Detecting floating litter in freshwater bodies with semi-supervised deep learning

Jia, Tianlong; de Vries, Rinze; Kapelan, Zoran; van Emmerik, Tim H.M.; Taormina, Riccardo

DOI
10.1016/j.watres.2024.122405

Publication date
2024

Document Version
Final published version

Published in
Water Research

Citation (APA)
Jia, T., de Vries, R., Kapelan, Z., van Emmerik, T. H. M., & Taormina, R. (2024). Detecting floating litter in freshwater bodies with semi-supervised deep learning. *Water Research*, *266*, Article 122405. https://doi.org/10.1016/j.watres.2024.122405

Contents lists available at ScienceDirect

Water Research

journal homepage: www.elsevier.com/locate/watres

Detecting floating litter in freshwater bodies with semi-supervised deep learning

Tianlong Jia [a,*], Rinze de Vries [b], Zoran Kapelan [a], Tim H.M. van Emmerik [c], Riccardo Taormina [a,*]

[a] *Delft University of Technology, Faculty of Civil Engineering and Geosciences, Department of Water Management, Stevinweg 1, 2628 CN Delft, The Netherlands*
[b] *Noria Sustainable Innovators, Schieweg 13, 2627 AN Delft, The Netherlands*
[c] *Wageningen University and Research, Hydrology and Environmental Hydraulics Group, Wageningen, The Netherlands*

ARTICLE INFO

Dataset link: https://github.com/TianlongJia/deep_plastic_SSL, https://doi.org/10.5281/zenodo.13730228, https://doi.org/10.5281/zenodo.13730298, https://doi.org/10.5281/zenodo.13730370, https://doi.org/10.5281/zenodo.13730384, https://doi.org/10.4121/78bb4822-7b70-4632-887a-7cacd344024e

Keywords:
Artificial intelligence
Object detection
Self-supervised learning
Environmental monitoring
Pollution
Plastics

ABSTRACT

Researchers and practitioners have extensively utilized supervised Deep Learning methods to quantify floating litter in rivers and canals. These methods require the availability of large amount of labeled data for training. The labeling work is expensive and laborious, resulting in small open datasets available in the field compared to the comprehensive datasets for computer vision, e.g., ImageNet. Fine-tuning models pre-trained on these larger datasets helps improve litter detection performances and reduces data requirements. Yet, the effectiveness of using features learned from generic datasets is limited in large-scale monitoring, where automated detection must adapt across different locations, environmental conditions, and sensor settings. To address this issue, we propose a two-stage semi-supervised learning method to detect floating litter based on the Swapping Assignments between multiple Views of the same image (SwAV). SwAV is a self-supervised learning approach that learns the underlying feature representation from unlabeled data. In the first stage, we used SwAV to pre-train a ResNet50 backbone architecture on about 100k unlabeled images. In the second stage, we added new layers to the pre-trained ResNet50 to create a Faster R-CNN architecture, and fine-tuned it with a limited number of labeled images (≈1.8k images with 2.6k annotated litter items). We developed and validated our semi-supervised floating litter detection methodology for images collected in canals and waterways of Delft (the Netherlands) and Jakarta (Indonesia). We tested for out-of-domain generalization performances in a zero-shot fashion using additional data from Ho Chi Minh City (Vietnam), Amsterdam and Groningen (the Netherlands). We benchmarked our results against the same Faster R-CNN architecture trained via supervised learning alone by fine-tuning ImageNet pre-trained weights. The findings indicate that the semi-supervised learning method matches or surpasses the supervised learning benchmark when tested on new images from the same training locations. We measured better performances when little data (≈200 images with about 300 annotated litter items) is available for fine-tuning and with respect to reducing false positive predictions. More importantly, the proposed approach demonstrates clear superiority for generalization on the unseen locations, with improvements in average precision of up to 12.7%. We attribute this superior performance to the more effective high-level feature extraction from SwAV pre-training from relevant unlabeled images. Our findings highlight a promising direction to leverage semi-supervised learning for developing foundational models, which have revolutionized artificial intelligence applications in most fields. By scaling our proposed approach with more data and compute, we can make significant strides in monitoring to address the global challenge of litter pollution in water bodies.

1. Introduction

Litter pollution in water bodies is a challenging global concern, that negatively affects aquatic ecosystems and human livelihood (Bellou et al., 2021). Plastics are the most dominant form of litter, due to their extensive use and their persistence in aquatic environments (Lebreton et al., 2018). Kaandorp et al. (2023) estimated an initial amount of floating marine plastics of 3.2 million tonnes in 2020. Recent studies indicate that river systems act as plastic reservoirs, where the majority of plastics accumulates, and even retains for decades (van Emmerik et al., 2022). They become micro- and nanoplastics over the years, associated with severe environmental and health risks (Xu et al., 2024).

Regardless of the type of litter, detecting and quantifying floating litter accurately in rivers and waterways is necessary for assessing

[*] Corresponding authors.
E-mail addresses: T.Jia@tudelft.nl (T. Jia), r.taormina@tudelft.nl (R. Taormina).

https://doi.org/10.1016/j.watres.2024.122405
Received 28 April 2024; Received in revised form 27 August 2024; Accepted 5 September 2024
Available online 11 September 2024

environmental risks and designing intervention strategies (Bellou et al., 2021; Hurley et al., 2023). Traditional approaches include debris sampling and visual observation (Hurley et al., 2023). However, the labor-intensive procedures and specific requirements for sampling equipment may limit the applicability of debris sampling to various locations over extended periods of time (van Lieshout et al., 2020). While visual observation is effective, it is not suitable for continuous monitoring and may be dangerous during extreme events, e.g., flood (van Emmerik et al., 2023). Moreover, visual counting is challenging for human counters in rivers with high litter fluxes (van Lieshout et al., 2020). Given these limitations, an automatic and efficient litter detection approach is needed. Currently, deep learning methods, especially Convolutional Neural Networks (CNNs) have drawn significant research attention for developing efficient alternatives (Jia et al., 2023a). Several studies have demonstrated the effectiveness of these approaches for litter detection with various computer vision tasks (Jia et al., 2023a). For instance, van Lieshout et al. (2020) applied Faster R-CNN with InceptionV2 to detect plastic litter from camera images collected from waterways in Jakarta, Indonesia, obtaining a precision of over 68%. Renfei et al. (2023) collected data from cameras mounted at multiple locations in a water conservation demonstration zone in Deqing, China, and proposed an improved Single Shot MultiBox Detector network to detect floating items with an average accuracy of 91.1%.

