ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

RSL Robotic Systems Lab
Prof. Dr. Marco Hutter

Semester Thesis

Refined detection and classification for trash sorting

Autumn Term 2021

Supervised by:
Kolvenbach Hendrik
Krämer Koen
Esquivel Estay Fidel

Author:
Liudi Yang

Declaration of Originality

I hereby declare that the written work I have submitted entitled

Refined detection and classification for trash sorting

is original work which I alone have authored and which is written in my own words.[1]

Author(s)

Liudi	Yang

Student supervisor(s)

Kolvenbach	Hendrik
Krämer	Koen
Esquivel Estay	Fidel

Committee members(s)

First name	Last name

Supervising lecturer

Marco	Hutter

With the signature I declare that I have been informed regarding normal academic citation rules and that I have read and understood the information on 'Citation etiquette' (`https://www.ethz.ch/content/dam/ethz/main/education/rechtliches-abschluesse/leistungskontrollen/plagiarism-citationetiquette.pdf`). The citation conventions usual to the discipline in question here have been respected.

The above written work may be tested electronically for plagiarism.

Zurich 18.02.2022
Place and date

Liudi Yang
Signature

[1] Co-authored work: The signatures of all authors are required. Each signature attests to the originality of the entire piece of written work in its final form.

Intellectual Property Agreement

The student acted under the supervision of Prof. Hutter and contributed to research of his group. Research results of students outside the scope of an employment contract with ETH Zurich belong to the students themselves. The results of the student within the present thesis shall be exploited by ETH Zurich, possibly together with results of other contributors in the same field. To facilitate and to enable a common exploitation of all combined research results, the student hereby assigns his rights to the research results to ETH Zurich. In exchange, the student shall be treated like an employee of ETH Zurich with respect to any income generated due to the research results.
This agreement regulates the rights to the created research results.

1. Intellectual Property Rights

1. The student assigns his/her rights to the research results, including inventions and works protected by copyright, but not including his moral rights ("Urheberpersönlichkeitsrechte"), to ETH Zurich. Herewith, he cedes, in particular, all rights for commercial exploitations of research results to ETH Zurich. He is doing this voluntarily and with full awareness, in order to facilitate the commercial exploitation of the created Research Results. The student's moral rights ("Urheberpersönlichkeitsrechte") shall not be affected by this assignment.
2. In exchange, the student will be compensated by ETH Zurich in the case of income through the commercial exploitation of research results. Compensation will be made as if the student was an employee of ETH Zurich and according to the guidelines "Richtlinien für die wirtschaftliche Verwertung von Forschungsergebnissen der ETH Zürich".
3. The student agrees to keep all research results confidential. This obligation to confidentiality shall persist until he or she is informed by ETH Zurich that the intellectual property rights to the research results have been protected through patent applications or other adequate measures or that no protection is sought, but not longer than 12 months after the collaborator has signed this agreement.
4. If a patent application is filed for an invention based on the research results, the student will duly provide all necessary signatures. He/she also agrees to be available whenever his aid is necessary in the course of the patent application process, e.g. to respond to questions of patent examiners or the like.

2. Settlement of Disagreements

Should disagreements arise out between the parties, the parties will make an effort to settle them between them in good faith. In case of failure of these agreements, Swiss Law shall be applied and the Courts of Zurich shall have exclusive jurisdiction.

Zurich 18.02.2022	Liudi Yang
Place and date	Signature

Contents

Abstract

The riverine trash crisis has been growing increasingly severe over years. The statistic exhibits 0.8 to 2.7 million metric tons waste has been emitted into the rivers. Our planet is suffering from the harm brought by all kinds of riverine waste. To develop an efficient method to deal with the trash crisis,in 2021, the Autonomous River Cleanup (ARC) project has designed the first prototype to collect and classify trash from river and tested it in Limmat River in Zurich. From real experiments, we have found some improvement that can potentially increase the recycling efficiency. This project proposes a method to generalize the classifier to more detailed categories for trash detection and classification. By introducing state-of-the-art backbone Swin Transformer for the Mask RCNN, the detecting accuracy can be improved to reach mAP(bbox) 0.569 and mAP(segmentation) 0.599. Furthermore, the detected trash can be tracked by SORT algorithm based on the detection result in order to obtain corresponding information. We have built ROS infrastructure to extract trash information for further comprehensive analysis and data reuse by other neural networks.

Chapter 1

Introduction

The issue of riverine trash has been impairing the ecosystem continuously and broadly. On a global scale, macroplastic emissions into the Ocean reach 0.8–2.7 million metric tons per year (mt/yr) according to the most recent model of riverine plastic export[1]. It has been witnessed the amount of plastic produced annually surge rapidly over the last few decades to an estimated 288 million tonnes in 2012 (Figure 1.1), and this total continues to grow at about 4% per year[2]. However, the properties why plastics are that useful also make inappropriately handled plastics waste a significant environmental threat. Their durability causes that they will exist in the environment for millions of years, and their low density means that they are simply dispersed by water and wind, sometimes traveling thousands of kilometers from source areas. Worse still, the plastic will break down into small particles. Mean values recorded in sediment from sea and beaches ranged between 1.5–671.0 items/kg. It is estimated the mean values ranging between 4.4 and 166.7 items/kg. Thompson et al. recorded mean values of 31.0 items/kg in estuarine sediments, while, Vianello et al. recorded mean values of microplastic in sediments of 1445.2 items/kg in the Venice lagoon[3].

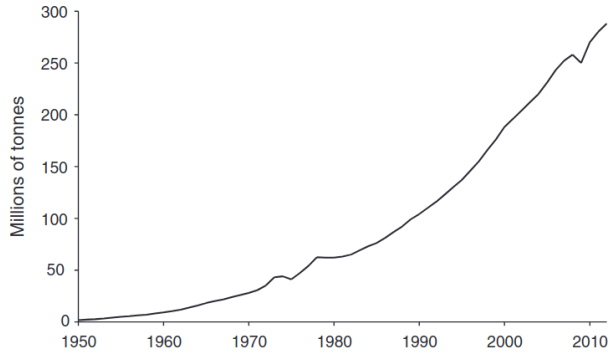

Figure 1.1: Growth in global plastic production from 1950 to 2012

To mitigate the worsening trash problem, the Autonomous River Cleanup (ARC) project has designed the platform deployed near the bank to collect, detect and classify trash automatically. The core of the detection system is composed of Deep CNN model and RGB-D camera. The trash in the dataset has been labeled to be distinguished from non-litter items and further been classified into plastic lit-

ter and other litter. The trained models are effective at detecting trash and can achieve a segmentation mask AP@0.5 of 74.84(1 class) and 49.74(2 classes). It is huge achievement to conduct real experiments on this first prototype vessel(Figure 1.2,1.3) in Limmat River in Zurich in the summer of 2021. Nevertheless, the trash recycling efficiency can be increased by generalizing the classifier into more categories so that all the collected trash will not be mixed up in the same container restricted by the binary classifier. Additionally,not only can the detected trash be classified into detailed classes, but also we hope the detection system to keep track of detected trash and extract corresponding information which can be utilized for further comprehensive data analysis and reuse.

