 presents the YOLOv8 architecture can divide into 3 parts: (1) Backbone is the main body of the network, based on EfficientNet. (2) Neck uses SPPF to connect between Backbone and Head. (3) Head generates the final output for prediction. YOLOv8 uses a single neural network to predict both class probabilities and the bounding boxes of objects in an image. The model outperforms all known models in terms of both accuracy and execution time. The recognition process involves only one step, making it faster and more efficient. During the image segmentation process, IoU and mAP are used to measure the results.

- IoU (Intersection over Union) value is generally used to measure the result. G represents the ground truth pixel area, P represents the prediction result pixel area. The IoU formula is shown in Equation (1).

$$IoU(G, P) = \frac{|G \cap P|}{|G \cup P|} \quad (1)$$

- mAP (mean Average Precision) is a common metric used to evaluate the performance of object detection models and is adopted as an evaluation index. The IoU threshold is set at intervals from 0.5 to 0.95 in increments of 0.05, the threshold index is denoted as i , the evaluation index is denoted as mAP_i , and the mAP formula is presented in Equation (2).

$$mAP = \frac{\sum_{i=0}^{10} mAP_i}{10} \quad i \in \{0.5, \dots, 0.95\} \quad (2)$$

III. EXPERIMENT ENVIRONMENT

The experiment environment can be divided into two parts: (1) Training Environment and (2) Testing Environment. Both environments use Python 3.10 and the PyTorch framework, along with YOLOv8 from Ultralytics detail as show in table 1.

TABLE I. TRAIN AND TEST ENVIRONMENT

Training Environment	Intel Core(TM) i7-8550U CPU running at 1.80 GHz and a built-in Intel UHD Graphics 620 GPU. Windows 10 Environment RAM 16 GB
Testing Environment	Raspberry Pi 4 8 GB BCM2711 ARM processor and Cortex-A72 architecture, running the Debian Bullseye ARM64 operating system installed on a 32 GB SD card.

In the training process, single object images from TrashNet dataset were used, with an addition 25 images capture directly from the Smart Bin environment to create custom instance segmentation by label studio. Label studio is a flexible data labeling tool suitable for preparing label training data in computer vision, natural language processing, speech, voice, and video models.

A. Setting hyperparameter

Before training the data, the first 10-layer's weight were frozen, and the hyperparameter were set as follows: rotate = 0.5, translate: 0.1, scale: 0.15, perspective: 0.1 and mosaic: 0.3. The training process was conducted for 50 and 80 epochs training process with same image resolution as YOLOv8n-seg pretrain model.

The number of classes was set to 6 and the label names were defined as follows: 0:glass, 1:glass_color 2: metal, 3:paper_box, 4:plastic and 5:plastic_color.

Training #1: Batch size = 4, epoch = 50 epochs with transfer leaning from pretrain of YOLOv8 by freezing layer 1-21 during training. The training was completed in 5.474 hours and saved the best weight as last50.pt.

Training #2: Batch size = 4, epoch = 80 epochs continued from best_50.pt without freezing any layer to adjust all weight in model. The model stopped training early at epoch 51 due to not improvement observed in last 50 epochs. 51 epochs completed in 4.838 hours.

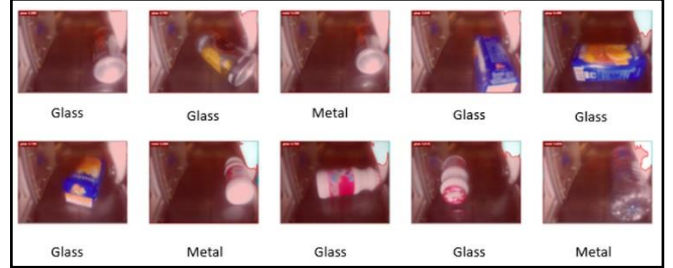


Fig. 4. 50 epoch training without using image enhancement

Fig. 4 displays the results after finish training without using image enhancement. Initially, when implemented in the Smart Bin environment, the model encountered difficulties in detecting object edges and distinguishing object backgrounds in real-time. To address this issue, image enhancements were implemented.

B. Image enhancements

Image enhancement is a process for adjusting some characteristics of image suitable for image analysis by blurring, sharpening, embossing, edge detecting, or adjusting image brightness to make it easier to identify some of the key features. The image enhancement process requires the application of a custom kernel to the image pixels.