While the current outcomes are promising, obtaining an accurate and robust deep learning model for detecting floating litter requires large quantities of annotated training data for supervised learning (Jia et al., 2023a). The manual labeling work is costly, time-consuming and relies on domain-specific knowledge on floating litter detection. While the community has released some open datasets, the amount of annotated data available is far below that of comprehensive datasets, e.g., ImageNet with over 14 million images and almost 20,000 categories (Deng et al., 2009). This may hinder achieving broad model generalization and effective transferability, which underpins robust and versatile computer vision systems for structural monitoring of floating litter.

To partially overcome this limitation, researchers usually used transfer learning approaches (Jia et al., 2023a; Wu et al., 2024). They usually involve (1) pre-training a base network on a base dataset and task (e.g., image classification on ImageNet), and (2) transferring the learned knowledge to a target network to be fine-tuned on a target dataset and task. In the base task, the first few layers of the base network extract generic low-level features (e.g., edges, lines, and corners), that generalizes to many datasets and tasks. The remaining layers extract more high-level, complex and abstract feature knowledge (e.g., object boundaries and contours), that specializes to a target dataset and task (Yosinski et al., 2014). While transfer learning is a powerful technique, its effectiveness declines when the base and target tasks become less similar (Yosinski et al., 2014). To develop deep learning models for floating litter detection, previous studies pre-trained models on comprehensive datasets (Jia et al., 2023b). However, the high-level features in these datasets have limited relevance with respect to floating litter imagery. This may hinder performances and generalization capability.

To address the constraints of supervised learning, the deep learning research community is increasingly investigating self- and semi-supervised learning methods due to their data efficiency and generalization capability (Liu et al., 2021; Reddy et al., 2018). Self-supervised learning operates by using the unlabeled input data to automatically generate its own labels, learning the underlying representations from the data itself without explicit guidance (Liu et al., 2021). More recently, contrastive self-supervision have gained momentum (Jaiswal et al., 2020). Contrastive self-supervision obtains representations by distinguishing between positive pairs (similar instances) and negative pairs (dissimilar instances) (Jaiswal et al., 2020). For example, the Simple framework for Contrastive Learning of visual Representations (SimCLR) generates two different views from each input image by performing data augmentation (Chen et al., 2020). The positive pairs include two augmented views from the same image, while the negative pairs are formed by sampling two augmented views from different images. Semi-supervised learning (SSL) enhances self-supervised pre-trained models regardless of the method used. SSL leverages a small amount of labeled data to address specific downstream tasks e.g., image classification and object detection (Reddy et al., 2018). Recent studies have shown that SSL methods outperform traditional supervised learning approaches for applications on large-scale datasets (e.g., ImageNet), as well as domain-specific applications, including agriculture (Güldenring and Nalpantidis, 2021). While SSL approaches are promising, they have not been applied to detect floating litter.

In this paper, we proposed a two-stage semi-supervised learning method based on the Swapping Assignments between multiple Views of the same image (SwAV) for detecting floating litter in (fresh)water bodies. We developed and validated the methodology for images collected in canals and waterways of the Netherlands, Indonesia, and Vietnam. Furthermore, we assessed the transferability of low-level and high-level representations learned via SwAV pre-training. The goal of this study is to help understand whether SSL can lead to the development of foundational models capable of better generalization across multiple locations with limited data available for fine-tuning (Oquab et al., 2023; Jakubik et al., 2023), through the aforementioned evaluation. Models extracting relevant high-level feature representations, thus requiring little or no fine-tuning, are crucial to develop litter monitoring strategies at scale (Jia et al., 2023a).

2. Case studies and related datasets

We trained the SSL method using data from three locations: (1) The TU Delft - Green Village (TUD-GV), the Netherlands (Jia et al., 2023b), (2) Oostpoort, the Netherlands, and (3) Jakarta, Indonesia (van Lieshout et al., 2020). Moreover, we tested the generalization capability of our method using images captured in three other locations: (1) Amsterdam and (2) Groningen, the Netherlands, and (3) Ho Chi Minh City, Vietnam. Table 1 summarizes the detailed information of these datasets. All data used in experiments, including images and bounding box annotations, is publicly accessible (see Data Availability Statement). The detailed information on the actual data used in experiments can be found in Section 4.

2.1. The TU Delft - Green Village dataset

The TUD-GV dataset includes nearly 10,000 images, introduced in our previous study (Jia et al., 2023b). These images were captured by two action cameras and a phone from semi-controlled experiments conducted during 10 days in February and April 2021, in a small drainage canal in the TU Delft Campus, the Netherlands. These images contain floating litter under two different weather condition (sunny and cloudy), taken from two device heights above the water surface (2.7 m and 4.0 m) and two viewing angles (0 and 45 degrees).

2.2. The Oostpoort dataset

We generated the Oostpoort dataset from experiments conducted during 26 days from February to March 2022, in a canal at Oostpoort, Delft, the Netherlands. We collected data employing action cameras with a viewing angle of 0 degree. Fig. A.1 shows monitoring setups including cameras mounted outside the windows of a tower at Oostpoort. We recorded video sequences with a time-lapse recording (1 image/30 s) and a FPS (frame per second) of 17.98. We generated the Oostpoort dataset by saving images from these videos. Examples of images can be found in Fig. A.2. Some images in this dataset contain fauna and various extents of organic material (e.g., leaves and branches), that increases the complexity of the environment owing to their diverse range of color patterns, shapes and sizes. Organic material and floating litter clutter together in garbage patches in some images, making litter harder to detect (van Lieshout et al., 2020).

Table 1
Details on case study locations and related imagery.

Name	Collection location	Collection device	Image resolution (pixel × pixel)	Device height (m)	No. images[a]
TU Delft - Green Village	Delft, the Netherlands	GoPro Hero 4, GoPro MAX 360	1920 × 1080	2.7	1501
Oostpoort	Delft, the Netherlands	GoCam3, GoPro MAX 360	3840 × 2160, 1920 × 1440	5	562
Jakarta	Jakarta, Indonesia	Dahua Easy4ip	2560 × 1440, 1920 × 1080	4.5	526
Amsterdam	Amsterdam, the Netherlands	GoPro Hero 10	5568 × 4176	1–2	9
Groningen	Groningen, the Netherlands	Obscape HQ	2592 × 1944	4	63
Ho Chi Minh City	Ho Chi Minh City, Vietnam	GoPro Hero 11, DJI Phantom 4 Pro	5568 × 4872, 5464 × 3070	7.4–18.6 (cameras) 11–14 (drones)	27

[a] In this column, we only reported the number of images we used in experiments (see Section 4.1).

