https://doi.org/10.20965/jaciii.2023.p0622

Research Paper:

Trash Detection Algorithm Suitable for Mobile Robots Using Improved YOLO

Ryotaro Harada, Tadahiro Oyama†, Kenji Fujimoto, Toshihiko Shimizu, Masayoshi Ozawa, Julien Samuel Amar, and Masahiko Sakai

Kobe City College of Technology
8-3 Gakuen-higashimachi, Nishi-ku, Kobe, Hyogo 651-2194, Japan
E-mail: oyama@kobe-kosen.ac.jp
†Corresponding author

[Received December 16, 2022; accepted March 24, 2023]

The illegal dumping of aluminum and plastic into cities and marine areas leads to negative impacts on the ecosystem and contributes to increased environmental pollution. Although volunteer trash pickup activities have increased in recent years, they require significant effort, time, and money. Therefore, we propose automated trash pickup robot, which incorporates autonomous movement and trash pickup arms. Although these functions have been actively developed, relatively little research has focused on trash detection. As such, we have developed a trash detection function by using deep learning models to improve the accuracy. First, we created a new trash dataset that classifies four types of trash with high illegal dumping volumes (cans, plastic bottles, cardboard, and cigarette butts). Next, we developed a new you only look once (YOLO)-based model with low parameters and computations. We trained the model on a created dataset and a dataset consisting of marine trash created during previous research. In consequence, the proposed models achieve the same detection accuracy as the existing models on both datasets, with fewer parameters and computations. Furthermore, the proposed models accelerate the edge device's frame rate.

Keywords: autonomous robot, trash detection, deep neural network, edge device, YOLO

1. Introduction

A 2020 study by the Organisation for Economic Co-operation and Development (OECD) estimated that two billion tons of urban trash were generated worldwide in 2016 [1], suggesting that an average of 270 kg of municipal trash is generated per capita. Furthermore, the OECD noted that the amount of illegally dumped trash is increasing annually.

Meanwhile, the National Oceanic and Atmospheric Administration reports that urban trash and marine debris are becoming increasingly serious concerns [2]. If this trend continues, the negative impacts of continued environmental pollution will negatively affect the ecosystem. In particular, as mentioned above, significant illegal dumping occurs in marine and city environments. The worst-case scenarios include ecological extinctions, abnormalities in the surrounding environments, and abnormal weather conditions.

Therefore, volunteer trash pickup activities have become a popular approach to solving this problem [3, 4]. However, trash pickup requires significant time and tools, as well as a great deal of effort on the part of the volunteers owing to human resources shortages. Thus, the present solutions are inadequate.

Recent studies have used robotic automation for trash pickup [5, 6]. Among the various functions expected of robots, we focus on a system for detecting trash from videos captured by robots. As trash pickup robots must adapt to various environments, we developed a trash detection system suitable for the environment in which robots operate. Although autonomous moving functions and trash collection arm control for trash pickup robots have been actively studied, the development of trash detection remains relatively understudied [7, 8]. Thus, we believe that there is scope for further development of trash detection systems.

Therefore, in this study, we propose a real-time trash detection method using the deep learning model, you only look once (YOLO) [9] to develop a technology for detecting trash from captured video. The main contributions of this study are as follows:

- We propose a new YOLO-based object detection model for real-time trash detection.
- We construct a new annotated image dataset for trash detection suitable for a robot.
- We show that the proposed models can detect trash in real time on edge devices through verification experiments using the constructed datasets.

2. Related Work

Many studies have been conducted on trash detection. For example, Proença and Simões collected images of illegally dumped trash in urban, coastal, and mountainous areas [10]. After collecting trash image data, they verified whether trash segmentation detection is feasible using a deep learning model based on a mask region-based convolutional neural network (Mask R-CNN) and evaluated its accuracy. Wang and Zhang proposed a trash detection technique based on deep learning and built a system for identifying trash in urban images with high accuracy [11]. Kulshreshtha et al. developed an autonomous mobile trash collection robot suitable for public spaces. Moreover, trash detection experiments and analysis using YOLOv4 and YOLOv4-tiny as detection models are underway [12]. Several other studies have used YOLO as a model to create trash detection systems, created various improved models, and evaluated them for accuracy [13–17].

Some studies have focused on the detection of marine trash. Valdenegro-Toro attempted to detect submerged marine trash from forward-looking sonar imagery using autonomous underwater vehicles (AUVs) [18]. To realize the detection system, they adopted convolutional neural networks (CNNs) to develop five classification models for metal, glass paper, cardboard, rubber, and plastic. They achieved 70.8% detection accuracy using a sliding window method detector. In a previous study using deep learning models, Fulton et al. created a large marine trash dataset by cropping images containing three types of objects (plastics, remotely operated vehicles (ROVs), and organisms) from videos taken from an AUV [19]. They combined this dataset with images taken from various sources to create a sizeable marine trash dataset. They also used various deep learning models to evaluate the detection accuracy (YOLOv2, Tiny-YOLO, Faster R-CNN, and a single-shot detector with MobileNetv2). In addition, they measured the frame rate on the embedded device (Jetson TX2) installed in the AUV and achieved 20.5 frames per second (FPS) on Tiny-YOLO.

As described above, several deep learning techniques have been used for urban and marine trash detection, and their accuracy has been evaluated using trash image datasets created for each study [20–22]. Many studies have also combined these deep learning techniques with edge devices [8, 14, 23].

However, prior studies have not evaluated the speed of detection or the ability to detect trash in real time with limited resources, such as when using embedded devices. In addition, the datasets proposed in prior research were suitable for specific applications, such as in assembling factory robots, distinguishing between reusable and non-reusable plastic [15], classifying biodegradable materials [16], and large-scale collection of trash by city trucks [13]. Although there are many trash datasets, those suitable for robots picking up trash in a city are relatively rare.

