J Inf Process Syst, Vol.18, No.4, pp.524~534, August 2022
https://doi.org/10.3745/JIPS.02.0178

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

# TOD: Trash Object Detection Dataset

Min-Seok Jo[1], Seong-Soo Han[2], and Chang-Sung Jeong[1,*]

**Abstract**

In this paper, we produce Trash Object Detection (TOD) dataset to solve trash detection problems. A well-organized dataset of sufficient size is essential to train object detection models and apply them to specific tasks. However, existing trash datasets have only a few hundred images, which are not sufficient to train deep neural networks. Most datasets are classification datasets that simply classify categories without location information. In addition, existing datasets differ from the actual guidelines for separating and discharging recyclables because the category definition is primarily the shape of the object. To address these issues, we build and experiment with trash datasets larger than conventional trash datasets and have more than twice the resolution. It was intended for general household goods. And annotated based on guidelines for separating and discharging recyclables from the Ministry of Environment. Our dataset has 10 categories, and around 33K objects were annotated for around 5K images with 1280×720 resolution. The dataset, as well as the pre-trained models, have been released at https://github.com/jms0923/tod.

**Keywords**

Dataset, Deep Learning, Recognition, Trash Detection

## 1. Introduction

In the aftermath of coronavirus disease 2019 (COVID-19), the use of online shopping and delivery is increasing rapidly, so disposable waste emissions are increased too. If the separation is not properly collected, it can't be collected or recycled again at the staging area and is incinerated or reclaimed. This results in large incidental costs, accompanied by environmental degradation.

Therefore, research has emerged to automate separation collection using artificial intelligence or to determine separation collection items instead. To automate, the problem of trash recognition, which determines the location, item and material is an essential issue underlying automation. All studies using artificial intelligence are possible only with sufficient datasets. However, regarding the problem of trash recognition, there is still a lack of quality datasets.

Therefore, to create a trash dataset that can be used for object detection model training using deep learning, a large-capacity, high-resolution trash dataset is produced to solve the object detection problem. It is intended for general household goods. Also, Our dataset is classified based on guidelines for separating and discharging recyclables from the Ministry of Environment.

The dataset has 10 categories, and 33K objects are annotated for 5K images of 1280×720 resolution.

Manuscript received June 2, 2021; first revision August 12, 2021; accepted September 12, 2021.

* **Corresponding Author:** Chang-Sung Jeong (csjeong@korea.ac.kr)

[1] Dept. of Electrical and Engineering, Korea University, Seoul, Korea (jms0923@korea.or.kr, csjeong@korea.ac.kr)

[2] Division of Liberal Studies, Kangwon National University, Samcheok, Korea (sshan1@kangwon.ac.kr)

The paper is structured as follows, In Section 2, we review related work and object detection network applications. In Section 3, we propose a dataset and describe it in detail. In Section 4, we implement some deep learning models and train them to experiment with our dataset. And analyze the results of the experiment. Finally, in Section 5, we summarize our research and conclude the paper.

# 2. Related Work

Dataset is a crucial factor in deep learning techniques. It is difficult to determine the optimal category, resolution, and size. The results depend on how the dataset is constructed through these values. In this section, we will look at the dataset studies related to waste detection.

## 2.1 Trash Datasets

Existing trash dataset-related studies mostly focus on the classification problem and consist of specifying one category on one image. Furthermore, since its size is small and limited, it is not sufficient for learning deep learning models. There are also no unified standards for determining categories. Therefore, each dataset is different. Therefore, sufficient-scale datasets are needed for object detection models that can detect and classify recyclables. We show detailed information about existing datasets in Table 1 [1-6].