2.3. The Jakarta dataset

The Jakarta dataset is an object detection dataset with 1272 images and 14,968 annotated floating macroplastic litter items. van Lieshout et al. (2020) collected these images using a camera mounted on bridges at five different waterways in Jakarta, Indonesia, from 30 April to 12 May 2018. These images were taken from the view angle of 6 degrees, under various levels of organic material on river surface (i.e., no organic debris, some organic debris, and many organic debris). Most images (1108) have relatively still water surfaces, but the remaining images (164) have waves.

2.4. The Amsterdam dataset

We created the Amsterdam dataset from one experiment conducted on 1st March 2023, in canals and ponds at Amsterdam, the Netherlands. We recorded images using an action camera. Examples of these images can be found in Fig. A.3.

2.5. The Groningen dataset

We conducted several experiments in a canal in Groningen, the Netherlands, in 2023. Fig. A.4 shows monitoring setups including security cameras mounted on a bridge. We recorded images with a time-lapse recording (1 image/6 s). Examples of images are shown in Fig. A.5.

2.6. The Ho Chi Minh City dataset

The Ho Chi Minh City dataset with 15,495 images was generated from experiments conducted during 8 weeks from February to April 2023, at five locations of the Saigon river at Ho Chi Minh City, Vietnam. They were collected by bridge-mounted cameras and drones that flew across the river width. Examples of images are shown in Fig. A.6.

3. Methodology

3.1. Overview of the semi-supervised learning approach

We propose a two-stage semi-supervised learning method for detecting floating litter based on Swapping Assignments between multiple Views of the same image (SwAV). The approach includes a self-supervised learning stage and supervised learning stage. Fig. 1 shows the schematic illustration of the SSL method. In the first stage, we used SwAV to pre-train a ResNet50 network (He et al., 2016) with a large quantity of unlabeled data. To obtain the final model, we first created a Faster R-CNN architecture for object detection (Ren et al., 2015) by adding extra deep learning layers after the pre-trained ResNet50. Then, we fine-tuned the resulting model using a limited amount of labeled data to perform the specific litter detection downstream task. We describe SwAV and Faster R-CNN in Sections 3.2 and 3.3, respectively. Section 3.4 presents details on the implementation of the self-supervised pre-training methods, while the supervised stage is illustrated in Section 3.5.

3.2. Swapping Assignments between multiple Views of the same image (SwAV)

SwAV is a cluster-based self-supervised contrastive learning method (Caron et al., 2020). Models learn the underlying representations from the data by performing a clustering assignment prediction between various augmentations (or “views”) of the same input image. Fig. 2 shows the schematic illustration of SwAV. The process begins with data augmentation (e.g., multi-crop and flipping) to generate multiple views of the input image X. In Fig. 2, we only show the multi-crop augmentation method, that crops an image randomly into two global views with standard resolution crops (e.g., 224 × 224 pixels) and several local views with smaller resolution crops (e.g., 96 × 96 pixels). For simplicity, we only present two views (x_1, x_2). These views are processed by the same encoder network f_θ (e.g., ResNet50) followed by a projection head (e.g., 2-layer multilayer perceptron) to generate two corresponding feature vectors (z_1, z_2). To perform the online clustering assignment, SwAV uses the Sinkhorn–Knopp algorithm (Cuturi, 2013) to map the feature vectors to a set of prototypes C comprising K prototype vectors. Each prototype represents a cluster in the feature space. This operation results in the generation of the codes Q_1 and Q_2. The uniqueness of SwAV lies in its “swapped” prediction mechanism. Here, the code Q_2, derived from the view x_2, is predicted using the characteristics of the view x_1 and vice versa. This prediction method leverages the inherent similarities between the views, as they originate from the same image. Consequently, SwAV refines its learning of data attributes by forecasting the code of one image view based on the features of its counterpart. Appendix B presents more detailed information of SwAV.

3.3. Faster R-CNN for litter detection

Fig. 3 shows the detailed architecture of the Faster R-CNN with a ResNet backbone. The Faster R-CNN includes four modules: (1) feature extraction; (2) object proposal generation; (3) Region of Interest (RoI) pooling; and (4) classification with a confidence level and location prediction. Confidence refers to the probability assigned by the Faster R-CNN when classifying each bounding box. Appendix B presents more detailed information of the Faster R-CNN.

The ResNet mainly includes two parts: (i) convolutional blocks Conv1 to Conv4, and (ii) Conv5 (He et al., 2016). Both parts are pre-trained by SwAV in the self-supervised learning stage. Then, the Faster