To overcome this problem, we proposed a fast object detection method based on YOLOv4-tiny [24] and evaluated it using an image dataset constructed for an outdoor trash pickup robot [25]. However, we found that the detection accuracy differed depending on the environment. Thus, we considered that improving trash detection was necessary to improve the robot's performance for practical use.

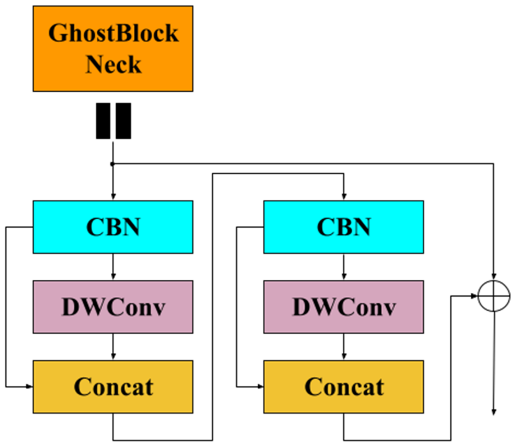


Fig. 1. Architecture of GhostBlockNeck.

Therefore, we propose a new fast object detection method based on YOLOv5-nano [26] and construct a new image dataset suitable for a trash pickup robot.

Furthermore, we apply the fast object detection method to a previous study [19] to demonstrate the model's generality.

3. Method

3.1. Proposed Model

In general, edge devices on trash pickup robots have limited memory. This limitation affects the processing speed of typical deep learning models, making real-time trash detection difficult.

Therefore, we proposed new deep learning models (YOLOv4-ghost and YOLOv5-ghost) using YOLOv4-tiny and YOLOv5-nano as the core network. The key to our method is using GhostBlockNeck (G-BN) [27] to further reduce the computational complexity. **Fig. 1** shows the structure of G-BN. In **Fig. 1**, each block represents the functions described below.

- CBN: the combination of convolutional and batch normalization layers
- DWConv: the depthwise convolutional layer
- Concat: the channel direction coupling layer

G-BN is a module that combines ordinary convolutional and linear operations. It performs a linear transformation on the generated convolutional feature maps to obtain similar features to generate higher-dimensional convolutions. This module reduces the effects of feature

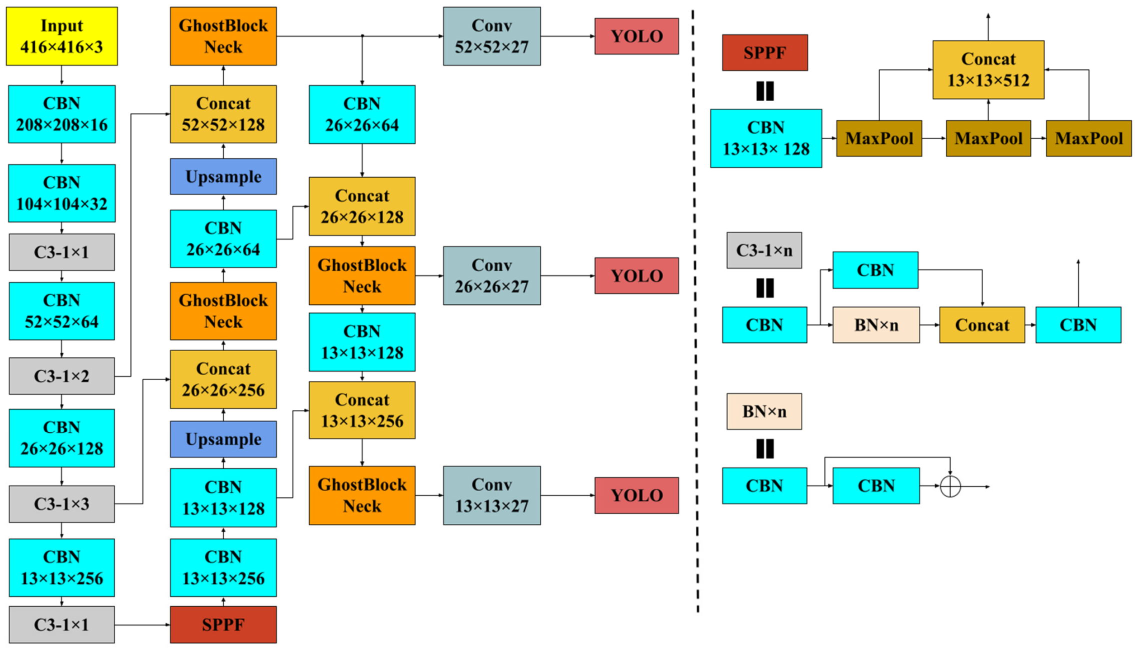


Fig. 2. Architecture of YOLOv5-ghost.

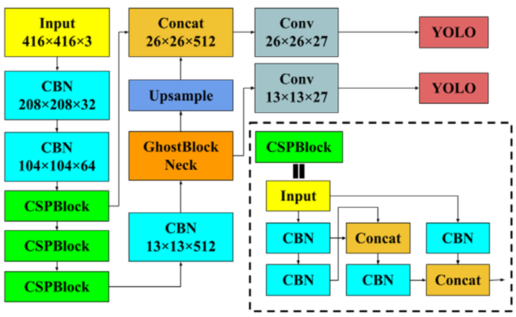


Fig. 3. Architecture of YOLOv4-ghost.

variation and parameter scaling. In addition, it can reduce the model parameters and computational complexity. This study used G-BN with one stride and depthwise convolution as a linear transformation.

The structures of YOLOv5-ghost and YOLOv4-ghost are shown in **Figs. 2** and **3**, respectively. Each block represents the functions described below.