The convolution filter applies the kernel value projection on the image window. This filtering process involves mathematically determining the average values of image points within a small set and across the entire picture. The convolution method comprises two main components: The kernel which is a square grid of image pixels with odd dimensions such as 3x3, 5x5 or 7x7, used to enhance the image quality base on desired characteristics. A moving window moves through the image one pixel at a time. The value at the center of the kernel in the moving window is calculated and replaced in the new image.

The individual coefficients within the moving window are multiplied by the numerical value of the image pixel in covers. These products are then summed to form a single image point. The moving window moves through the image column by column, applying the formula repeatedly. Once a column is complete, it moves down to the next row and starts anew. This process continues until the entire image has been processed.

Edge detection is a process in deep learning, image processing, and computer vision to identify boundaries within the area of interest.

This study uses $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ as edge detection kernel.

Filter2D is used to apply a custom kernel to an image. Filter2D masks include various types of kernels for tasks such as blurring, sharpening images, and detecting edges. This kernel function convolves the 2D matrix of the image at the pixel level, resulting in an output image:

```
cv2.filter2D(image, -1, filter).
```

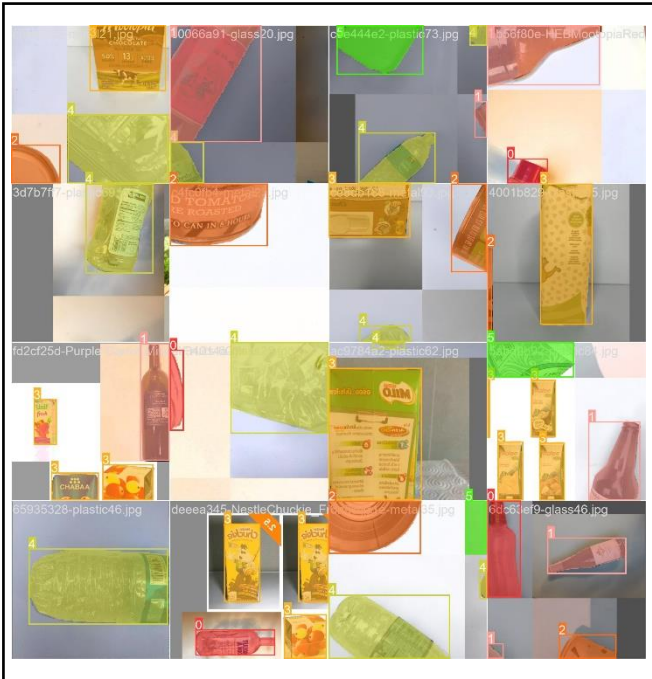


Fig. 5. preview of augmented image generated for enhanced training purposes.

Fig. 5 shows a preview of augmented images generated for enhanced training purposes. These images are created and used in the training process.

Data augmentation involves creating new data from existing data to train a machine learning model effectively. It serves as a form of regularization, helping to reduce overfitting by promoting better generalization rather than memorization.

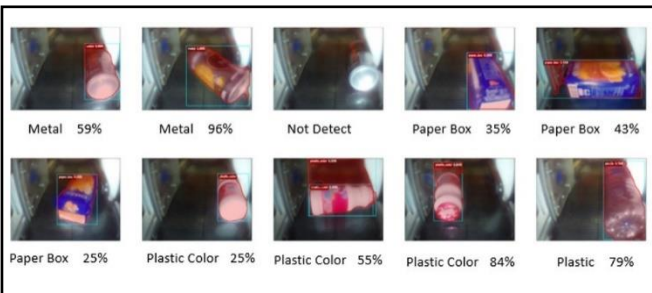


Fig. 6. 50 epoch training with using image enhancement

Fig. 6 shows the results after completing 50 epochs of training with image enhancement. The experiment demonstrates that with image enhancement, the model can correctly identify the edges of objects. When applied to the real-time Smart Bin environment, the transfer learning model accurately detects garbage object edges and identifies specific labels.