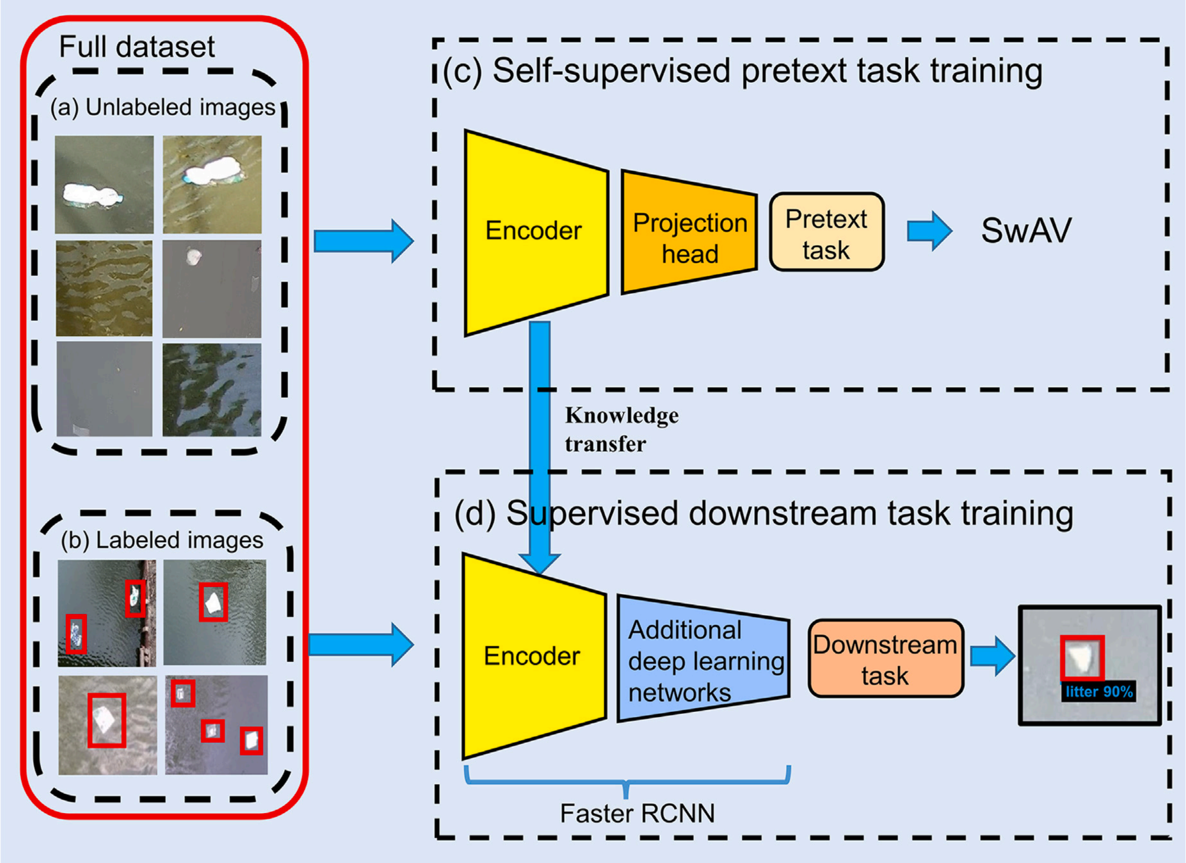

Fig. 1. The schematic illustration of the proposed two-stage semi-supervised learning method. In the self-supervised learning stage (c), we used SwAV to pre-train a ResNet50 encoder network combined with a projection head, using a large number of unlabeled images (a); Then, we added additional deep learning network to ResNet50 backbone to create a Faster R-CNN architecture. In the supervised learning stage (d), we fine-tuned the Faster R-CNN to learn a specific litter detection downstream task in a supervised manner, using a limited amount of labeled data (b).

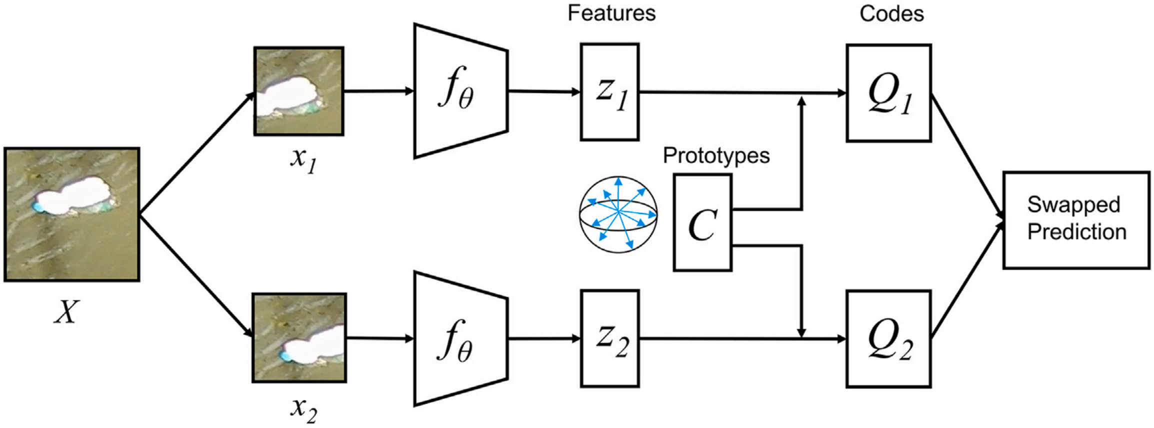

Fig. 2. The schematic illustration of SwAV adapted from Caron et al. (2020). First, each image X is augmented into two different views (x_1, x_2), that are processed by the encoder f_θ to obtain two feature vectors (z_1, z_2). Then, the codes of these two features (Q_1, Q_2) are computed by mapping them to prototypes C. Finally, SwAV learns data representations by solving a "swapped" prediction problem, where the code Q_2 is predicted using the view x_1 and vice versa.

R-CNN is constructed by using Conv1 to Conv4 as the backbone and adding Conv5 after the RoI pooling layer.

3.4. SwAV pre-training

To evaluate the benefits of self-supervised pre-training, we used two pre-training methods for all experiments: (1) SwAV-FTAL, and (2) SwAV-Scratch (Jia et al., 2023b). The SwAV-FTAL method first initializes the ResNet backbone with ImageNet weights, and then uses SwAV to fine-tune all the layers (FTAL) of the backbone on the unlabeled images. ImageNet weights used in this study were created by training the ResNet50 on 1.2 million images (1000 categories) from the full ImageNet dataset. We selected ImageNet weights since transferring features learned from the ImageNet image classification task to other