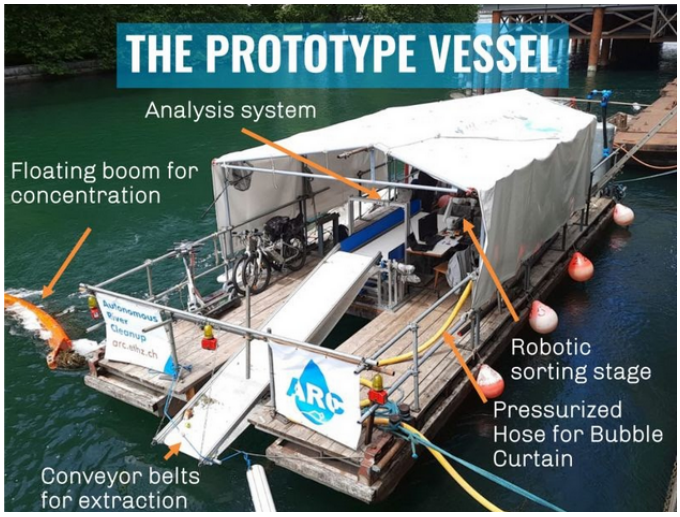

Figure 1.2: ARC first prototype vessel

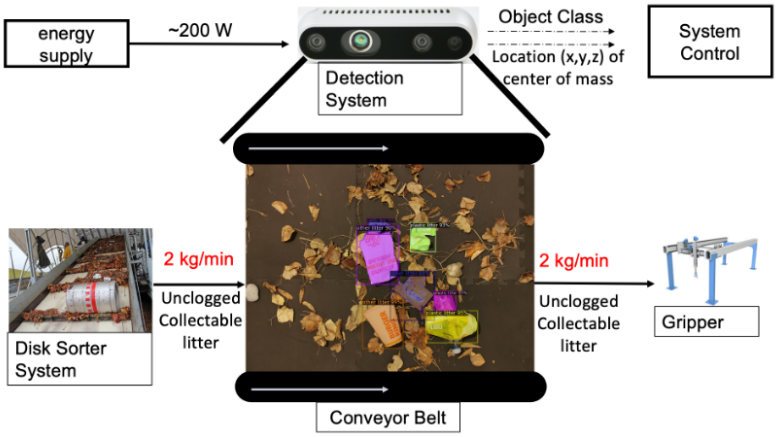

Figure 1.3: Overview of ARC detection system 1.0

This work has implemented the whole system for trash detection and data extraction. The overview is as follows(Figure 1.4). In order to improve the multi-category detector accuracy and robustness, the novel neural network architecture Swin-Transformer[4] has been introduced as the backbone for the Mask RCNN[5]. The attention mechanism[6] plays the key role during the feature extraction procedure. Rest of the architecture has inherited the original Mask RCNN. In the instance

segmentation area, the Mask RCNN brings breakthrough insight and inspiration to attain excellent accuracy. The tracker realized according to the SORT[7] algorithm can exploit the detection result transmitted from detector to keep track of detected trash and assign a unique ID for every object as part of the object information. All of the object information including object ID, category and cropped image will be saved for database for trash composition and quantity statistics. Meanwhile, the annotation generated from neural network prediction will be utilized to label images automatically that can save plenty of boring labor from human beings and possibly used by other neural networks.

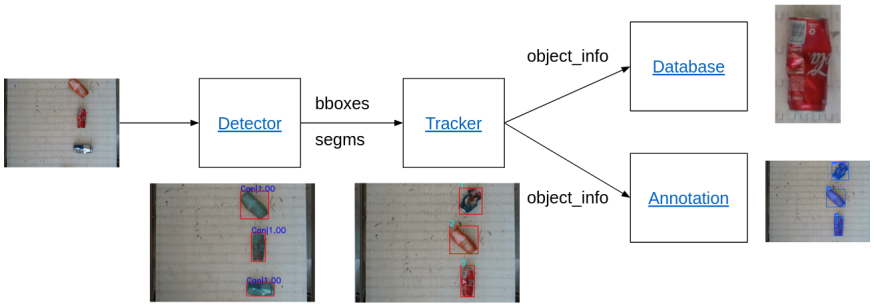

Figure 1.4: Refined system overview

Chapter 2

Related Work

The primary goal of this project is to apply deep neural networks into the trash detection system. This section introduces some related work and technology of instance segmentation and some datasets used for training and test of trash detection.

2.1 Instance Segmentation

Instance segmentation is a computer vision task for detecting and localizing an object in an image. The model assigns a label to each pixel of the image. It mainly consist of two parts. First work is to classify individual objects and localize each object instance using a bounding box. The second part is to classify pixels into categories based on the instances rather than classes and output individual instance mask.Integrated model of instance segmentation, the detection system can output the class, bounding box and segmentation mask of the trash(Figure 2.1). Object information can be subscribed by sequential components of the trash recycling system. For instance, the robotic arm can calculate the grasping parameters and trajectories with the centroid and shape of the trash.

Figure 2.1: Example of instance segmentation output

Inspired by the significant success of deep convolutional nerual network in computer vision area, a great quantity of frameworks with foundation of this has de-

veloped swiftly. In 2014, Hariharan et al. proposed the earliest instance segmentation algorithm SDS that realized the combination of object detection and semantic segmentation[8]. Girshick et al. first explored the influence of CNNs and region suggestion block on instance segmentation in the same year.[9]. In 2015, Dai et al. presented the CFM (convolutional feature masking) algorithm, which put forward the pixedl mask into instance segmentation for the first time[10].In 2015, Girshick proposed the object detection framework Fast R-CNN, which regards image and a set of object proposals as input[11]. Faster R-CNN is a target detection algorithm proposed by He et al. in 2015, which won many first prizes in COCO contests in 2015[12]. Based on Fast R-CNN,RPN proposal box generation algorithm is put forward in this algorithm, which greatly speeds up the target detection. In 2017, He et al. proposed the Mask R-CNN[5] detection algorithm. Mask R-CNN is an instance segmentation algorithm most widely used with the highest efficiency in the current stage.This method will be discussed later in the next section. In 2019, Chen et al. proposed the hybrid task cascade (HTC) algorithm. The key for this algorithm to perform instance segmentation is to make the best use of the inverse relationship between object detection and object instance segmentation. [13] A multi-stage object detection architecture, the Cascade R-CNN, composed of a sequence of detectors trained with increasing IoU thresholds, is proposed to address overfitting.[14]

The evolution of the instance segmentation algorithm based on Deep CNN has stimulated the progress of detection accuracy. The state-of-the-art model Mask RCNN presents high-quality segmentation mask for every instance. Compared to more complicated models, it can reach exceptional prediction accuracy with light weight framework. It is promising to leverage this as the basic architecture to better the classifier performance with some adjustment.