- Conv: standard convolutional layer
- MaxPool: kernel sizes five maxpooling layer
- Upsample: upsampling layer
- SPPF: spatial pyramid pooling fast [28]
- CSPBlock: cross-stage partial [29]

C3-1 and BN$\times$n shown on the right side of **Fig. 2** represent CSPBlock with different bottlenecks.

Our approach makes the following improvements from the conventional YOLOv4-tiny and YOLOv5-nano approaches, as shown in **Figs. 2** and **3**.

- We replace the previous feature integration module Neck with G-BN.
- We change the activation functions in the CBN modules of the Neck to the sigmoid linear unit (SiLU) [30] for better accuracy.

For the proposed method G-BN, we adopt the G-BN with one stride from the previous study [27], as the training results for the stride G-BN with two strides were worse than the accuracy for the G-BN with one stride; in addition, the computational complexity increased for G-BN with two strides.

We used depthwise convolution for the linear transformation in G-BN from previous studies [27]. In addition, depthwise convolution has also been reported to reduce the computational complexity [27].

We replaced the activation function in the G-BN with the SiLU function, because the SiLU function can express a broader range of values than the rectified linear unit function [30]. Hence, we assume that this broader range of expression allows the proposed models to learn additional detailed features and lead to improved accuracy.

3.2. Building Dataset for Robots

This section elaborates on how we devised our trash dataset. Considering the countless trash types, we think it is practical to work jointly with volunteers. Therefore, in the first step, we build a dataset targeting trash discarded in large quantities in cities, as this is considered easier for the robot to pick up. This is expected to reduce the amount of trash handled by volunteers, saving volunteers' time and reducing their workload.

To develop a trash dataset suitable for the robot, we first identified four types of trash (cans, plastic bottles,

Table 1. Our dataset details.

Usage	Cans	Plastic bottles	Card-board	Cigarette butts	Back-ground
Train	2,823	2,582	1,643	3,638	49
Valid	298	941	185	545	12

(a) Our dataset (b) Trash-ICRA19

Fig. 4. Examples of training images.

Table 2. Trash-ICRA19 details.

Usage	Plastics	ROVs	Organisms
Train	4,580	1,792	1,951
Valid	853	141	70

cardboard, and cigarette butts) from publicly available datasets [10, 31–36]. In addition to obtaining similar images from these public datasets, we collected additional diverse images using a crawling module [37]. We also collected background images without trash from an existing dataset in a previous study [38], as the addition of background images has been reported to improve detection accuracy [39]. After that, we annotated the acquired images by creating a text file containing the object location and category information. Generally, the difference in the distribution of the number of objects per class in each image makes it difficult to neatly partition the dataset. Therefore, we adjusted the distribution of the number of trash images by visually checking the dataset to make them as consistent as possible. As a result, the total number of images (including background images) was 8,415. In this experiment, the images were divided into 6,732 for training and 1,683 for validation for a ratio of 8 : 2, considering the number of trash per class. **Table 1** shows a detailed breakdown of the dataset used in the experiment, and **Fig. 4(a)** shows an example of the training data images.

To testify to the performance of YOLOv4-ghost and YOLOv5-ghost for trash detection, we adopted the Trash-International Conference on Robotics and Automation (ICRA) 19 [19] dataset to train and verify the algorithm. As described in Section 2, the dataset consists of three objects culled from images taken by AUVs in the deep sea. **Table 2** shows a detailed breakdown of the dataset, and an example of a training data image is shown in **Fig. 4(b)**.

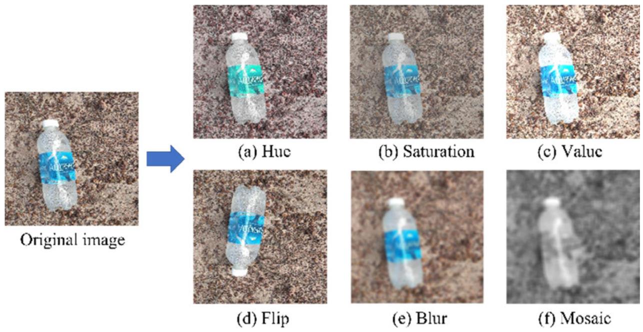


Fig. 5. Examples of data augmentation.

4. Experiment

4.1. Experiment Setup

Measuring computing performance measured in floating point operations per second (FLOPs), we selected YOLO models with of 10 GFLOPs or less in this experiment. Notably, the definition of FLOPs here refers to the total number of operations processed by the model, as edge devices tend to have limited memory. Accordingly, we used YOLOv4-tiny, YOLOv5-nano, YOLOX-nano, and YOLOX-tiny as the models in this experiment [24, 26, 40]. We used Ubuntu 20.04.3 LTS as the operating system for training and NVIDIA GeForce 2070 Super as the graphical processing unit (GPU). We set the batch size to 64, the epoch number to 475, and the training image size to 416×416. The size of the input image was set according to previous studies [15, 25]. In general, the number of model parameters and computational complexity increases proportionately to the image size, improving accuracy; however, this requires a more powerful GPU. This can lead to insufficient memory for processing the robot's autonomous movements and trash pickup arm functions. Therefore, considering the accuracy, speed, and the above reason, an input size of 416×416 was considered optimal. All models used a stochastic gradient descent optimizer with a learning rate of 0.00261, weight decay of 5×10^{-4}, and momentum of 0.9.

We adopted data augmentation during training to increase the detection accuracy. Data augmentation involves making various changes to the images in the training data and allows the models to learn a wider variety of situations; correspondingly, it has been reported to improve detection accuracy [41]. **Fig. 5** shows an example of data augmentation.