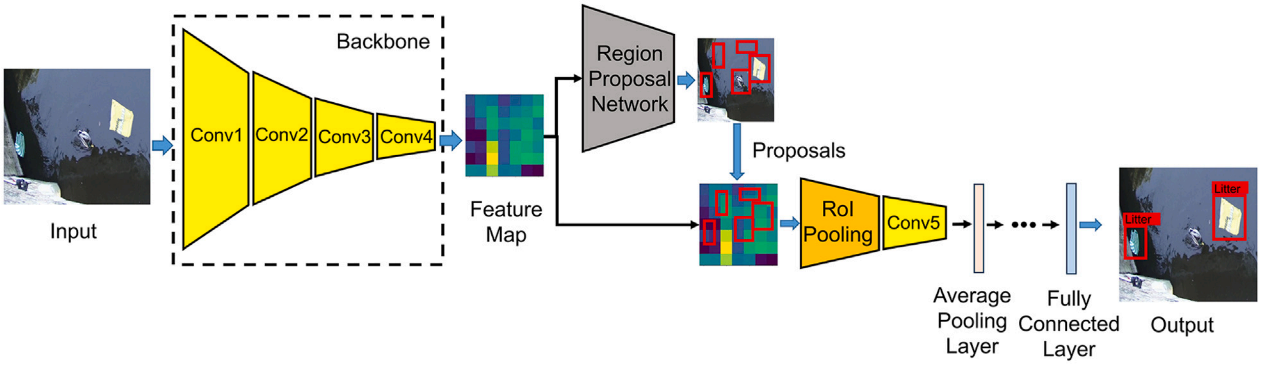

Fig. 3. The schematic illustration of the Faster R-CNN with ResNet backbone. The basic ResNet (yellow blocks) mainly includes two parts: (1) convolutional blocks Conv1 to Conv4, and (2) Conv5. In the first stage of the Faster R-CNN, the backbone first extracts feature maps from the input data. Then, the Region Proposal Network produces region proposals from these feature maps. Furthermore, the feature maps and region proposals are fed into the RoI Pooling layer, that converts the feature maps of proposals into fixed size feature maps for the final classification and location prediction in the second stage. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

domain tasks is a widely used approach to detect floating litter (Jia et al., 2023a). The SwAV-Scratch method uses SwAV to pre-train the ResNet50 from scratch. It involves initializing the ResNet50 backbone with random weights, and then using SwAV to pre-train all the layers of the backbone on the unlabeled images.

3.5. Fine-tuning for litter detection

To perform the litter detection downstream task, we fine-tuned Faster R-CNN architectures built on the pre-trained ResNet50 backbone. We compared two different approaches for fine-tuning, that entail freezing either 4 convolutional blocks (F4, from Conv1 to Conv4 in Fig. 3) or 2 (F2, Conv1 and Conv2) of the ResNet backbone, respectively. During fine-tuning, only the unfrozen layers of the Faster R-CNN are updated. The F2 method is a common method used to transfer low-level feature knowledge learned from pre-training to the downstream task. In contrast, the F4 method transfers both low-level and high-level feature knowledge. In situations where only a small dataset is available for model fine-tuning, maintaining relevant high-level features becomes crucial as it drastically reduces the number of weights to fine-tune. By examining the F4 modality, we aim to evaluate whether the high-level features learned via SwAV pre-training enhances the model's generalization capabilities in data scarce conditions. This investigation can help understand whether this approach can lead to the development of foundational models for litter quantification across multiple locations (Jia et al., 2023a; Oquab et al., 2023).

4. Experiments

We conducted multiple experiments to investigate the potential of SSL for floating litter detection. We evaluated both *in-domain* as well as *out-of-domain* generalization capability. In-domain generalization refers to the model performance on new, unseen images from the same geographic locations, while out-of-domain generalization refers to unseen images from other geographic locations. We compared the results with those obtained from a supervised learning benchmark, providing a robust reference point. Additionally, we investigated how the litter detection performance varies with the availability of labeled data for fine-tuning. This aspect is crucial for assessing the models' practical applicability in scenarios with limited annotated resources. Complementing this analysis, we evaluated the relevance of low-level and high-level representations learned from SwAV pre-training with respect to generalization. This examination can share further insights on the suitability of SSL for developing large-scale monitoring networks for quantifying floating litter across multiple locations.

Table 2
TU Delft Green Village (TUD-GV), Oostpoort (Delft, Netherlands) and Jakarta datasets.

	Subset			Total
	TUD-GV	Oostpoort	Jakarta	
Total images	1501	562	526	2589
Total image tiles	44,188	71,445	16,762	132,395
No. image tiles with litter annotated	1969	401	1399	3769
No. annotated litter items	2542	457	2531	5530

4.1. Data selection

We created the Delft-Jakarta dataset by selecting random images from the TUD-GV, Oostpoort, and Jakarta locations, as reported in Table 2. These images were sliced into tiles with a standard size of 224 × 224 pixels, to match the input dimensions of ResNet50 (Pham et al., 2021). Example image tiles are shown in Fig. A.7. We used the Delft-Jakarta dataset to train and validate the models, and to test their in-domain generalization performance. We extracted a total of 132,395 image tiles from the Delft-Jakarta datasets. These were used to randomly create the non-overlapping subsets for self-supervised pre-training (116,286 tiles), supervised fine-tuning (1756 tiles), validation (164 tiles), and testing (14,189 tiles), detailed in Table 3. Almost 90% of the tiles were used for self-supervised pre-training with SwAV ($\text{Train}_{\text{self}}$). These tiles have no labels. We used a maximum of 1756 image tiles for supervised fine-tuning ($\text{Train}_{100\%}$), containing a total of 2628 annotated litter items. The annotations are bounding boxes representing the location of floating litter items, without further categorization. To better assess model performance with respect to the availability of labels, we created six smaller fine-tuning datasets by reducing the number of tiles and annotations down to 5% ($\text{Train}_{80\%}$ to $\text{Train}_{5\%}$). We used a maximum of 164 image tiles and 282 annotations for model validation ($\text{Validation}_{100\%}$), maintaining a 9-to-1 ratio with respect to the data available for fine-tuning. For consistency, we created six smaller validation datasets ($\text{Validation}_{80\%}$ to $\text{Validation}_{5\%}$). We created a Test dataset by including 1849 tiles with 2620 annotations. To better evaluate the models performance with respect to false positives, we included 12,340 image tiles with no floating litter.

To evaluate out-of-domain generalization, we sliced randomly selected images from the Amsterdam, Groningen and Ho Chi Minh City datasets, as detailed in Table 4. The tiles in these subsets contain both images with annotated litter and without litter. Example image tiles are shown in Fig. A.8.

Table 3
The Delft-Jakarta subsets used in the experiments.

Learning method	Training dataset			Validation dataset			Test dataset			No. tiles without litter
	Name	No. annotated litter items	No. tiles	Name	No. annotated litter items	No. tiles	Name	No. annotated litter items	No. tiles	
Self-supervised	$\text{Train}_{\text{self}}$	0	116,286							
Semi-supervised and supervised	$\text{Train}_{100\%}$	2628	1756	$\text{Validation}_{100\%}$	282	164	Test	2620	1849	12,340
	$\text{Train}_{80\%}$	2076	1389	$\text{Validation}_{80\%}$	224	117				
	$\text{Train}_{60\%}$	1594	1059	$\text{Validation}_{60\%}$	171	100				
	$\text{Train}_{40\%}$	1013	702	$\text{Validation}_{40\%}$	115	70				
	$\text{Train}_{20\%}$	527	368	$\text{Validation}_{20\%}$	62	55				
	$\text{Train}_{10\%}$	282	180	$\text{Validation}_{10\%}$	27	22				
	$\text{Train}_{5\%}$	124	84	$\text{Validation}_{5\%}$	13	9				

Table 4
The Amsterdam, Groningen and Ho Chi Minh City datasets used to evaluate out-of-domain generalization.