**Table 1.** Summary of trash datasets

| Study | Category | Image | Annotation | Format | Resolution |
|---|---|---|---|---|---|
| Sakr et al. [1] | 3 | 2,000 | N/A | Classification | 256×256 |
| Chu et al. [2] | 4 | 5,000 | N/A | Classification | 240×240 |
| Liao [3] | 240 | 17,690 | N/A | Classification | 320×240 |
| Zhang et al. [4] | 9 | 681 | N/A | Bounding Box | 420×400 |
| Mittal et al. [5] | 2 | 450 | N/A | Bounding Box | 256×256 |
| Rad et al. [6] | 25 | 469 | 4,338 | Bounding Box | 640×480 |
| Proposed | 10 | 4,977 | 33,434 | Bounding Box | 1280×720 |

In the case of [3], the dataset for classification was collected and classified through crawling based on Shanghai's policy. In [4] and [6], a dataset for object detection was constructed and experimented with garbage on city streets. In [5], garbage images in the city were collected through crawling and only the coordinates were annotated. In addition, it is possible to share geo-tags as well as detect trash through smartphones.

## 2.2 Object Detection Models

Object detection models are largely divided into 2-stage and 1-stage models. The 2-stage model performs localization and classification sequentially, whereas the 1-stage model performs localization and classification simultaneously. Generally, a 2-stage model is slower than a 1-stage model but more accurate.

One of the 1-stage models is Single Shot Multibox Detector (SSD) [7]. SSD uses Visual Geometry

Group (VGG) [8] as its backbone network. It extracts features from VGG and passes them through convolution layers from features extracted, producing six multi-feature maps of different sizes. For each feature map, detection is performed using a bounding box of a predetermined size and ratio. Among the six multi-feature maps, a small object is detected in a large feature map, and a large object is detected in a small feature map. There are 8,732 bounding boxes predicted in the multi-function map, and intersection-over-union (IoU) and non-maximum suppression (NMS) are applied to the bounding boxes to produce the final results.

The other 1-stage model is You Look Only Once version 3 (YOLOv3) [9]. The YOLOv3 is an object detection model that was further developed using YOLOv1 [10] and YOLOv2 [11] as its baseline. YOLOv3 changed its backbone network from DarkNet-19 to DarkNet-53 to improve accuracy. For the classification loss, softmax, which is widely used in multi-label classification, was used in the past, but it was changed to a logistic loss to improve accuracy and to learn well even on complex datasets. In addition, they made a model robust to the scale by predicting the bounding box at three different scales.

As a 2-stage model, there is the Cascade R-CNN [12]. Cascade R-CNN is an object detection model developed using Faster R-CNN [13] as its baseline. While Faster R-CNN used only one classifier, Cascade R-CNN used several classifiers sequentially. When training, higher IoU values are used for later classifiers. This method improves the accuracy of the model.

### 2.3 Object Detection Application

Some studies have used object detection models to solve real-world problems. Convolution neural networks (CNNs) demonstrated superior performance in recognizing image-based objects compared with other algorithms, and their broad tolerance was demonstrated in many fields.

Kim et al. [14] have overcome the shortcomings of one of the ensemble techniques, AdaBoost, with fuzzy theory. They noted that AdaBoost typically gives the same weight to the first round of the boosting process. They solved these problems by allocating distributed initial values through fuzzy theory based on the statistical features of the data. Zhu et al. [15] proposed a novel face recognition model based on game theory for call-over in the classroom. They increased face recognition performance and lowered the error rates. Yadav et al. [16] improved the Hindi language recognition model, which had previously low recognition rates. They proposed three feature extraction techniques; histogram of projection based on mean distance, histogram of projection based on pixel value, and vertical zero crossing. These techniques are also influential for extracting features of distorted characters as well. Wang and Yagi [17] used shadows that were traditionally treated as noise for pedestrian detection. They used shadow and motion information simultaneously to increase the recognition rate for other scenes. The method that they suggested was to detect pedestrians using shape information of shadow regions and appearance. Wan Zaki et al. [18] proposed a real-time moving object detection model divided into naming, modeling, matching, and subtraction modules. They reduced pulse-negative and true negative ratios and improved the accuracy by 98%.

## 3. Dataset Description

In this section, we will discuss the dataset we have constructed.