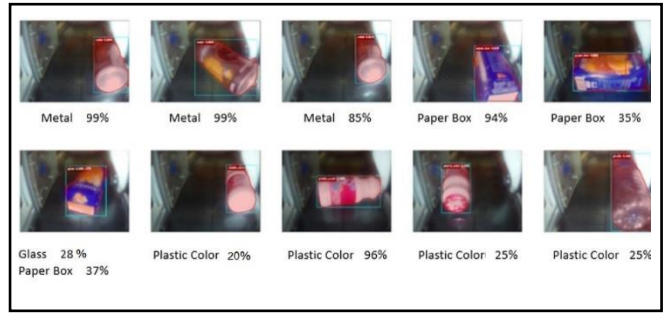


Fig. 7. 100 epochs of training with using image enhancement

In Fig. 7, after finishing 100 epochs of training, the results from the experiment show that the model can correctly identify the edges of objects. When using these weights in a real-time Smart Bin environment, the transfer learning model accurately detects the edges of garbage objects and precisely identifies their labels.

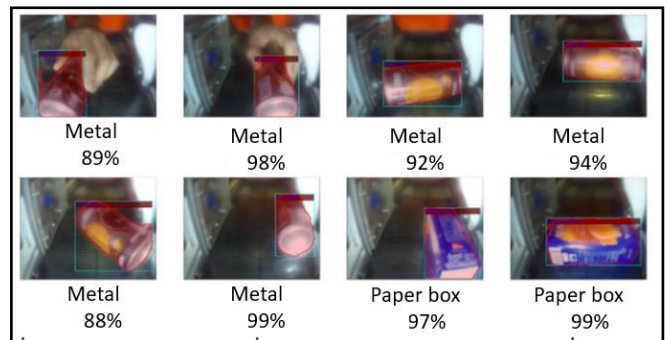


Fig. 8. 110 epochs of training with using image enhancement

In Fig. 8 shows 110 epochs of training. The results from the experiment show that with image enhancement, the model can correctly identify the edges of objects.

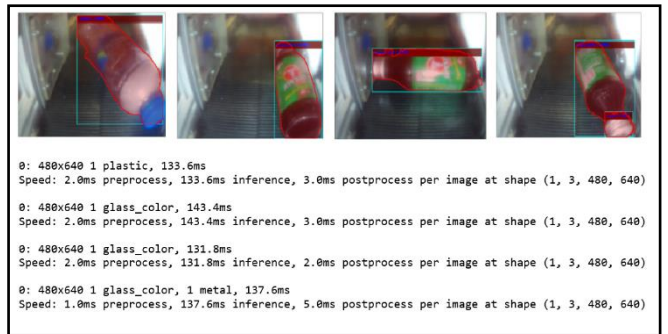


Fig. 9. Prediction result in Smart Bin environment

Fig. 9 shows Prediction result in smart bin environment. When using these weights in a real-time Smart Bin environment, the model can accurately detect the edges of garbage objects and identify labels precisely.

TABLE II. PERFORMANCE OF TRANSFER LEARNING OF YOLOV8 IN SMART BIN ENVIRONMENT

Class	Training Set	Box	R	mAP50
All	334	0.991	0.996	0.989
Glass	35	1	0.901	0.988
Glass color	55	0.99	0.982	0.995
Metal	77	0.998	0.987	0.995
Paper Box	73	0.998	0.986	0.995
Plastic	64	1	0.976	0.995
Plastic color	30	0.96	0.967	0.995

The performance of transfer learning of YOLOv8 in Smart Bin Environment. The results reveal that in the 334 dataset, the performance metrics: BOX score 0.991, R (Recall) score 0.0076 and mAP50 (Mean Average Precision with IoU 50) score 0.989 on average. These metrics indicate that the model accurately identifies both the location and label of objects in the prediction results.

IV. DISCUSSION

This section presents the experimental results. Table II presents a comparison of the performance between transfer learning model of YOLOv8n-seg. utilizing datasets with 334 sampling image. Specifically, This paper represents transfer learning with custom instance segmentation by Label Studio, The dataset with 344 images represents the dataset from TrashNet and supplemented by additional capture images sourced from the Smart Bin environment.