	Subset			Total
	Amsterdam	Groningen	Ho Chi Minh City	
Total images	9	63	27	99
Total image tiles	3623	5544	13,032	22,199
No. image tiles with litter annotated	152	439	766	1357
No. annotated litter items	204	525	1091	1820
No. image tiles without litter	3471	5105	12,266	20,842

4.2. Developed models and experiments

For brevity, we indicated models built via pre-training with the SwAV-FTAL method and fine-tuning with the F2 method, as SwAV-FTAL-F2 across all experiments. Other models are named in the same way, e.g., SwAV-FTAL-F4, SwAV-Scratch-F2, and SwAV-Scratch-F4. We compared the effectiveness of SSL against baseline supervised learning models which are developed without the SwAV pre-training step. These models are Faster R-CNNs fine-tuned on labeled data, built on ResNet50 backbones initialized with ImageNet weights (see Fig. 1(b) and (d)). For consistency, we used two types of baseline models: (1) Baseline-F2, and (2) Baseline-F4, that uses the F2 and F4 methods for fine-tuning, respectively.

We developed all models by using the Delft-Jakarta subsets in Table 3. Specifically, we built the SSL models by first pre-training a ResNet50 encoder with a projection head of 2-layer multilayer perceptron on the $\text{Train}_{\text{self}}$ subset. We then fine-tuned the Faster R-CNN derived from the ResNet50 backbone on all the seven available subsets for supervised learning, i.e., $\text{Train}_{100\%}$ to $\text{Train}_{5\%}$. We performed model validation on the respective Validation subsets. The Baseline supervised learning models are developed in the same fashion, but without SwAV pre-training. The Delft-Jakarta Test subset is used for evaluating the in-domain generalization. On the other hand, we evaluated out-of-domain generalization using the image tiles from Amsterdam, Groningen and Ho Chi Minh City detailed in Table 4. For out-of-domain generalization, we tested only the models fine-tuned using the maximum amount of the Delft-Jakarta labeled data, i.e., $\text{Train}_{100\%}$. We used the SwAV-FTAL-F2, SwAV-Scratch-F2 and Baseline-F2 methods to evaluate the quality of transferred low-level representations. Similarly, we investigated the relevance of high-level representations by implementing the SwAV-FTAL-F4, SwAV-Scratch-F4 and Baseline-F4 methods.

4.3. Performance assessment

To assess model performance of floating litter detection, we used two commonly employed metrics: (i) AP50, representing the Average Precision (AP) with an Intersection over Union (IoU) threshold of 50% and (ii) F1-score computed using the same threshold (Jia et al., 2023a). The IoU measures the ratio of the overlap area of prediction and ground truth to their union area, which is described as follows Padilla et al. (2020):

$$IoU = \frac{area(bbox_{pred} \cap bbox_{gt})}{area(bbox_{pred} \cup bbox_{gt})} \tag{1}$$

where $bbox_{pred}$ and $bbox_{gt}$ are the predicted bounding box and the ground-truth bounding box, respectively. The larger the IoU, the greater the overlap of these two bounding boxes. After setting an IoU threshold, we can compute the elements of the confusion matrix for the object detection task. For each ground-truth box, we have a True Positive (TP) if there is at least one overlapping predicted box with IoU equal or above the threshold. Predicted boxes overlapping the ground-truth with IoU less than the threshold are marked as False Positives (FP). If more bounding boxes sufficiently overlap with the ground truth, we mark as TP only the one with the highest confidence (Dollár and Lin, 2014). The others are marked as FP. FPs also include incorrect detection of nonexistent objects. False Negatives (FN) are the undetected ground-truth bounding boxes.

The AP is the average precision of the models for a given IoU threshold. It is computed as the area under the precision–recall curve (Jia et al., 2024). Appendix B presents more detailed information of the precision–recall curve. The precision p and recall r are expressed as follows:

$$p = \frac{TP}{TP + FP} \tag{2}$$

$$r = \frac{TP}{TP + FN} \tag{3}$$

Precision measures the accuracy of the positive predictions, denoted by the ratio of correctly identified positive cases (TP) to the total number of cases identified as positive (TP + FP). On the other hand, recall is the ratio of correctly identified positive cases (TP) to the actual total positive cases (TP + FN). It assesses the model's ability to detect all relevant instances. After creating the precision–recall curve, we can calculate AP by integrating the area under it:

$$AP = \int_0^1 p(r)dr \tag{4}$$

The F1-score is computed as the harmonic mean of p and r, is calculated as follows:

$$F1 - score = \frac{2 * p * r}{p + r} \tag{5}$$

More detailed information of the AP and F1-score can be found in Appendix B.

4.4. Training setup and procedure

We implemented all experiments with *Python 3.8.16* and *PyTorch 1.8.1*, in combination with the *VISSL* (Goyal et al., 2021) and the

Detectron2 (Wu et al., 2019) libraries. We trained and tested all models on a NVIDIA Tesla V100S PCIe GPU (32 GB) (Delft High Performance Computing Centre (DHPC), 2022). We used default *VISSL* hyperparameters for SwAV pre-training, including a cluster with 3000 prototypes. We pre-trained for 100 epochs, using the SGD optimizer with cosine annealing learning rate scheduling (Loshchilov and Hutter, 2016), with the initial rate of 0.075 and the minimum value of 7.5 × 10^{-5}. We applied four default *VISSL* data augmentation methods: (1) multi-crop with 8 views (2 × [224 × 224] + 6 × [96 × 96]), (2) horizontal flipping, (3) color distortion, and (4) Gaussian blur. In the supervised learning stage, we fine-tuned the Faster R-CNN with default *Detectron2* hyperparameters, including an SGD optimizer with a fixed learning rate of 0.02, a weight decay of 0.0001 and a momentum of 0.9. Before making the final predictions and computing performances, we refined the output bounding boxes via Non-Maximum Suppression (NMS) (Hosang et al., 2017). Appendix B presents the detailed information of NMS. For all experiments and developed models, we used a commonly employed IoU NMS threshold value of 0.5. We implemented the Baseline methods using the same fine-tuning hyperparameters. We trained all models for 100 epochs, saving the learned parameters yielding the highest validation accuracy.