## 3.1 Data Acquisition

In our dataset, three or more trashes in a single image, and on average, a piece of trash is made up of one or more objects. An object can be composed of several substances. Objects placed on the floor were photographed at a height of about 160 cm. The target object was kept in its original shape or arbitrarily transformed, such as crumpling or tearing. The angle between the camera and the ground was maintained at around 90º, and the direction and position of the object looking at the camera were randomly changed. The brightness was controlled in two stages using lighting.

## 3.2 Object Categories

The purpose was to detect and classify recyclable household items, so the material was set as an important criterion. Therefore, categories were set based on the material as paper, can, bottle, pet, plastic, and vinyl were set. In the case of paper, it corresponds to general books such as books and magazines, boxes for packaging, and paper cups have very different appearances, so it can be learned more qualified features due to CNN-based on computer vision. It was determined as a different material so that labels and caps were added to the category since it is a separate waste collection.

In the case of label categories like Nutrition Facts, the material is mostly made of paper and vinyl. Paper labels are generally used for glass bottles and cans, and vinyl labels are often used for PET bottles. It was considered that two categories could not be accurately judged by only visual judgment used in deep learning without classification mark information, and thus, it was unified into one label category.

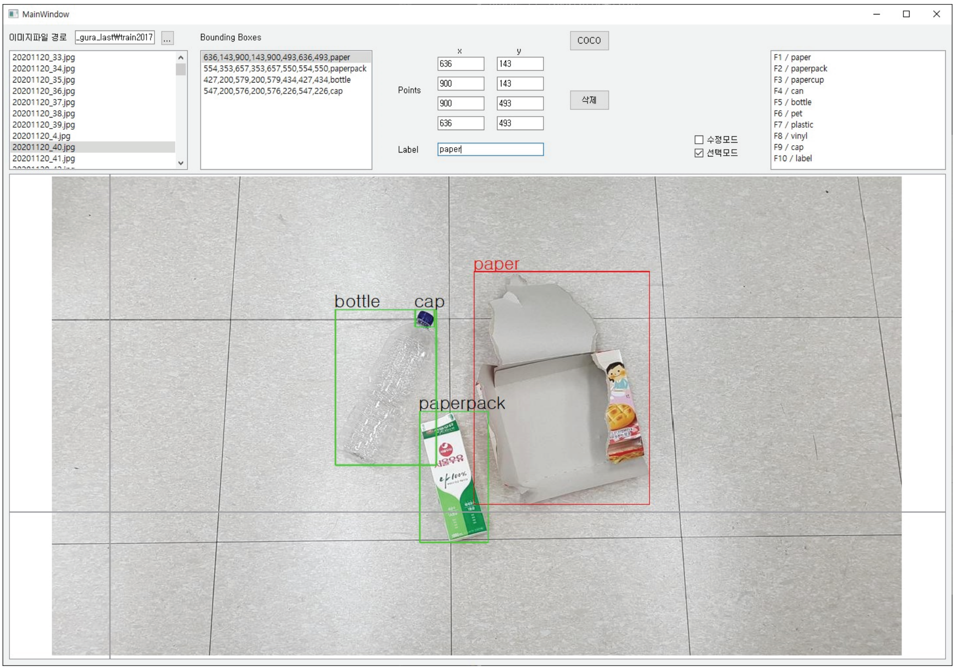
In the case of caps, there are lids made of cans, which are often used for glass bottles, and lids made of plastic, which are often used for PET bottles. However, the cap occupied less area in the entire image compared to other categories, so it was considered that it could not be visually judged like the label. Therefore, it was composed of one cap. Finally, 10 categories were composed of paper, paper pack, paper cup, can, bottle, PET, plastic, vinyl, cap, and label.

## 3.3 Annotations

Like other object detection-related studies, our datasets must also exist in a form that can be used in deep learning models. For a single image, there should be data on the bounding box, the coordinates of the target object you want to learn, and what category the object is in that area. The format of the dataset is the same as that of the COCO [19] dataset format. We made a tool to perform an annotation. An image of our annotation tool is in Fig. 1.