4.2. Evaluation Metrics

From previous researches [25, 42], we used the precision, recall, F1-score (F1), average precision (AP), and mean AP (mAP) for the accuracy evaluation. In Eqs. (1)–(5), TP is true positive, FP is false positive, and FN is false negative, $R_{intep}(r)$ is the function of the P–R curve, and N is the total number of object classes. We defined the intersection over union (IoU) at 50%. The IoU shown

$$\mathrm{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} =$$

Fig. 6. Concept of IoU.

in **Fig. 6** is a threshold for evaluating the degree of overlap between the correct answer bounding boxes (BBoxes) and the predicted BBox. On the right side of **Fig. 6**, filled black represents the sum and common parts between the correct answer BBox and the predicted BBox.

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}}, \quad (1)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}}, \quad (2)$$

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (3)$$

$$\text{AP} = \int_0^1 R_{intep}(r)dr, \quad (4)$$

$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N} \text{AP}_{(i)}, \quad (5)$$

$$\text{FPS} = \frac{1}{30}\sum_{i=1}^{30} \text{T}_{(i)}. \quad (6)$$

To evaluate detection processing performance, we compared the frame rate of each model. This study used the Jetson Nano and Jetson Xavier NX as the edge devices [43, 44]. To assess the performances, we randomly selected 100 images from the evaluation data of Trash-ICRA19. Next, we measured the processing detection time 30 times for each edge device. Finally, we calculated the frame rate using Eq. (6), where T represents the total time required for detection processing.

5. Results

Tables 3 and **4** show the results of the accuracy metrics for the evaluation data of our dataset and Trash-ICRA19, respectively. In **Table 3**, Param represents the number of parameters in the model. **Fig. 7** shows the detection results for the validation data. **Table 3** shows that YOLOX-nano has the lowest value in terms of FLOPs and Param, and YOLOv5-nano has the highest value in terms of mAP and F1. **Table 4** shows that YOLOv5-ghost is the highest in terms of the precision and mAP. Moreover, YOLOv5-ghost and YOLOv4-ghost achieve the highest frame rate for Jetson Nano and Jetson Xavier NX, respectively.

6. Discussion

6.1. Our Dataset

Comparing YOLOv5-nano and YOLOv5-ghost from **Table 3**, we can see that YOLOv5-ghost has significantly fewer FLOPs and parameters than YOLOv5-nano. Nevertheless, the YOLOv5-ghost result has a higher recall than YOLOv5-nano. In addition, YOLOv5-ghost achieves values similar to those from YOLOv5-nano in the AP for cans and plastic bottles and F1. We believe that redundant features could have been eliminated owing to the reduced convolution layers. Another possibility is that the use of G-BN may have resulted in improved feature integration.

By comparing YOLOv4-tiny and YOLOv4-ghost, we can see that the mAP and F1 of YOLOv4-ghost are slightly higher than those of YOLOv4-tiny, even though the number of parameters is significantly reduced. In this case, we believe that the feature propagation and number of parameters may have been appropriate for training on our dataset.

Next, comparing the APs in **Table 3**, it can be seen that all the trash achieved 85%–97.5% AP. This is likely owing to the high accuracy when defining image annotations. Moreover, the addition of a background image may have improved the AP. Therefore, we assume that increasing the total number of objects and obtaining more accurate annotations effectively improve the detection accuracy. Furthermore, to further improve the detection accuracy and increase the versatility of the model, we split the dataset and perform K-fold cross-validation. One reason for the slight bias in the APs for each type of trash was that slightly fewer cardboard box images were collected during the dataset's creation. In addition, as shown in **Fig. 8**, the dataset created in this study contained some images of deformed trash that were difficult to distinguish.

Therefore, it is no exaggeration to say that defining trash is the biggest challenge of this study. Countermeasures include conducting a questionnaire survey and asking people to visually confirm whether an object is trash. Moreover, conducting annotations with multiple people can lead to the development of an appropriate trash dataset for each area.

6.2. Trash-ICRA19

From the AP of each class in **Table 4**, we can see that the APs of organisms are much lower than those for other objects. The lower AP of organisms in all models may be because the ICRA dataset contains many definitions of organisms and a variety of marine organisms, as shown in **Fig. 9(a)**. In addition, the previous study [19] stated that the APs of organisms are significantly lower than those of other objects. Hence, it is likely that the same result was obtained in this experiment.

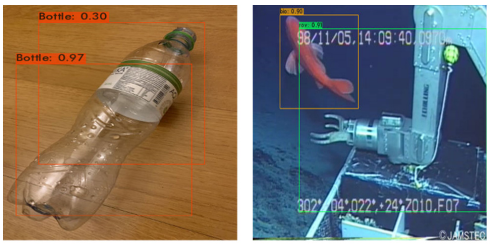
Next, comparing YOLOv4-tiny and YOLOv4-ghost, we can see that the APs of plastics and organisms are lower in YOLOv4-tiny, whereas the AP of ROVs is approximately 6% higher. This is because the ROVs in the training data are usually relatively large in the images, as

Table 3. Evaluation results using our created dataset.

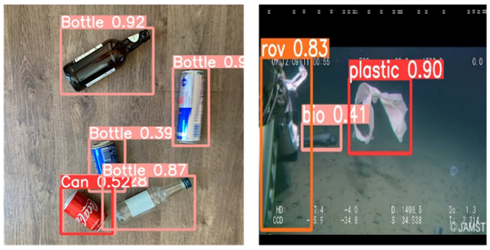
Models	Param	FLOPs	AP [%]				mAP [%]	Precision [%]	Recall [%]	F1 [%]
			Cans	Plastic bottles	Card-board	Cigarette butts				
YOLOv4-tiny	5.5M	6.8G	91.8	97.2	85.0	90.2	91.0	83.0	93.0	88.0
YOLOv5-nano	1.8M	4.2G	**94.6**	**97.5**	**90.7**	**95.8**	**94.7**	**95.7**	87.3	**91.3**
YOLOX-nano	**0.9M**	**1.1G**	90.5	95.0	87.6	93.9	91.7	68.5	95.1	79.6
YOLOX-tiny	5.0M	6.4G	93.7	95.5	89.0	95.5	93.4	54.7	**96.4**	69.8
YOLOv4-ghost	5.2M	1.5G	92.0	97.4	88.2	91.1	92.2	85.0	93.0	89.0
YOLOv5-ghost	1.5M	1.5G	94.0	96.6	87.7	91.5	92.5	90.0	92.0	91.0

Table 4. Evaluation results using the Trash-ICRA19 dataset.