5. Results and discussion

5.1. In-domain detection performances for varying data availability

Fig. 4 compares the AP50 detection performance on Delft-Jakarta Test subset for the SwAV-FTAL-F2, SwAV-Scratch-F2 and Baseline-F2 methods. The three methods perform similarly when relatively more data is available for fine-tuning (i.e., $\text{Train}_{60\%}$ to $\text{Train}_{100\%}$ subsets), with an AP50 ranging from 62.8% to 65.8%. When less labeled data is available (i.e., $\text{Train}_{5\%}$ to $\text{Train}_{40\%}$ subsets), the SwAV-FTAL-F2 method performs best in most cases, obtaining an AP50 ranging from 44.3% to 60.4%. This yields a slight improvement in AP50 of up to 2.3%, compared to the baseline method (AP50 = 44.4%~59.3%). The SwAV-Scratch-F2 method performs worst (AP50 = 37.3%~57.4%), yielding a slight decrease in AP50 varying from 5% to 7.1%, compared to the baseline method in half of these cases. Fig. 4 also indicates a general upward trend in performance with increasing amount of labeled data, regardless of the approach used. The observed performance plateau could be attributed not only to the limited size of our labeled dataset, but also to the lack of hyper-parameter tuning and the fact that only a single training run was conducted, due to computational limitations (SwAV pre-training time: 12 min/epoch). The stochastic nature of neural network training means that multiple runs yields different results, possibly influencing the observed performance ceiling (Punjani and Fleet, 2021).

At first glance, these results suggest that transferring low-level representations learned by SwAV on unlabeled, but relevant data, does not yield substantial improvements with respect to simple transfer from ImageNet. In particular, learning from scratch via SwAV hinders performance when little data is available for fine-tuning, although the situation rapidly improves when more labels are available. However, one must consider that the ImageNet dataset (1.2 million images) contains over 10 times more images than the $\text{Train}_{\text{self}}$ subset used for SwAV pre-training. The availability of large amounts of data enables ResNet50 to learn robust low-level features that are used by the deeper layers fine-tuned for the downstream litter detection task with Faster R-CNN. Furthermore, the ImageNet pre-trained weights are the product of extensive optimization on substantial computational resources, which contrasts sharply with our constrained SwAV pre-training that involved limited runs and no hyper-parameter tuning. Despite these limitations, we achieved comparable results, showcasing the potential effectiveness of our methodology. Better performances can be obtained by scaling the datasets and the computational efforts. Literature reports strong increases in SSL performances with larger SwAV pre-training datasets, e.g., from 1.2 million to 14 million to 1 billion (Goyal et al., 2022).

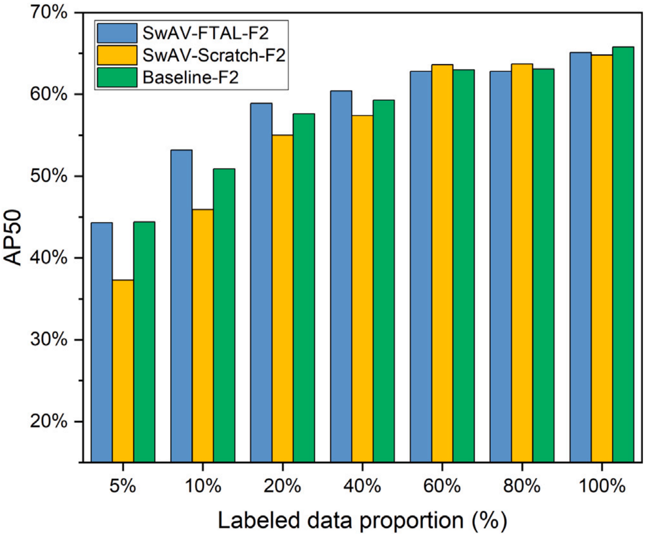

Fig. 4. AP50 detection performance of the SwAV-FTAL-F2, SwAV-Scratch-F2 and Baseline-F2 methods on the Test subset with different proportion of labeled data for fine-tuning.

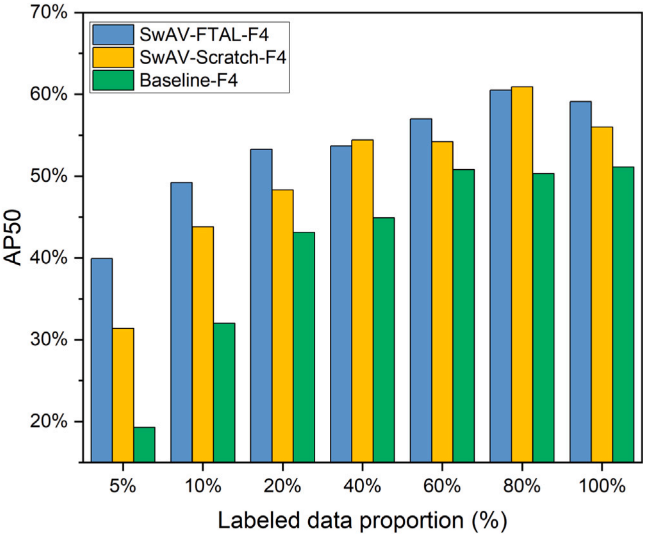

Fig. 5. AP50 detection performance of the SwAV-FTAL-F4, SwAV-Scratch-F4 and Baseline-F4 methods on the Test subset with different proportion of labeled data for fine-tuning.

Regardless of the above limitations in our SwAV implementation, the SSL methods outperform the baseline when considering other metrics. Table 5 reports the Test dataset confusion matrix, precision, recall and F1-score for the three methods fine-tuned on $\text{Train}_{100\%}$. The Baseline-F2, yields overall marginally better recall (0.74 vs. 0.71), but substantially lower precision (0.48 vs. 0.57) than the SSL methods. This results in a lower F1-score (0.58 vs. 0.63) due to a much higher number of FPs. Similar worse performances are found for images without litter, where the number of FPs of the baseline is around double that of the SSL methods.