## 3.4 Pre-processing

Most of the images were composed of various resolutions of 2560×1440 (QHD) or higher, such as 2024×2024 and 3024×4032. To create a dataset optimized for deep learning, the image was resized collectively to a resolution of 1280×720 (HD), which satisfies the large input size of the latest deep learning models and does not break the image significantly. Secondly, it was unified into a 3-channel JPG format to be used as an input for a deep learning model. Third, by randomly performing augmentation during training, it was possible to learn about various data while maintaining the total number of images.



이미지파일 경로 _gura_last₩train2017

20201120_33.jpg
20201120_34.jpg
20201120_35.jpg
20201120_36.jpg
20201120_37.jpg
20201120_38.jpg
20201120_39.jpg
20201120_4.jpg
20201120_40.jpg
20201120_41.jpg

Bounding Boxes

636,143,900,143,900,493,636,493,paper
554,353,657,353,657,550,554,550,paperpack
427,200,579,200,579,434,427,434,bottle
547,200,576,200,576,226,547,226,cap

| | x | y |
|---|---|---|
| Points | 636 | 143 |
| | 900 | 143 |
| | 900 | 493 |
| | 636 | 493 |

Label paper

□ 수정모드
☑ 선택모드

F1 / paper
F2 / paperpack
F3 / papercup
F4 / can
F5 / bottle
F6 / pet
F7 / plastic
F8 / vinyl
F9 / cap
F10 / label

**Fig. 1.** Annotation tool.

## 3.5 Statistics

Based on our proposed method described in Sections 3.1 and 3.3, all images are annotated in our Trash Object dataset. Around 5K images were composed of 33K annotated data. It is the largest among the trash datasets that can be used for object detection models. The resolution is also the largest and it has sufficient size to train deep learning models.

To train deep learning models and validate them, we split our dataset into a train, validation, and test set. The split of train, validation, and test sets is given in Table 2. About 76% of all images, 3.8K images, were split into a train set. About 15% of all images were split into a validation set and about 8% of the remaining images were split into a test set.

The number of Annotated data for each category is in Fig. 2. Cap was the highest with 7K, followed by paper and label with about 4.6K. It was followed by approximately 3.5K cans and approximately 2.9K plastics and pieces of vinyl. There were 2.4K, 2.2K, and 1.6K bottles, respectively. Paper cups were the fewest with about 1.5K. There was an average of 6.71 annotated data per image.

Anchor-based models project anchors for each pixel of the extracted feature to find the location and size of the object. The anchor defines the size and ratio based on the dataset in advance. Therefore, knowing the size and ratio of each annotated data is important for performance improvement. In our dataset, the correlation between regions and ratios of the annotated data is in Fig. 3. The aspect ratio values are plotted against the area. The aspect ratio of annotated data is the most common, around 1. In the case of annotated data with an area of more than 50000 pixels, most of them were around 1 and converged to 1 as the area grew larger. Conversely, the smaller the area, the larger the aspect ratio. Most of the annotated data below 25000 pixels had an aspect ratio between 1 and 5. Some data have an aspect ratio of more than 10.

**Table 2.** Split of our dataset

| | Train | Validation | Test | All |
|---|---|---|---|---|
| Images | 3,807 | 747 | 423 | 4,977 |
| Annotations | 25,519 | 5,058 | 2,857 | 33,434 |