Models	AP [%]			mAP [%]	Precision [%]	Recall [%]	F1 [%]	FPS	
	Plastics	ROVs	Orga-nisms					Jetson Nano	Jetson Xavier NX
YOLOv4-tiny	73.4	**56.9**	1.2	43.8	74.0	57.0	64.0	14.8	55.6
YOLOv5-nano	75.9	52.9	2.6	43.8	54.9	40.0	46.5	20.4	28.9
YOLOX-nano	72.0	56.0	0.5	42.8	14.2	**78.4**	24.0	11.6	17.5
YOLOX-tiny	71.3	55.1	0.7	42.4	12.1	76.7	20.9	13.1	19.8
YOLOv4-ghost	76.8	50.6	4.1	43.8	74.0	59.0	66.0	17.8	**60.7**
YOLOv5-ghost	**86.4**	47.1	**6.2**	**46.6**	**80.0**	68.0	**73.0**	**23.0**	24.3



(a) YOLOv4-tiny

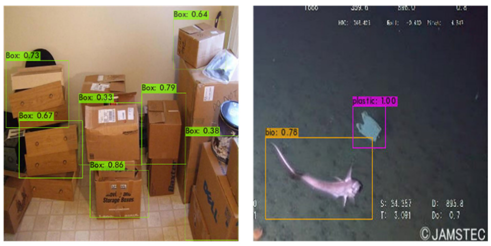


(b) YOLOv5-nano

(c) YOLOX-nano

(d) YOLOX-tiny

(e) YOLOv4-ghost

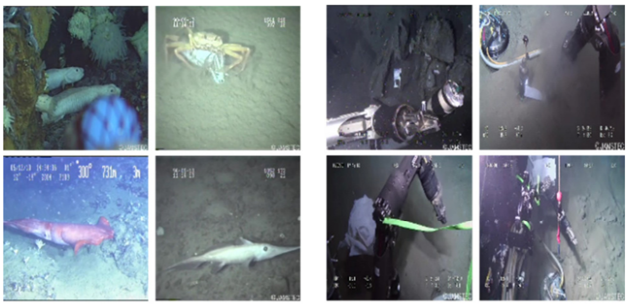


(f) YOLOv5-ghost

Fig. 7. Results of validation data of our dataset and Trash-ICRA19 using different detectors: (a) YOLOv4-tiny, (b) YOLOv5-nano, (c) YOLOX-nano, (d) YOLOX-tiny, (e) YOLOv4-ghost, and (f) YOLOv5-ghost.

(a) Cans (b) Cardboard

Fig. 8. Examples of deformed trash.



(a) Organisms (b) ROVs

Fig. 9. Examples of organisms in the Trash-ICRA19.

shown in **Fig. 9(b)**. In addition, we imagine the possibility that YOLOv4-ghost cannot extract the local features of the ROVs owing to the removal of the convolution layer near the head.

Next, comparing YOLOv5-nano with YOLOv5-ghost, we can find that the APs for plastics and organisms of YOLOv5-ghost are significantly higher. We believe that the layer replaced by G-BN can better integrate the features of the training dataset than the original layer of YOLOv5-nano. Notably, YOLOv5-ghost achieves 86.4% in the AP of plastics of **Table 4**. We think the result that the highest AP of organisms allows YOLOv5-ghost to improve precision. Furthermore, YOLOv5-ghost exceeds YOLOv5-nano in mAP and F1. Moreover, we found that the AP of plastics is approximately 3% higher than its highest value of 83.3% observed in the previous study [19].

6.3. Comparison of FPS

The FPS results in **Table 4** show that YOLOv5-ghost achieves the highest results for Jetson Nano, achieving 23.0 fps, which is 2.6 fps higher than YOLOv5-nano. Moreover, YOLOv4-ghost achieved the third highest FPS, 3.0 higher than YOLOv4-tiny. This can be attributed to the proposed models having fewer convolutional layers than the base model and using the G-BN module to reduce the computations. Hence, this scheme reduces the computational complexity.

However, YOLOv4-ghost achieved the highest value of 60.7 fps for Jetson Xavier NX, while YOLOv5-ghost resulted in a significantly lower FPS. This is because depth convolution generally consumes more memory [45]. We also found that previous research has demonstrated that the optimal model varies from edge device to edge device because the computational process varies depending on the structure of each edge device [46]. Therefore, the processing method may have required additional time, regardless of the small number of FLOPs. **Table 4** confirms that this phenomenon also occurs in YOLOX-nano. Meanwhile, YOLOv4-ghost shows a high FPS on Jetson Xavier NX despite using depth convolution. We assumed the above result was because only one G-BN module was used. Thus, the depth convolution did not significantly affect the processing speed in the case of YOLOv4-ghost. Based on these results, we must develop an optimal trash detection model for each edge device. To realize a practical trash detection system, we can quantify the model for each edge device to further reduce computation and make real-time trash detection more achievable for robots [47–49].

7. Conclusion

In this study, we developed a trash detection system that can be mounted on a robot. We used an improved model based on the deep learning model YOLO and an existing detection model. Then, learning evaluations were conducted on created and existing datasets. The results showed that the model developed in this study achieved equivalent accuracy with fewer FLOPs and parameters than those of the base model. Additionally, it was confirmed that the FPS on the edge devices was improved. In the future, we would like to increase the number of trash types, improve the model, and implement distance estimation using a depth camera [50]. Finally, we plan to adopt real-time trash detection systems for autonomous mobile robots.