2.2 Mask RCNN

Our detector model inherits most architecture from Mask RCNN for the object detection task. Mask RCNN is a two-stage detection system built on the Faster RCNN neural network. It generates region proposals where there may be an object in the input image as the first step. After that,it predicts the class of the object, refines the bounding box and generates a mask in pixel level of the object based on the first stage proposal. Both stages are linked to the backbone section.

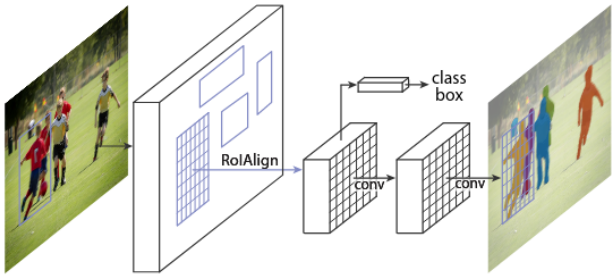

Figure 2.2: Architecture of Mask RCNN[5]

The input image is fed into a CNN, functioning as the backbone to extract features from raw images, which is usually a pretrained network such as ResNet101. Normally, to obtain higher accuracy and scale invariance, the ResNet is refined by Feature Pyramid Network(FPN). The Region Proposal Network (RPN) uses

a shifted window method to attain relevant anchor boxes from the feature maps. Those anchor boxes are the areas object might exist. The second stage is essentially Fast R-CNN, but compared to previous version it replaces RoI pooling layer with RoI Align layer to improve the quantization precision. The scheme of Mask RCNN is shown in Figure 2.2. The prediction consists of class probability and bounding box in addition to the binary segmentation mask provided by the mask branch. The loss function is defined as :

$$L = L_{cls} + L_{bbox} + L_{mask}$$

The Mask RCNN decouples the loss function into 3 tasks. The class loss (L_{cls}) and bounding box loss (L_{bbox}) are similar to Faster RCNN. The segmentation mask loss is defined as the average cross entropy for the mask branch.

2.3 Vision Transformer

Ever since the attention mechanism was put forward, it has brought revolutionary progress not only in Natural Language Processing (NLP) but also CV area.Self-attention, sometimes called intra-attention is an attention mechanism relating different positions of a single sequence in order to compute a representation of the sequence[6].In practice, the attention function is calculated on a set of queries simultaneously, packed together into a matrix Q. The keys and values are also packed together into matrices K and V . Q,K and V are obtained from the transformation of original sequence. We compute the matrix of outputs as:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

The Vision transformer (ViT) is a transformer used in the field of computer vision to regard image as input sequence following the properties of Transformer. The pioneering work of ViT directly applies a Transformer architecture on non-overlapping medium-sized image patches for image classification. It partitions the input image as sequence and encode the patches as the extracted features.[15] It definitely enlightens the progress to apply Transformer into vision area. The overview of ViT is as Figure 2.3.

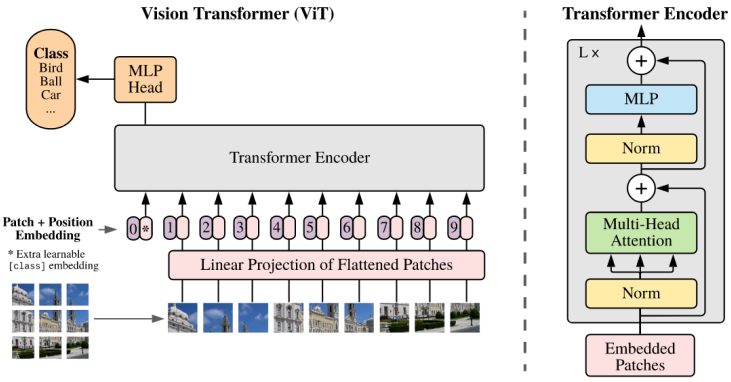

Figure 2.3: ViT overview[15]

2.4 Dataset

One crucial part for deep-learning based algorithm is the supervised dataset. In order to obtain trained neural network with acceptable performance, it requires a great quantity of labeled images. Some open source datasets are available for training but not all of them satisfy our requirement completely. The selection and supplement work are still needed.Some commonly used datasets are listed in Table 2.1.

TACO It is an open image dataset of waste in the wild. It contains photos of trash taken with diverse background, from tropical beaches to city streets. These images are manually labeled and segmented according to a hierarchical taxonomy to train and evaluate object detection algorithms[16]. The official remapping sheet has been provided to decrease the detector classes.

TrashNet The dataset includes six classes: glass, paper, cardboard, plastic, metal, and other trash. Currently, the dataset consists of 2527 images captured by mobile devices. The pictures were taken by placing the object on a white posterboard and using sunlight and/or room lighting[17]. It is excellent resource due to the plain background.

TrashCan The TrashCan dataset is comprised of annotated images (7,212 images currently) which contain observations of trash and a wide variety of undersea flora and fauna. The categories span 34 classes. The annotations in this dataset take the format of instance segmentation annotations: bitmaps containing a mask marking which pixels in the image contain each object[18].

UAVVaste Using an unmanned aerial vehicle (UAV) equipped with a GPS locator and performing a flight along a given route (automatically or manually controlled), locate abandoned waste in a given area.The UAVVaste dataset consists of 772 images and 3716 annotations of trash[19].

MJU-Waste The current version of dataset, MJU-Waste V1, contains 2475 coregistered RGB and depth image pairs. Specifically,it is randomly split the images into a training set, a validation set and a test set of 1485, 248 and 742 images, respectively[20].

Dataset	No.classes	No.images	Annotation	Comment
TACO	60	1500	segmentation	widely used
Trashnet	6	2527	classification	clear background
TrashCan	34	7212	segmentation	underwater
UAVVaste	1	772	segmentation	drone view
MJU-Waste	1	2475	segmentation	indoor

Table 2.1: Overview of commonly used trash datasets

Chapter 3

Method

This chapter introduces the method to build the whole system. The system is mainly comprised of the detector and tracker.

3.1 Detection

3.1.1 Model Selection

The objective of the detector is to obtain the bounding box and segmentation mask of the detected trash. Therefore when selecting the model, the mask AP is the key metric to determine the appropriate model. Considering the generalization performance on COCO instance segmentation test(Table 3.1). The model Swin-L utilizing Swin-Transformer as the backbone beats other rivals on COCO dataset. Compared to the previous model ARC detection system has utilized MaskRCNN with ResNet as backbone, it can bring about 14% inprovement regarding mask AP. The newly designed backbone based on Transformer block is promising to improve our detector performance. Considering the GPU capability and real-time calculation requirement, the Mask RCNN framework is still inherited for prediction although other more complicated models such as HTC and Cascade MRCNN may bring higher accuracy. More advanced models require superior training and deployment device, leading to difficulty of application. There must be a trade-off between the performance and application difficulty. The total architecture of our neural network is shown in Figure 3.1. The original ResNet backbone is replaced by Swin-Transformer in Mask RCNN.