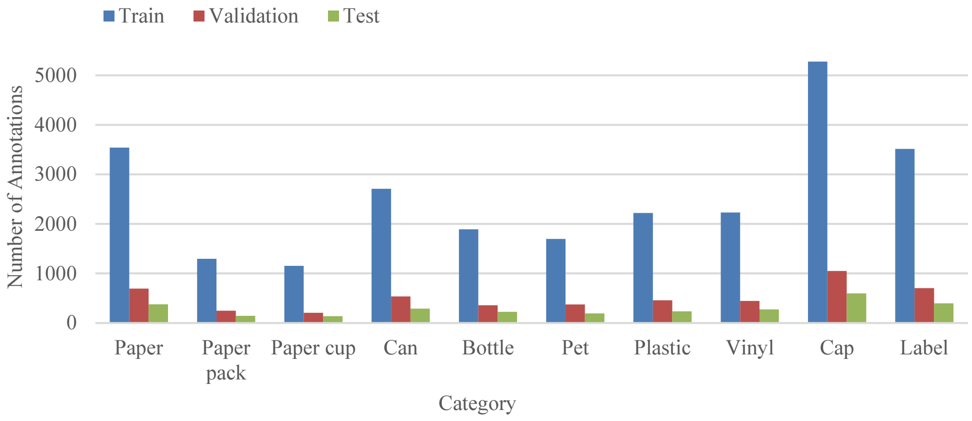


**Fig. 2.** Distribution of annotation for each category.

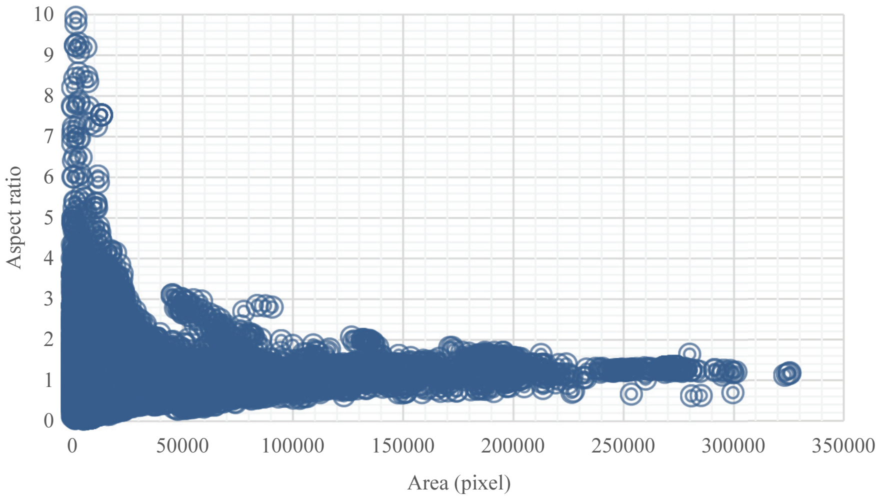


**Fig. 3.** The bounding box area against the bounding box ratio.

Area information for each category is in Fig. 4. Paper had the largest area, followed by vinyl, pet, plastic, paper pack, bottle, paper cup, can, label and cap. Because paper is generally thin and wide, it has the largest area in the image. The area of the paper was approximately 70000 pixels, more than twice as large as the area of all other categories. The area of vinyl was about 34000 pixels, about half of the area of the paper. Pet, plastic, and paper pack have areas of more than 10000 and less than 20000 pixels. Bottle, paper cup, and can are about 9000 pixels, similar in size to each other. The smallest category in Area is cap, which is about 1000 pixels, four times smaller than the next smallest area. The average area of our dataset is about 180000 pixels.

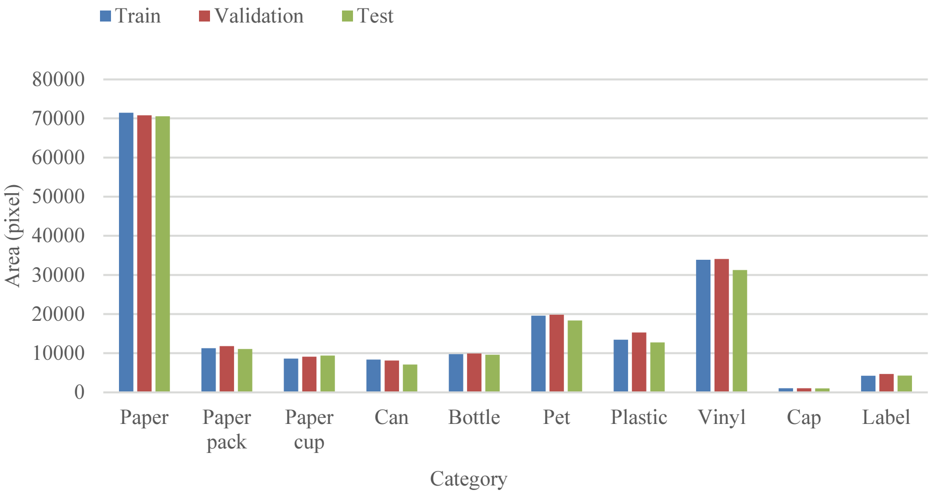


**Fig. 4.** Area average for each category.

# 4. Experiments

We tested our dataset on some object detection models. Three models with different backbones were used; SSD, YOLOv3, and Cascade R-CNN. The SSD used the VGG as the backbone. YOLOv3 used DarkNet-53 and the Cascade R-CNN used ResNet-50 [20] as the backbone.