References:

[1] Organisation for Economic Co-Operation and Development (OECD), "Environment at a glance 2020," OECD Publishing, 2020. https://doi.org/10.1787/4ea7d35f-en

[2] National Oceanic and Atmospheric Administration (NOAA), "What is marine debris?" https://oceanservice.noaa.gov/facts/marinedebris.html [Accessed May 3, 2021]

[3] Keep America Beautiful, "Keep America Beautiful's Volunteer Portal." https://Volunteer.kab.org [Accessed May 3, 2021]

[4] OR&R's Marine Debris Division, NOAA, "Removal." https://marinedebris.noaa.gov/our-work/removal [Accessed May 3, 2021]

[5] S. Hossain et al., "Autonomous trash collector based on object detection using deep neural network," 2019 IEEE Reg. 10 Conf. (TENCON 2019), pp. 1406-1410, 2019. https://doi.org/10.1109/TENCON.2019.8929270

[6] M. Kraft et al., "Autonomous, onboard vision-based trash and litter detection in low altitude aerial images collected by an unmanned aerial vehicle," Remote Sens., Vol.13, No.5, Article No.965, 2021. https://doi.org/10.3390/rs13050965

[7] R. Miyagusuku et al., "Toward autonomous garbage collection robots in terrains with different elevations," J. Robot. Mechatron., Vol.32, No.6, pp. 1164-1172, 2020. https://doi.org/10.20965/jrm.2020.p1164

[8] S. Gupta et al., "Gar-Bot: Garbage collecting and segregating robot," J. Phys.: Conf. Ser., Vol.1950, Article No.012023, 2021. https://doi.org/10.1088/1742-6596/1950/1/012023

[9] J. Redmon et al., "You only look once: Unified, real-time object detection," 2016 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 779-788, 2016. https://doi.org/10.1109/CVPR.2016.91

[10] P. F. Proença and P. Simões, "TACO: Trash annotations in context for litter detection," arXiv: 2003.06975, 2020. https://doi.org/10.48550/arXiv.2003.06975

[11] Y. Wang and X. Zhang, "Autonomous garbage detection for intelligent urban management," MATEC Web Conf., Vol.232, Article No.01056, 2018. https://doi.org/10.1051/matecconf/201823201056

[12] M. Kulshreshtha et al., "OATCR: Outdoor autonomous trash-collecting robot design using YOLOv4-tiny," Electronics, Vol.10, No.18, Article No.2292, 2021. https://doi.org/10.3390/electronics10182292

[13] B. D. Carolis, F. Ladogana, and N. Macchiarulo, "YOLO TrashNet: Garbage detection in video streams," 2020 IEEE Conf. Evol. Adapt. Intell. Syst. (EAIS), 2020. https://doi.org/10.1109/EAIS48028.2020.9122693

[14] Q. Chen and Q. Xiong, "Garbage classification detection based on improved YOLOV4," J. Comput. Commun., Vol.8, No.12, pp. 285-294, 2020. https://doi.org/10.4236/jcc.2020.812023

[15] L. Zhao et al., "Skip-YOLO: Domestic garbage detection using deep learning method in complex multi-scenes," Research Square, 2021. https://doi.org/10.21203/rs.3.rs-757539/v1

[16] A. Aishwarya et al., "A waste management technique to detect and separate non-biodegradable waste using machine learning and YOLO algorithm," 2021 11th Int. Conf. Cloud Comput. Data Sci. Eng. (Conflu.), pp. 443-447, 2021. https://doi.org/10.1109/Confluence51648.2021.9377163

[17] J. Xue et al., "Garbage detection using YOLOv3 in Nakanoshima Challenge," J. Robot. Mechatron., Vol.32, No.6, pp. 1200-1210, 2020. https://doi.org/10.20965/jrm.2020.p1200

[18] M. Valdenegro-Toro, "Submerged marine debris detection with autonomous underwater vehicles," 2016 Int. Conf. Robot. Autom. Humanit. Appl. (RAHA), 2016. https://doi.org/10.1109/RAHA.2016.7931907

[19] M. Fulton et al., "Robotic detection of marine litter using deep visual detection models," 2019 Int. Conf. Robot. Autom. (ICRA), pp. 5752-5758, 2019. https://doi.org/10.1109/ICRA.2019.8793975

[20] J. Hong, M. Fulton, and J. Sattar, "TrashCan: A semantically-segmented dataset towards visual detection of marine debris," arXiv: 2007.08097, 2020. https://doi.org/10.48550/arXiv.2007.08097

[21] H. Panwar et al., "AquaVision: Automating the detection of waste in water bodies using deep transfer learning," Case Stud. Chem. Environ. Eng., Vol.2, Article No.100026, 2020. https://doi.org/10.1016/j.cscee.2020.100026

[22] C. Wu et al., "Underwater trash detection algorithm based on improved YOLOv5s," J. Real-Time Image Process., Vol.19, No.5, pp. 911-920, 2022. https://doi.org/10.1007/s11554-022-01232-0

[23] X. Ma et al., "Light-YOLOv4: An edge-device oriented target detection method for remote sensing images," IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., Vol.14, pp. 10808-10820, 2021. https://doi.org/10.1109/JSTARS.2021.3120009

[24] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Scaled-YOLOv4: Scaling cross stage partial network," 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 13024-13033, 2021. https://doi.org/10.1109/CVPR46437.2021.01283

[25] R. Harada et al., "Development of an AI-based illegal dumping trash detection system," Artif. Intell. Data Sci., Vol.3, No.3, pp. 1-9, 2022. https://doi.org/10.11532/jsceiii.3.3_1