Model	Backbone	Mask AP	Year
Swin-L	Swin-Transformer	51.1%	2021
DetectoRS	ResNeXt101	47.1%	2020
Cascade MRCNN	ResNeXt152	43.3%	2019
SOLOv2	Res-DCN	41.7%	2020
Mask-RCNN	ResNeXt101	37.1%	2017

Table 3.1: Instance Segmentation on COCO test-dev

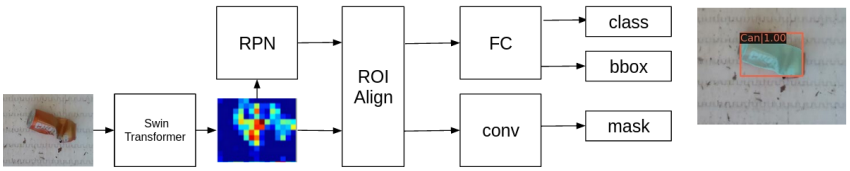

Figure 3.1: Model architecture

3.1.2 Swin Transformer

Swin Transformer represents hierarchical vision transformer using sliding windows. The core function is to extract features from raw images to feed into rest of the neural network. In comparison with the preceding vision transformers, it incorporates hierarchical downsample and shifted window to address the global information. The overall overview of the architecture is presented in Figure 3.2. Several Transformer blocks with refined self-attention computation are applied on these patch tokens. Swin Transformer is established by replacing the standard multi-head self attention (MSA)[6] module in a Transformer block with a module based on regular windows(W-MSA). Other layers remains the same[4]. Another MSA is modified to a shifted window based MSA(SW-MSA) module, followed by a 2-layer Multiple Layer Perceptron (MLP) with GELU non-linearity in between. A LayerNorm (LN) layer is applied before each MSA module and each MLP, and a residual connection is applied after each module. The window-based self-attention module omits connections across windows, which limits its observation. To reduce computation overhead by remaining non-overlapping windows and combine features with other windows, a shifted window partitioning approach is proposed in Figure 3.3 which alternates between two partitioning configurations in consecutive Swin Transformer blocks. This can bridge adjacent window information to current window to avoid local focus.

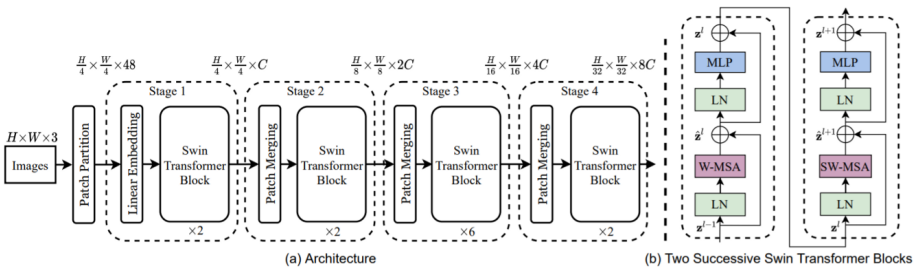

Figure 3.2: Swin Transformer architecture[4]

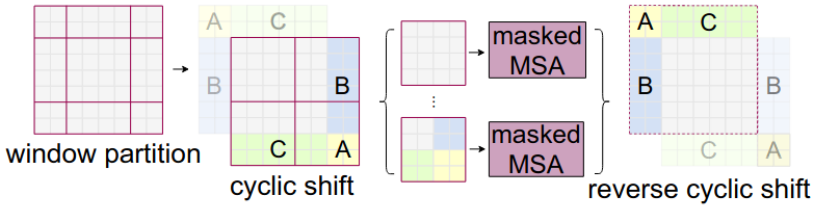

Figure 3.3: Illustration of shifted window partitioning[4]

3.1.3 Synthetic Dataset

TACO dataset satisfies our need because it contains enough trash categories and images meanwhile it can provide segmentation information in the COCO format which can facilitate the training. Dataset has played an important role to guarantee the model accuracy during training. When the dataset annotation is imbalanced, it will influence the generality and correction rate of the neural network. However the categories of annotations in TACO that are worthwhile and efficient to recycle, for instance,plastic bottles and can are not the majority of the annotation according to Figure 3.4. Other uncorrelated objects will definitely disturb the prediction.

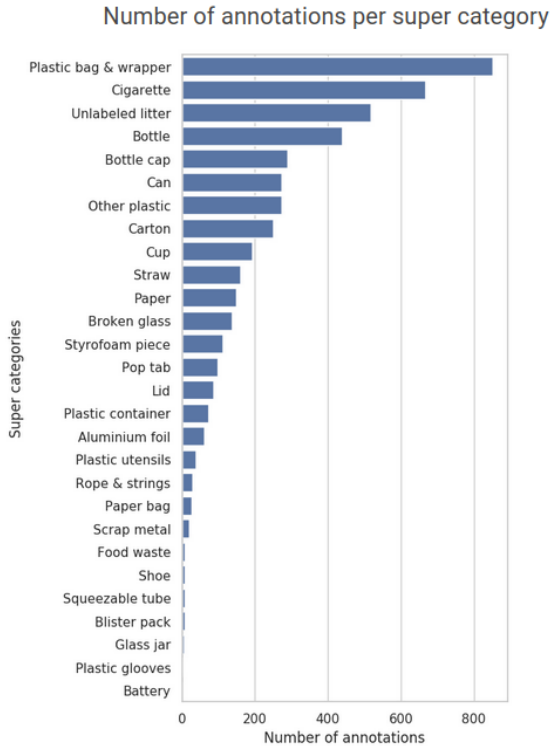

Figure 3.4: Number of annotations per super category in TACO dataset

It is necessary to supplement TACO with the annotations the detector should target. TrashNet is a commonly used trash dataset with clear background. And with some extra images from ARC platform, the detector can focus more on the images with conveyor belt as the background. The final version of the train dataset is

synthetic, comprised of TACO,TrashNet and ARC experiment images. The supplement dataset is self-labeled as multiple categories. The volume of the dataset is 1734 (1500+200+34).

Figure 3.5: From left to Right: TACO,TrashNet and ARC experiment

3.1.4 Train Procedure

The detector model has been trained on the synthetic dataset described before.The initialized weights has been pretrained on COCO dataset including 80 classes. By using the transfer learning technique, the converging procedure can be sped up and generality of the model can be optimized by larger volume of datasets. Some tricks has also been exploited to counteract overfitting and local optimum problem.

5-fold cross validation Modern advance neural network contains millions of parameters which requires loads of labeled data. The advantage of this approach is a lower-variance estimate of the model performance than the traditional method. This technique is used because it helps to avoid overfitting, which can occur when a model is trained using all of the data. For every run,20% of the dataset will be regarded as the validation set to help determine whether or not the model tends to be overfitted.