## 4.1 Implementation

Each model was trained using the train set in Table 2 for 300 epochs using three v100s, and the epoch with the best performance for the validation set was used. Only one v100 was used for inference. Average precision (AP) was used as the evaluation metric.

The dataset was resized to the appropriate input size for each model and padded appropriately. Each model used transfer learning, which was trained in advance on the COCO dataset and learned on our dataset. During the learning period, several augmentation techniques were applied. Horizontal flip, vertical flip, hue, saturation, contrast, brightness, and center crop were applied. In addition, hyper-parameters such as running rate and optimizer used the settings of the original paper as they were.

## 4.2 Results

Table 3 allows us to compare the AP performance of the models for each category. By category, the “can” category had the highest average value of 0.578 and the lowest cap at 0.316 on average. Can category was composed of good data to extract features from the model's perspective, even though the amount of annotated data was average, while the “cap” category had the largest amount of annotated data, but had a smaller footprint and ambiguous criteria for extracting features from the model's perspective. In Fig. 5, we show some detection examples on our dataset with SSD, YOLOv3, and Cascade R-CNN.

**Table 3.** AP performance by category of each model

| Category | SSD | YOLOv3 | Cascade R-CNN | Average |
|---|---|---|---|---|
| Paper | 0.314 | 0.536 | 0.33 | 0.393 |
| Paper pack | 0.335 | 0.502 | 0.387 | 0.408 |
| Paper cup | 0.282 | 0.569 | 0.327 | 0.393 |
| Can | 0.44 | 0.595 | 0.698 | 0.578 |
| Bottle | 0.392 | 0.488 | 0.604 | 0.495 |
| Pet | 0.337 | 0.366 | 0.327 | 0.343 |
| Plastic | 0.294 | 0.314 | 0.436 | 0.348 |
| Vinyl | 0.289 | 0.405 | 0.404 | 0.366 |
| Cap | 0.26 | 0.285 | 0.404 | 0.316 |
| Label | 0.164 | 0.395 | 0.406 | 0.322 |