[26] G. Jocher et al., "ultralytics/yolov5: v6.1 - TensorRT, TensorFlow Edge TPU and OpenVINO export and inference," Zenodo, 2022. https://doi.org/10.5281/ZENODO.6222936

[27] K. Han et al., "GhostNet: More features from cheap operations," 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 1577-1586, 2020. https://doi.org/10.1109/CVPR42600.2020.00165

[28] F. Meslet-Millet, E. Chaput, and S. Mouysset, "SPPNet: An approach for real-time encrypted traffic classification using deep learning," 2021 IEEE Glob. Commun. Conf. (GLOBECOM), 2021. https://doi.org/10.1109/GLOBECOM46510.2021.9686037

[29] C.-Y. Wang et al., "CSPNet: A new backbone that can enhance learning capability of CNN," 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), pp. 1571-1580, 2020. https://doi.org/10.1109/CVPRW50498.2020.00203

[30] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," Neural Netw., Vol.107, pp. 3-11, 2018. https://doi.org/10.1016/j.neunet.2017.12.012

[31] A. Kuznetsova et al., "The Open Images Dataset V4," Int. J. Comput. Vis., Vol.128, No.7, pp. 1956-1981, 2020. https://doi.org/10.1007/s11263-020-01316-z

[32] I. Krasin et al., "OpenImages: A public dataset for large-scale multi-label and multi-class image classification," 2017. https://storage.googleapis.com/openimages/web/index.html [Accessed June 6, 2021]

[33] S. Sekar, "Waste Classification Data," Kaggle. https://www.kaggle.com/datasets/techsash/waste-classification-data?resource=download [Accessed June 6, 2021]

[34] A. Serezhkin, "Drinking Waste Classification," Kaggle. https://www.kaggle.com/datasets/arkadiyhacks/drinking-waste-classification [Accessed June 6, 2021]

[35] M. Yang and G. Thung, "Classification of trash for recyclability status," Stanford University, 2016.

[36] Immersive Limit LLC, "Cigarette Butt Dataset." https://www.immersivelimit.com/datasets/cigarette-butts [Accessed June 6, 2021]

[37] E. Uzun et al., "An effective and efficient Web content extractor for optimizing the crawling process," Softw.: Pract. Exp., Vol.44, No.10, pp. 1181-1199, 2014. https://doi.org/10.1002/spe.2195

[38] B. Zhou et al., "Places: A 10 million image database for scene recognition," IEEE Trans. Pattern Anal. Mach. Intell., Vol.40, No.6, pp. 1452-1464, 2018. https://doi.org/10.1109/TPAMI.2017.2723009

[39] G. Qin, B. Vrusias, and L. Gillam, "Background filtering for improving of object detection in images," 2010 20th Int. Conf. Pattern Recognit., pp. 922-925, 2010. https://doi.org/10.1109/ICPR.2010.231

[40] Z. Ge et al., "YOLOX: Exceeding YOLO series in 2021," arXiv: 2107.08430, 2021. https://doi.org/10.48550/arXiv.2107.08430

[41] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," J. Big Data, Vol.6, No.1, Article No.60, 2019. https://doi.org/10.1186/s40537-019-0197-0

[42] R. Padilla et al., "A survey on performance metrics for object-detection algorithms," 2020 Int. Conf. Syst. Signals Image Process. (IWSSIP), pp. 237-242, 2020. https://doi.org/10.1109/IWSSIP48289.2020.9145130

[43] D. Franklin, "Jetson Nano brings AI computing to everyone," NVIDIA Technical Blog, 2019. https://developer.nvidia.com/blog/jetson-nano-ai-computing/ [Accessed November 6, 2022]

[44] D. Franklin, "Introducing Jetson Xavier NX, the world's smallest AI supercomputer," NVIDIA Technical Blog, 2019. https://developer.nvidia.com/blog/jetson-xavier-nx-the-worlds-smallest-ai-supercomputer/ [Accessed November 6, 2022]

[45] Z. Qin et al., "Diagonalwise refactorization: An efficient training method for depthwise convolutions," 2018 Int. Jt. Conf. Neural Netw. (IJCNN), 2018. https://doi.org/10.1109/IJCNN.2018.8489312

[46] P. Zhang, E. Lo, and B. Lu, "High performance depthwise and pointwise convolutions on mobile devices," 34th AAAI Conf. Artif. Intell. (AAAI-20), pp. 6795-6802, 2020.

[47] ONNX, "Open Neural Network Exchange." https://onnx.ai/ [Accessed December 2, 2022]

[48] H. Vanholder, "Efficient inference with TensorRT." https://on-demand.gputechconf.com/gtc-eu/2017/presentation/23425-han-vanholder-efficient-inference-with-tensorrt.pdf [Accessed December 2, 2022]

[49] R. David et al., "TensorFlow Lite Micro: Embedded machine learning on TinyML systems," arXiv: 2010.08678, 2020. https://doi.org/10.48550/arXiv.2010.08678

[50] Intel, "Intel® RealSense™ Depth Camera D435i." https://www.intelrealsense.com/depth-camera-d435i/ [Accessed December 2, 2022]

Name:
Ryotaro Harada

Affiliation:
Division of Electrical, Electronic and Infocommunications Engineering, Graduate School of Engineering, Osaka University

Address:
2-1 Yamadaoka, Suita, Osaka 565-0871, Japan

Brief Biographical History:
2016- Department of Electronics, Kobe City College of Technology
2021-2023 Major of Electrical and Electronic Engineering, Kobe City College of Technology
2023- Division of Electrical, Electronic and Infocommunications Engineering, Graduate School of Engineering, Osaka University

Main Works:
- "Development of communication system from EMG of suprahyoid muscles using deep learning," 2022 IEEE 4th Glob. Conf. Life Sci. Technol. (LifeTech), pp. 5-9, 2022.
- "Development of an AI-based illegal dumping trash detection system," Artif. Intell. Data Sci., Vol.3, No.3, pp. 1-9, 2022.