Data augmentation In order to make full use of the supervised data, we incorporate data augmentation into the training pipeline, including random flip,random crop,resize and normalization. It is a fact that most models perform well with more data. The transformation of the image dataset can make it more robust to deal with more complicated environment.

Drop-out Dropout is a regularization method that approximates training a large number of neural networks with different architectures in parallel.During training, some number of layer outputs are randomly ignored or forced to 0. This has the effect of making the layer as a new layer with a different number of nodes and connectivity to the prior layer. In effect, every update to a layer during training is performed with a different view of the configured layer[21].

Momentum Stochastic gradient descent is an optimization algorithm which works by finding the direction of steepest slope iteratively. One drawback of this method is the tendency of falling into the local optimum. Therefore it's preferable for us to make the algorithm keep the direction it has already been going along for sometime

before it changes its direction[22]. We define the velocity at the step t as V_t. The descent update is:

$$V_t = \beta V_{t-1} + \alpha \nabla_w L(W, X, y)$$
$$W = W - V_t$$

3.2 Tracking

The tracker is realized according to the SORT algorithm. First based on Kalman filter,the prior state estimation can be propagated to get next frame state. The inter-frame displacements for every object are approximated with a linear constant velocity model which is independent of other objects and camera motion. The state of each target is modelled as:

$$x = [u, v, s, r, \dot{u}, \dot{v}, \dot{s}]$$

where u and v represent the horizontal and vertical pixel location of the centre of the target, while the scale s and r represent the area and the aspect ratio of the target's bound- ing box respectively[7]. Second when detection result is associated to a target, the detected bounding box is used as the measurement to update the posterior target state where the velocity components are solved optimally via a Kalman filter measure update model. If no detection is associated to the tar- get, its state is simply predicted without correction using the linear velocity model.The state equations are:

$$x_k = Ax_{k-1} + w_{k-1}$$

where

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & dt & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & dt & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & dt \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$w_{k-1} \sim \mathcal{N}(0, R)$$

w_{k-1} represents the system noise. Its covariance matrix is R.The measurement model is described as

$$z_k = Hx_{k-1} + v_k$$

where

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$v_k \sim \mathcal{N}(0, Q)$$

v_k is the measurement noise and its covariance matrix is Q.

When assigning detected bounding boxes to stored targets, each target's bounding box is estimated by predicting its new location in the current frame. The assignment cost matrix is then calculated as the intersection-over-union (IoU) distance between each detection and all predicted bounding boxes from the existing targets. The assignment is solved optimally using the Hungarian algorithm (Table 3.2). Additionally, a minimum IoU is imposed to reject assignments where the detection to target overlap is less than IoUmin. An example is given in Figure 3.6

Detection/Tracking	Tracking 1	Tracking 2	Tracking 3
Detection A	IoU=0	IoU=0	IoU=0
Detection B	**IoU=0.56**	IoU=0	IoU=0
Detection C	IoU=0	**IoU=0.77**	IoU=0

Table 3.2: Example of Hungarian algorithm[23]

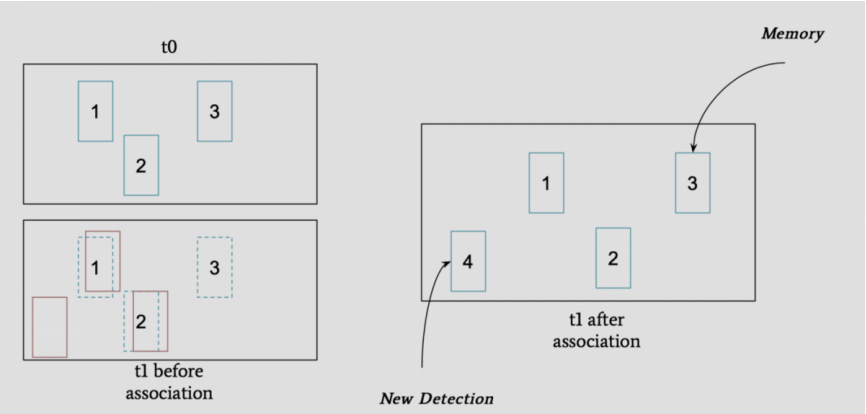

Figure 3.6: Association of detection and existing bbox example[23]

Chapter 4

Result and Analysis

4.1 Evaluationi Metric

AP (Average precision) is a widely used metric in measuring the accuracy in instance segmentation area. Avearge precision computes the average precision value for recall value over 0 to 1 for all detection results[24].

Precision & recall Presion represents the accuracy of the accuracy. Recall measures the probability of finding all the positives.They are defined as follows.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

TP = True positive, TN=True negative , FP=False positive , FN=False negative

Intersection over Union IoU measures the overlap part between two bounding boxes. It is used to rate the accuracy of that predicted result overlaps with the ground truth result. can be set the threshold to trigger positive or negative prediction of the detector.

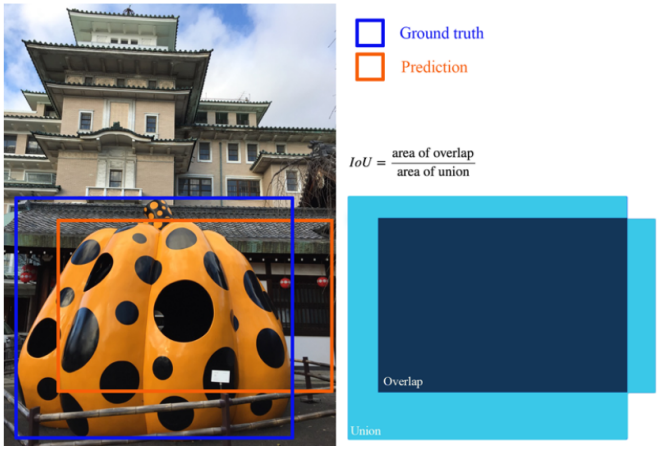

Figure 4.1: Illustration of IoU[24]

AP From all the prediction results, they can be ranked in descending order according to the predicted confidence level and be visualized as the precision-recall curve.The general definition of average precision is the area under the curve.

$$AP = \int_0^1 p(r)dr$$

mAP(mean average precision) is the average of AP for each class of the detector.mAP@[.5:.95] corresponds to the average AP for IoU from 0.5 to 0.95. Researchers usually compare different models under this metric.

4.2 Train

The Mask RCNN with Swin Transformer is implemented on MMDetection platform[25]. It can provide stable training environment and optimize the learning rate. The device for training is RTX2080 with 8G memory. Training takes about 5 hours totally. And the GPU memory condition during training is shown in Figure 4.2. The training loss curve and validation result curve are ploted in Figure 4.3, Figure 4.3 respectively. During every run, the loss function has been decreased and gone to converge.From the validation result, the evaluation metric mAP has been increased over iterations. In conclusion, there is no obvious overfitting during procedure due to the tricks introduced above.