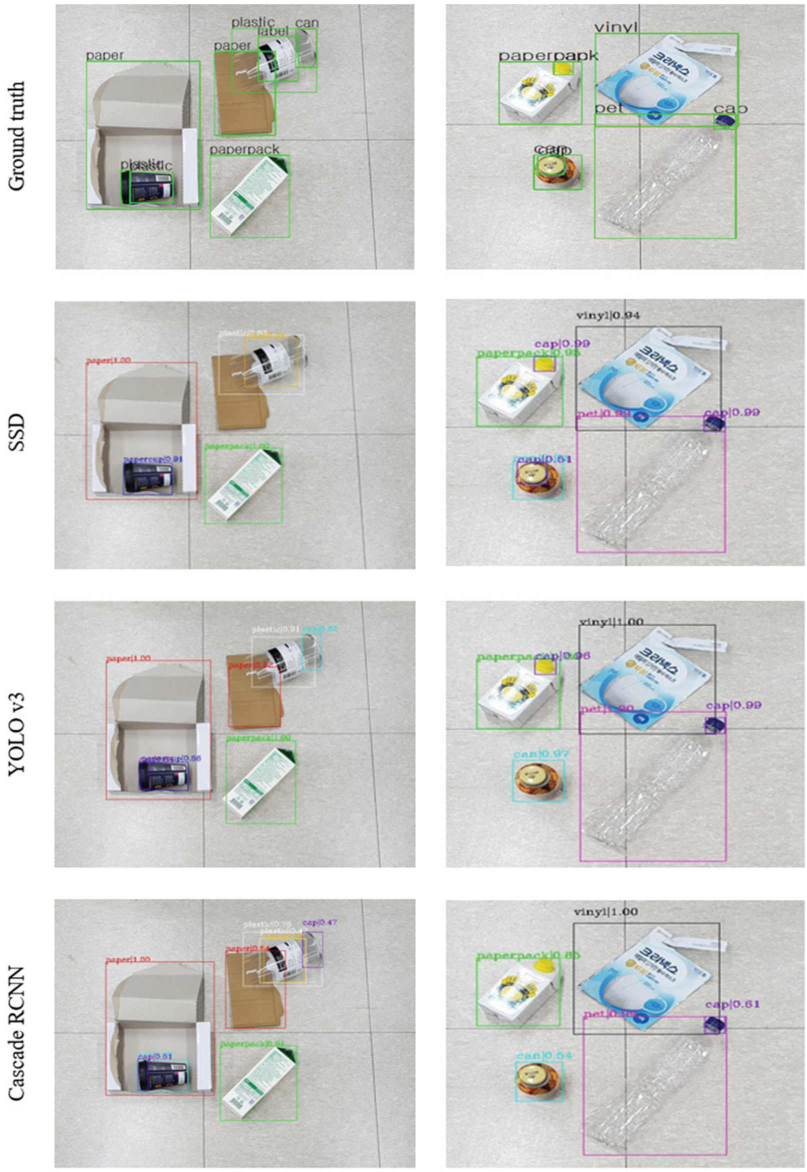


**Fig. 5.** Cropped images of the correct dataset and the inference results for each model.

## 5. Conclusion and Future Work

We have collected images of general household waste. The number of images is approximately 5K, and the annotated data is 33K. Our dataset is created by annotating the images so that they can be used for deep learning. For separate waste collection, categories are created and classified based on the material. The categories are selected by the Ministry of Environment's separate waste collection criteria: paper, paper pack, paper cup, can, bottle, pet, plastic, vinyl, cap, and label. We analyze our dataset as a perspective for training deep learning models. To experiment with our dataset, we implement trash detection networks using three different deep learning models; SSD, YOLOv3, and Cascade R-CNN. We trained those models using our dataset. As a result, “can” has the best detection value, and “cap” is the most difficult to detect. For future work, more detailed detection is needed not only for the 10 categories of general household waste presented in this paper, but also for other types of waste such as metal, wood, and food. In addition, more data that can solve the class imbalance problem and specialized models for trash detection are needed.

## Acknowledgement

This research was results of a study on the “HPC Support” Project, supported by the ‘Ministry of Science and ICT’ and NIPA.

This work was supported by Artificial intelligence industrial convergence cluster development project funded by the Ministry of Science and ICT (MSIT, Korea) and Gwangju Metropolitan City.

## References

[1] G. E. Sakr, M. Mokbel, A. Darwich, M. N. Khneisser, and A. Hadi, “Comparing deep learning and support vector machines for autonomous waste sorting,” in *Proceedings of 2016 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)*, Beirut, Lebanon, 2016, pp. 207-212.

[2] Y. Chu, C. Huang, X. Xie, B. Tan, S. Kamal, and X. Xiong, “Multilayer hybrid deep-learning method for waste classification and recycling,” *Computational Intelligence and Neuroscience*, vol. 2018, article no. 5060857, 2018. https://doi.org/10.1155/2018/5060857

[3] Y. Liao, “A web-based dataset for garbage classification based on Shanghai’s rule,” *International Journal of Machine Learning and Computing*, vol. 10, no. 4, pp. 599-604, 2020.

[4] P. Zhang, Q. Zhao, J. Gao, W. Li, and J. Lu, “Urban street cleanliness assessment using mobile edge computing and deep learning,” *IEEE Access*, vol. 7, pp. 63550-63563, 2019.

[5] G. Mittal, K. B. Yagnik, M. Garg, and N. C. Krishnan, “Spotgarbage: smartphone app to detect garbage using deep learning,” in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, Heidelberg, Germany, 2016, pp. 940-945.