TABLE III. LATENCY TIME OF YOLOV8 IN SMART BIN ENVIRONMENT (RASPBERRY PI 4)

TEST NO.	PREPROCESSING	INFERENCE TIME	POST PROCESS (PREVIEW RESULT)
1	80.3	6038.2	149.8
2	27.4	4835.7	36.7
3	29.7	4896.5	2.9
4	36.6	3783.5	22.0
5	26.4	3635.7	18.8
6	38.6	3867.2	26.3
7	14.8	3602.7	21.4
8	37.0	3682.6	11.3
9	10.5	3691.5	34.3
10	37.2	3813.9	13.5

TABLE IV. TOTAL AVERAGE PREDICTION TIME

Process time	Min	Max	Average
Preprocessing	10.5	80.3	28.68
Inference Time	3602.7	6038.2	3977.7
Post Process (Preview Result)	2.9	148.8	20.8
Total Time	3616.1	6268.3	4027.18

Table III and IV shows the latency time in the prediction process of YOLOv8 in The Smart Bin environment (Raspberry Pi 4). The minimum latency time for each process are preprocessing at 10.5 ms, inference at 3602.7 ms and postprocess at 2.9 ms. The maximum latency times are: preprocessing at 80.3 ms, inference at 6038.2 ms and postprocessing at 149.8 ms. The average latency times are: preprocessing at 28.68 ms, inference at 3977.7 ms and postprocessing at 20.8 ms per predicted label. The total average prediction time ranges from approximately 3 to 6 seconds per image prediction.

V. CONCLUSION

This article proposes the development of the Smart Bin: Garbage Sorting Machine. It uses a combination of metal detection sensors and software image processing to identify objects in the Smart Bin.

Metal detection is used to detect metal directly. If the object is identified as metal, it will be transferred into the sorting area. If the object is not a metal, it will start the prediction process to identify the label of the object.

In Image detection process split into two main processes. (1) Image enhancement of objects by using kernel filters to separate objects from the Smart Bin environment. (2) Transfer learning in the YOLOv8 network is used to predict both the location and label of an object.

The combination of image enhancement and label prediction. It can accurately identify both the boundaries and label of an object in the Smart Bin environment. The Smart Bin: Garbage Sorting Machine proposes an idea to solve pollution and waste problems, as well as further ideas for developing new projects in the future.

REFERENCES

- [1] SCG Circular Way, "3-challenges-in-successful-recycling", SCG Circular Economy, Jul. 14, 2021. <https://www.scg.com/sustainability/circular-economy/interesting/3-challenges-in-successful-recycling/> (accessed Jun. 30, 2024).
- [2] www.thairath.co.th, "Waste to Energy"(online), <https://www.thairath.co.th/news/sustainable/2727929>, (accessed Jun. 30, 2024).
- [3] Joseph Redmon, Santosh Divvala, Ross Girshick and Ali Farhadi, "You Only Look Once: Unified, Real-Time Object Detection (2015)", <http://arxiv.org/abs/1506.02640>
- [4] Jocher, G., Chaurasia, A., & Qiu, J. (2023), "YOLO by Ultralytics (Version 8.0.0) [Computer software]", <https://github.com/ultralytics/ultralytics>
- [5] glenn-jocher, Laughing-q, and AyushExel, "Instance Segmentation, <https://docs.ultralytics.com/tasks/segment>", online, Created 12 November 2023
- [6] Jansi Rani S.V., Raghu Raman V., Rahul Ram M., and Prithvi Raj A. (2022), "Multi object detection and classification in solid waste management using region proposal network and YOLO model", Global NEST Journal, Vol 24(4), 743-751
- [7] Ting-Wei Wu, Hua Zhang, Wei Peng, Fan Lu, and Pin-Jing He. (2023), "Applications of convolutional neural networks for intelligent waste identification and recycling: A review", Resources, Conservation & Recycling 190 (2023) 106813
- [8] Shengping Wen, Yue Yuan and Jingfu Chen (2023), "A Vision Detection Scheme Based on Deep Learning in a Waste Plastics Sorting System", Appl Sci 2023, <https://doi.org/10.3390/app13074634>
- [9] Tkachenko, Maxim, Malyuk, Mikhail, Holmanyuk, Andrey, & Liubimov, Nikolai. (2020). "Label Studio: Data labeling software", Retrieved from <https://github.com/heartexlabs/label-studio>
- [10] Zhang Q., Yang Q., Zhang X., Bao Q., Su J., and Liu X. (2021)., "Waste image classification based on transfer learning and convolutional neural network.", Waste Management, 135, 150-157.