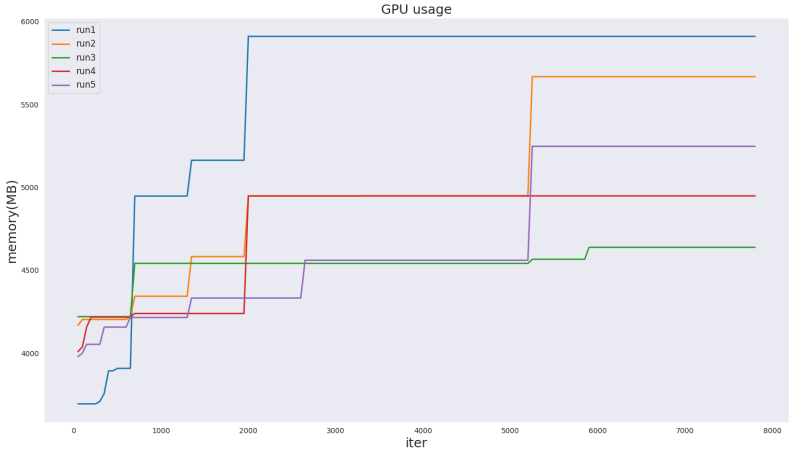

Figure 4.2: GPU usage during training

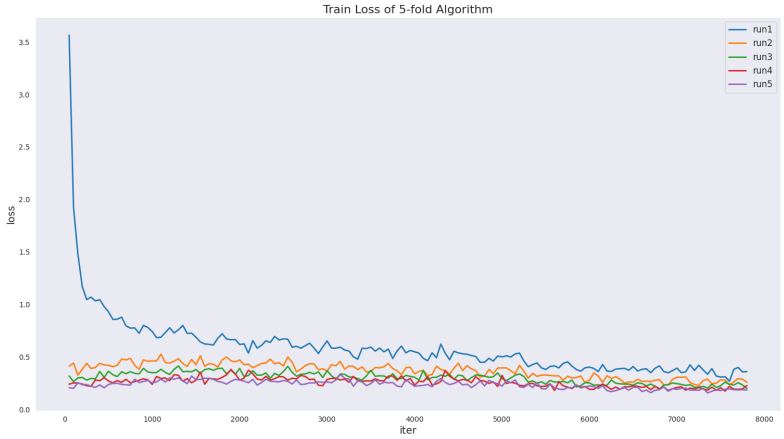

Figure 4.3: Train loss of 5-fold algorithm

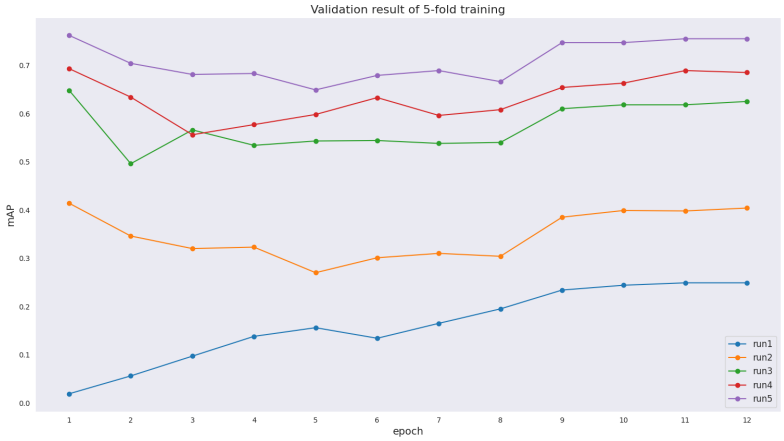

Figure 4.4: Validation result of 5-fold algorithm

4.3 Test

The test dataset is comprised of self-labeled images from ARC experiment and part of TrashNet dataset.The volume is 105 images and 156 annotations. ARC experiment dataset contains the trash images collected from Limmat River in 2021.We have conducted two criterion to label things: 7 classes and 9 classes. With more categories to differentiate, the model tends to have more difficulty to raise the probabilities of prediction. Remapping 9 classes into 7 is reasonable to decrease the ambiguity between interclasses according to the confusion matrix in Figure 4.5 and Figure 4.6.

The final categories that the detector can recognize are Can, Carton/Paper, GlassBottle, Other, PlasticBottle, PlasticOther and Wrapper. The mAP of the model can reach $56.9\%, 59.9\%$ relatively for bounding box and segmentation mask

(Table 4.1). When the background is clear enough and the object is not deformed a lot, the detector can output relatively accurate results. Some test images are shown in Figure 4.7. But there are still some factors that will lead to the detection failure(Figure 4.8). First, owing to the shadow cast on the surface of the object, the extracted features is quite vague and that will impair the output accuracy.The second factor is the similarity between some categories.For instance, at the current stage it is even difficult for humans to recognize between glass bottles and plastic bottles from the first glance. Third the occlusion of other objects will disturb the prediction of the detector. The information of shape and appearance is missing to give the correct classification.