[6] M. S. Rad, A. V. Kaenel, A. Droux, F. Tieche, N. Ouerhani, H. K. Ekenel, and J. P. Thiran, “A computer vision system to localize and classify wastes on the streets,” in *Computer Vision Systems*. Cham, Switzerland: Springer, 2017, pp. 195-204.

[7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, “SSD: single shot multibox detector,” in *Computer Vision – ECCV 2016.* Cham, Switzerland: Springer, 2016, pp. 21-37.

[8] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014 [Online]. Available: https://arxiv.org/abs/1409.1556.

[9] J. Redmon and A. Farhadi, "YOLOv3: an incremental improvement," 2018 [Online]. Available: https://arxiv.org/abs/1804.02767.

[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, 2016, pp. 779-788.

[11] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, 2017, pp. 6517-6525.

[12] Z. Cai and N. Vasconcelos, "Cascade R-CNN: delving into high quality object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, 2018, pp. 6154-6162.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 28, pp. 91-99, 2015.

[14] K. Kim, H. I. Choi, and K. Oh, "Object detection using ensemble of linear classifiers with fuzzy adaptive boosting," *EURASIP Journal on Image and Video Processing*, vol. 2017, article no. 40, 2017. https://doi.org/10.1186/s13640-017-0189-y

[15] J. Zhu, F. Yu, G. Liu, M. Sun, D. Zhao, Q. Geng, and J. Su, "Classroom roll-call system based on ResNet networks," *Journal of Information Processing Systems*, vol. 16, no. 5, pp. 1145-1157, 2020.

[16] D. Yadav, S. Sanchez-Cuadrado, and J. Morato, "Optical character recognition for Hindi language using a neural-network approach," *Journal of Information Processing Systems*, vol. 9, no. 1, pp. 117-140, 2013.

[17] J. Wang and Y. Yagi, "Shadow extraction and application in pedestrian detection," *EURASIP Journal on Image and Video Processing*, vol. 2014, article no. 12, 2014. https://doi.org/10.1186/1687-5281-2014-12

[18] W. M. D. B. Wan Zaki, A. Hussain, and M. Hedayati, "Moving object detection using keypoints reference model," *EURASIP Journal on Image and Video Processing*, vol. 2011, article no. 13, 2011. https://doi.org/10.1186/1687-5281-2011-13

[19] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. Lawrence Zitnick, "Microsoft coco: common objects in context," in *computer vision – ECCV 2014*. Cham, Switzerland: Springer, 2014, pp. 740-755.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, 2016, pp. 770-778.

**Min-Seok Jo** **https://orcid.org/0000-0002-7483-4142**

He is currently pursuing a M.S. degree in the Department of Electrical Engineering from Korea University. His research interests include deep learning and distributed parallel processing.

**Seong-Soo Han** **https://orcid.org/0000-0002-4915-6247**

He is a professor in the Division of Liberal Studies at Kangwon National University. Before joining Kangwon National University in 2019, he was a professor at Soonchunhyang University during 2018–2019. He received the B.S. degree in Information and Communication Engineering from Gyeongsang National University, He received the M.S. degree in Information and Communication Engineering from Soonchunhyang University, Korea in 2005, the Ph.D. degree in Visual Information Processing from Korea University in 2019. He was a Director of the Orion Technology in 2015–2016. His research interests include computer education, AI, blockchain, deep learning and distributed parallel processing.

**Chang-Sung Jeong** **https://orcid.org/0000-0001-9654-8406**

He is a professor at the Department of Electrical Engineering at Korea University. Before joining Korea University in 1992, he was a professor at POSTECH during 1982–1992. He was on editorial board for Journal of Parallel Algorithms and Application in 1992–2002. Also, he was a chair of IEEE Seoul Section and has been working as a chairman of Computer Chapter at Seoul Section of IEEE region 10. He was a chairman of EE department in Korea University and a leader of BK21 project. His research interests include distributed parallel computing, cloud computing, networked virtual environment, and distributed parallel deep learning for real-time image processing and visualization.