Membership in Academic Societies:
- The Institute of Electronics, Information and Communication Engineers (IEICE)

Name:
Tadahiro Oyama

Affiliation:
Associate Professor, Department of Electronic Engineering, Kobe City College of Technology

Address:
8-3 Gakuen-higashimachi, Nishi-ku, Kobe, Hyogo 651-2194, Japan

Brief Biographical History:
2010- Department of Electronic Engineering, Kobe City College of Technology

Main Works:
- "Development of an AI-based illegal dumping trash detection system," Artif. Intell. Data Sci., Vol.3, No.3, pp. 1-9, 2022.
- "Development of communication system from EMG of suprahyoid muscles using deep learning," 2022 IEEE 4th Glob. Conf. Life Sci. Technol. (LifeTech), pp. 5-9, 2022.
- "Estimation of tongue motion and vowels of silent speech based on EMG from suprahyoid muscles using CNN," IEEJ Trans. Electron. Inf. Syst., Vol.138, No.7, pp. 828-837, 2018.

Membership in Academic Societies:
- The Institute of Electronics, Information and Communication Engineers (IEICE)
- The Institute of Electrical Engineers of Japan (IEEJ)

Name:
Kenji Fujimoto

ORCID:
0009-0001-6663-6840

Affiliation:
Professor, Department of Electronics, Kobe City College of Technology

Address:
8-3 Gakuen-higashimachi, Nishi-ku, Kobe, Hyogo 651-2194, Japan

Brief Biographical History:
2003- Department of Electronics, Kobe City College of Technology

Main Works:
- "A Study on Anomaly Detection in Industrial Parts," Res. Rep. Kobe City Coll. Technol., No.60, pp. 39-44, 2022 (in Japanese).

Membership in Academic Societies:
- The Institute of Electronics, Information and Communication Engineers (IEICE)
- Japanese Society for Fracture Repair (JSFR)

Name:
Toshihiko Shimizu

Affiliation:
Associate Professor, Department of Mechanical Engineering, Kobe City College of Technology

Address:
8-3 Gakuen-higashimachi, Nishi-ku, Kobe, Hyogo 651-2194, Japan

Brief Biographical History:
2010- Visiting Researcher, Italian Institute of Technology (IIT)
2012- JSPS Research Fellow
2013- Kobe City College of Technology

Main Works:
- "A mobile dual-arm manipulation robot system for stocking and disposing of items in a convenience store by using universal vacuum grippers for grasping items," Adv. Robot., Vol.34, Nos.3-4, pp. 219-234, 2020.
- "Q-bot: Heavy object carriage robot for in-house logistics based on universal vacuum gripper," Adv. Robot., Vol.34, Nos.3-4, pp. 173-188, 2020.
- "Development of universal vacuum gripper for wall-climbing robot," Adv. Robot., Vol.32, No.6, pp. 283-296, 2018.

Membership in Academic Societies:
- The Robotics Society of Japan (RSJ)
- The Japan Society of Mechanical Engineers (JSME)
- The Society of Instrument and Control Engineers (SICE)

Name:
Masayoshi Ozawa

Affiliation:
Associate Professor, Department of Mechanical Engineering, Kobe City College of Technology

Address:
8-3 Gakuen-higashimachi, Nishi-ku, Kobe, Hyogo 651-2194, Japan

Brief Biographical History:
2019- Kobe City College of Technology.

Main Works:
• "Verification of communication characteristics for utilization of radio communication device in undersea exploration equipment," J. Robot. Soc. Jpn., Vol.37, No.6, pp. 507-513, 2019.

Membership in Academic Societies:
• The Robotics Society of Japan (RSJ)
• The Japan Society of Mechanical Engineers (JSME)
• Marine Technology Society

Name:
Julien Samuel Amar

Affiliation:
Associate Professor / Lecturer, Department of Mechanical Engineering, Kobe City College of Technology

Address:
8-3 Gakuen-higashimachi, Nishi-ku, Kobe, Hyogo 651-2194, Japan

Brief Biographical History:
2016- Ph.D. Student in Robotic Design and Analysis, Wakayama University
2021- Kobe City College of Technology

Main Works:
• "A unified framework for dynamics and control of tree-type systems using exponential coordinates," Mech. Syst. Signal Process., Vol.131, pp. 446-468, 2019.
• "Genetic-algorithm-based global design optimization of tree-type robotic systems involving exponential coordinates," Mech. Syst. Signal Process., Vol.156, Article No.107461, 2021.

Membership in Academic Societies:
• The Japan Society of Mechanical Engineers (JSME)

Name:
Masahiko Sakai

Affiliation:
Assistant Professor, Department of Electrical Engineering, Kobe City College of Technology

Address:
8-3 Gakuen-higashimachi, Nishi-ku, Kobe, Hyogo 651-2194, Japan

Brief Biographical History:
2013-2015 Kawasaki Heavy Industries, Ltd.
2021- Kobe City College of Technology

Main Works:
• "Characteristics analysis of a helical teethed linear actuator," Int. J. Appl. Electromagn. Mech., Vol.52, Nos.1-2, pp. 571-578, 2016.
• "Force estimation method for a magnetic lead-screw-driven linear actuator," IEEE Trans. Magn., Vol.54, No.11, Article No.8207805, 2018.

Membership in Academic Societies:
• The Institute of Electrical Engineers of Japan (IEEJ)
• The Japan Society of Mechanical Engineers (JSME)
• The Japan Society of Applied Electromagnetics and Mechanics (JSAEM)