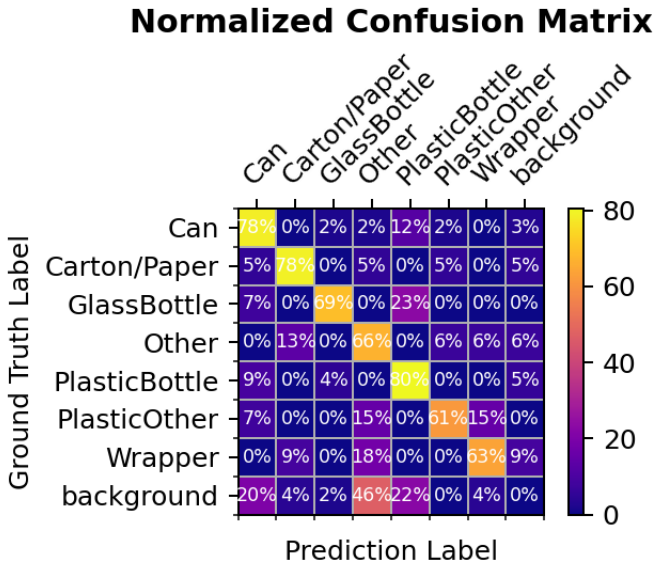

Figure 4.5: Confusion matrix of 7 classes

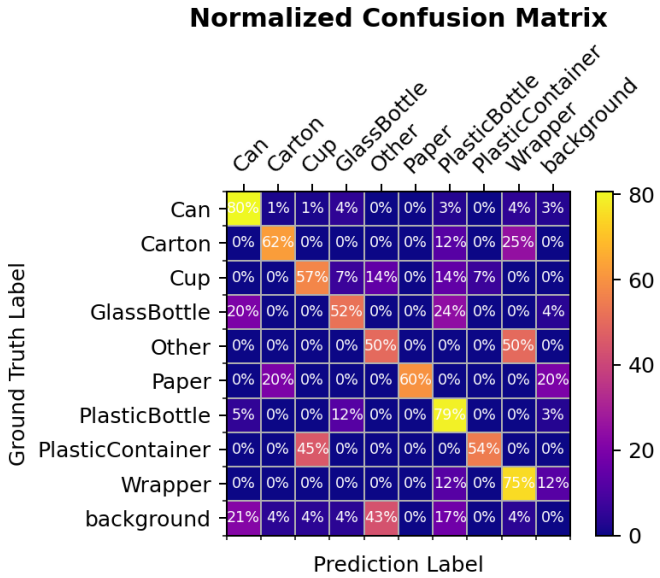

Figure 4.6: Confusion matrix of 9 classes

Figure 4.7: Test image examples

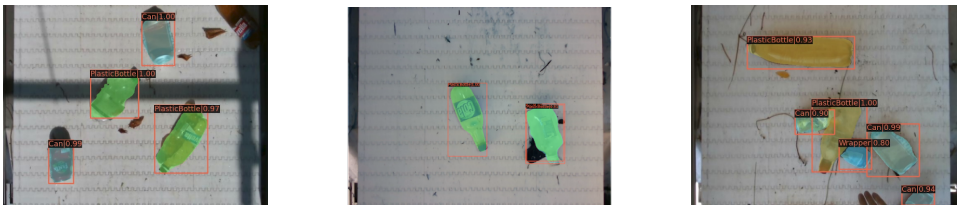

Figure 4.8: Failure cases

No.classes	bbox_mAP	bbox_mAP@.5	bbox_mAP@.75	segm_mAP	segm_mAP@0.5	segm_mAP@.75
9	0.554	0.718	0.663	**0.618**	0.710	0.681
7	**0.569**	**0.750**	**0.703**	0.599	**0.761**	**0.699**

Table 4.1: Test result

4.4 Contrast

In contrast to the ARC detection system 1.0, the current detector can not noly generalize to more detailed categories but also improve the robustness of detection and tracking. When the input object is deformed, the previous version can just identify part of the trash(Figure 4.9). The refined detector can output the correct segmentation mask(Figure 4.10). Plus, due to the composition of TACO dataset, the previous detector will be interfered with other kinds like the label and cap of the bottle(Figure 4.11).Now the detector can recognize whole body of the trash(Figure 4.12)

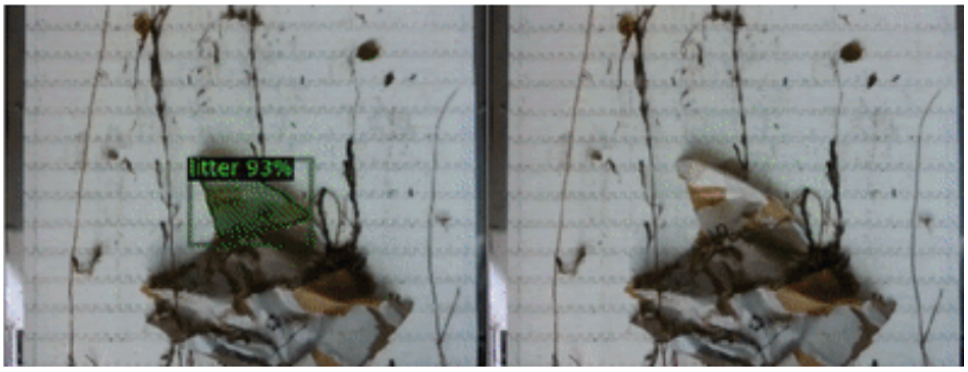

Figure 4.9: Previous detection result of deformed trash

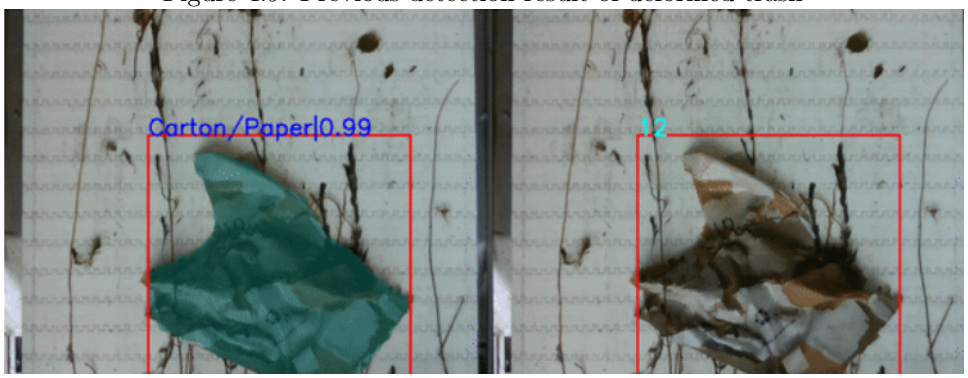

Figure 4.10: Current detection result of deformed trash

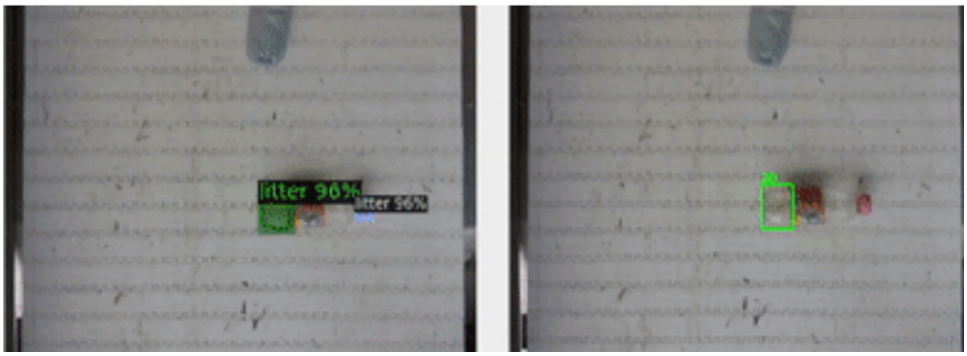

Figure 4.11: Previous detection result of plastic bottles

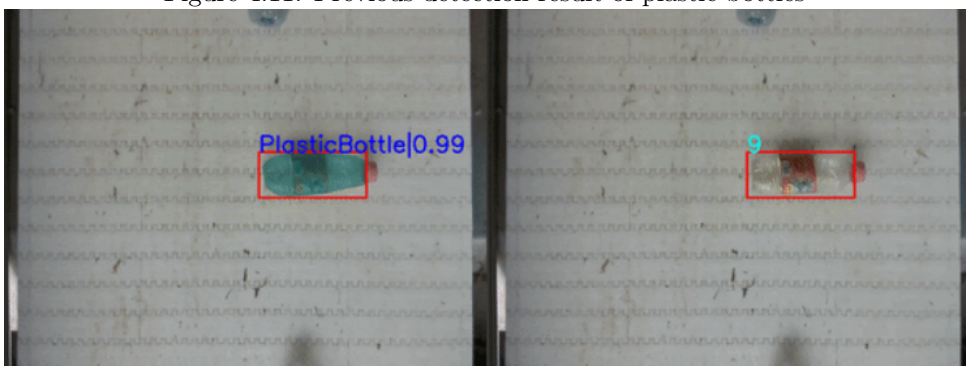

Figure 4.12: Current detection result of plastic bottles

4.5 ROS Implementation

To collect corresponding information from the detected object, we also design and build the ROS infrastructure to extract trash information and transmit it via self-defined message.

Database The detected trash information including object ID, categories, cropped image, prediction probability and size information will be stored into json file which can be uploaded to the database. The detected bounding box with highest prediction score will be regarded as the candidate. Some examples of cropped images are shown in Figure 4.13.

Autonomous annotation The labeling work to provide instance segmentation information is boring and exhausting. We hope to use the prediction result from the detector to annotate selected images automatically. When the repeated times of detected object information that has been received by the annotation node is greater than set threshold, via some format conversion, the annotation will be saved in COCO format. This kind of data can be fed into other neural networks for training. Some annotation demos are shown by COCO-viewer[26] in Figure 4.14.

Figure 4.13: Examples of cropped images

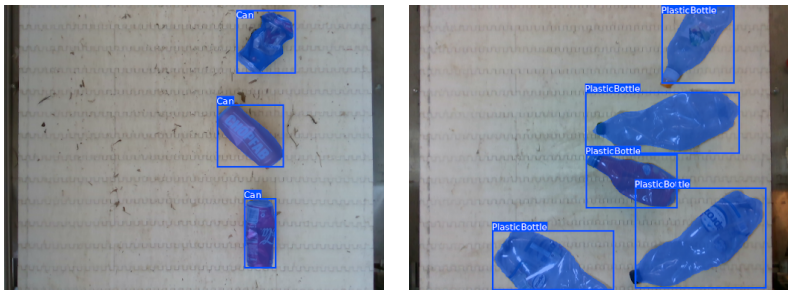

Figure 4.14: Examples of autonomous annotation

Chapter 5

Conclusion

The aim of this project is to refine the detection system on the basis of detector of ARC first prototype. To improve the classifier performance and robustness, the newly designed backbone Swin Transformer has been introduced to achieve great performance. Because TACO dataset is quite general and imbalanced, we self-labeled hundreds of images to enrich the synthetic dataset. The Mask RCNN model with Swin Transformer has been successfully trained for instance segmentation task. To keep track of detected trash for intention to extract information, the tracker based on SORT algorithm is incorporated to encapsulate the object information. The ROS implementation for storing data has also been realized. It can be used to collect plenty of data in the future. Compared to the first generation detection system, the classes has been generalized, the robustness has been increased, the data reuse has been facilitated.

Further work can explore more complicated models such as HTC[13] and Cascade Mask RCNN[14]. We can resort to powerful cluster for efficient training. Another enhancement measure can append more images of the trash collected directly from the river. This kind of dataset can notify the model with the real conditions it may encounter like extremely deformed and wet surface. It is also worthwhile and interesting to try some self-supervised frameworks[27] to fully use the images. This will definitely help to focus on the riverine trash detection. As for the tracker, now the mechanism is totally dependent on the motion estimation. It is difficult to tackle the motion blur and intense variation. Thus the keypoints extracted according to the vision algorithm can be supplemented to the distance calculated by Hungarian algorithm to boost the robustness.

Bibliography

[1] F.-C. Mihai, S. Gündoğdu, L. A. Markley, A. Olivelli, F. R. Khan, C. Gwinnett, J. Gutberlet, N. Reyna-Bensusan, P. Llanquileo-Melgarejo, C. Meidiana, S. Elagroudy, V. Ishchenko, S. Penney, Z. Lenkiewicz, and M. Molinos-Senante, "Plastic pollution, waste management issues, and circular economy opportunities in rural communities," *Sustainability*, vol. 14, no. 1, 2022.

[2] M. Bergmann, L. Gutow, and M. Klages, *Marine Anthropogenic Litter*, 06 2015.

[3] M. Renzi, V. H. Pauna, F. Provenza, C. Munari, and M. Mistri, "Marine litter in transitional water ecosystems: State of the art review based on a bibliometric analysis," *Water*, vol. 12, p. 612, 2020.

[4] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," 2021.

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," 2018.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[7] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3464–3468.

[8] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," 2014.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," 2014.

[10] J. Dai, K. He, and J. Sun, "Convolutional feature masking for joint object and stuff segmentation," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2015.

[11] R. Girshick, "Fast r-cnn," 2015.

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," 2016.

[13] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang, C. C. Loy, and D. Lin, "Hybrid task cascade for instance segmentation," 2019.

[14] Z. Cai and N. Vasconcelos, "Cascade r-cnn: High quality object detection and instance segmentation," 2019.

[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.

[16] P. F. Proença and P. Simões, "Taco: Trash annotations in context for litter detection," 2020.

[17] M. Yang, "trashnet," https://github.com/garythung/trashnet, 2017.

[18] J. Hong, M. Fulton, and J. Sattar, "Trashcan: A semantically-segmented dataset towards visual detection of marine debris," 2020.

[19] M. Kraft, M. Piechocki, B. Ptak, and K. Walas, "Autonomous, onboard vision-based trash and litter detection in low altitude aerial images collected by an unmanned aerial vehicle," *Remote Sensing*, vol. 13, no. 5, 2021.

[20] T. Wang, Y. Cai, L. Liang, and D. Ye, "A multi-level approach to waste object segmentation," *Sensors*, vol. 20, no. 14, 2020.

[21] J. Brownlee, "A gentle introduction to dropout for regularizing deep neural networks," https://machinelearningmastery.com/dropout-for-regularizing-deep-neural-networks/.

[22] V. Bushaev, "Stochastic gradient descent with momentum," https://towardsdatascience.com/stochastic-gradient-descent-with-momentum-a84097641a5d.

[23] "Computer vision for multi-object tracking — live example," https://www.thinkautonomous.ai/blog/?p=computer-vision-for-tracking.

[24] J. Hui, "map (mean average precision) for object detection," https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173.

[25] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, Z. Zhang, D. Cheng, C. Zhu, T. Cheng, Q. Zhao, B. Li, X. Lu, R. Zhu, Y. Wu, J. Dai, J. Wang, J. Shi, W. Ouyang, C. C. Loy, and D. Lin, "MMDetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.

[26] T. Savchyn, "coco-viewer," https://github.com/trsvchn/coco-viewer, 2019.

[27] Z. Weng, M. G. Ogut, S. Limonchik, and S. Yeung, "Unsupervised discovery of the long-tail in instance segmentation using hierarchical self-supervision," 2021.