The benefits of SwAV pre-training clearly emerge when preserving the high-level feature representations, as reported in Fig. 5. The results show that both the SwAV-FTAL-F4 and SwAV-Scratch-F4 methods significantly outperform the Baseline-F4 benchmark, regardless of the amount of labeled data available for fine-tuning. The SwAV-FTAL-F4

Table 5
Confusion matrix, Precision, Recall and F1-score on the Delft-Jakarta Test subset for models fine-tuned on the $\text{Train}_{100\%}$ dataset. False positives are also reported for 12,340 additional images without litter.

Method	Test dataset						Images without litter
	TP	FN	FP	Precision	Recall	F1-score	FP
SwAV-FTAL-F2	1850	770	1391	**0.57**	0.71	**0.63**	3666
SwAV-Scratch-F2	1832	788	**1359**	**0.57**	0.70	**0.63**	**3594**
Baseline-F2	**1926**	**694**	2093	0.48	**0.74**	0.58	7453
SwAV-FTAL-F4	**1775**	**845**	2024	0.47	**0.68**	0.55	6788
SwAV-Scratch-F4	1680	940	**1373**	**0.55**	0.64	**0.59**	**3192**
Baseline-F4	1590	1030	2296	0.41	0.61	0.49	9167

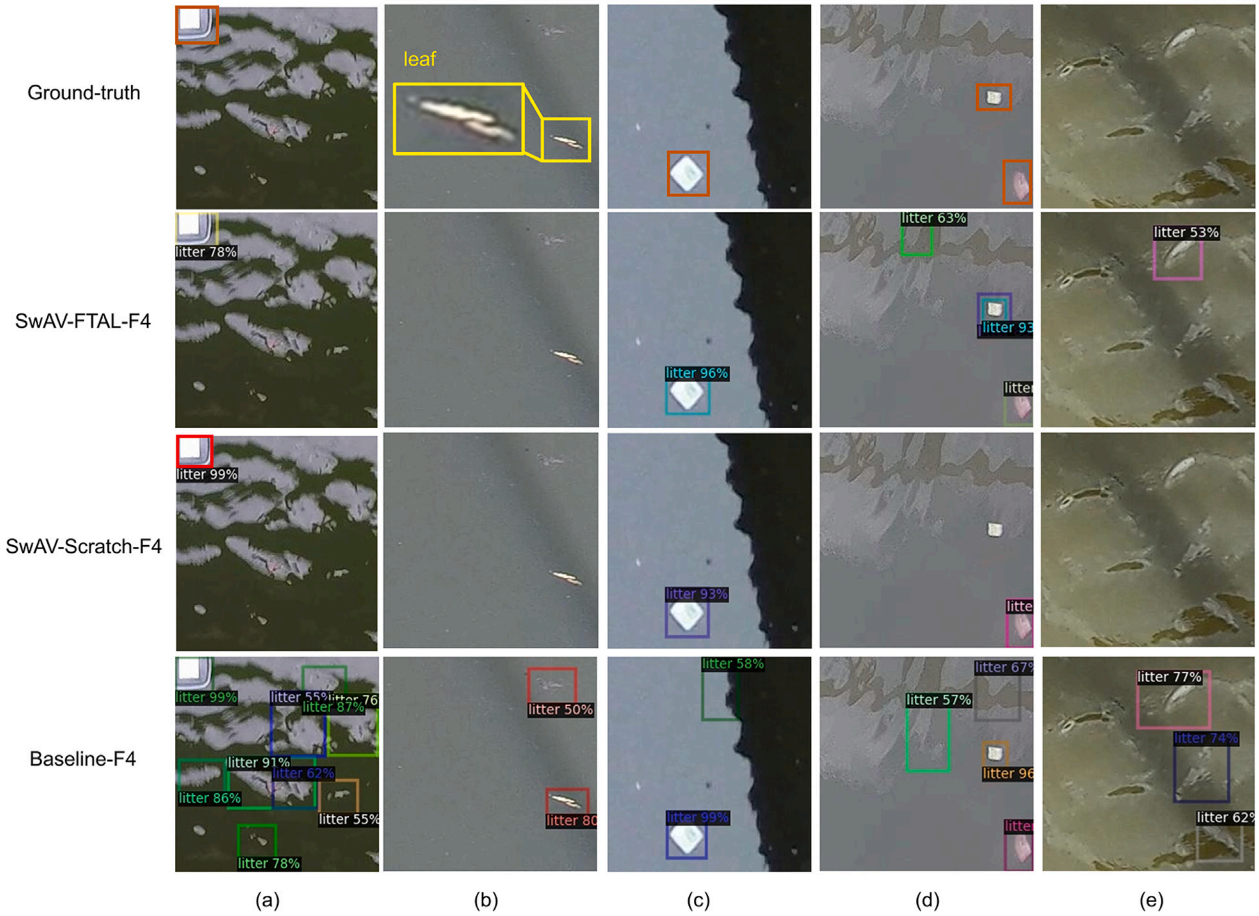

Fig. 6. Example of predicted bounding boxes for the Faster R-CNN on the Delft-Jakarta Test subset and images without litter using (1) SwAV-FTAL-F4, (2) SwAV-Scratch-F4, and (3) Baseline-F4 methods. The models were fine-tuned on the $\text{Train}_{100\%}$ subset. Common misdetections of Baseline-F4 include the identification of waves ((a) and (e)), organic materials (b), and reflection of structures on banks (c) and bridge (d) as litter. Ground-truth litter is shown in red bounding boxes in the top row. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

method performs best in most cases, achieving an AP50 ranging from 39.9% to 60.5%. The SwAV-Scratch-F4 method performs worse when very limited labeled data is available, but then achieves comparable or higher scores, with the highest reported score of 60.9% for $\text{Train}_{80\%}$. The baseline method obtains AP50 varying between 19.3% and 51.1%. These values are particularly low when little data is available for fine-tuning (i.e., Train_{5} and $\text{Train}_{10\%}$ subsets), where SwAV-FTAL-F4 and SwAV-Scratch-F4 yield improvements in AP50 of up to 20%. The SSL approaches only requires 20% of the labeled data (527 annotated litter items) to achieve similar or better performance (AP50 = 53.3%) than what obtained by the baseline method with 100% of labeled data (2628 annotated litter items, AP50 = 51.1%). Similar to the plateau discussed in Fig. 4, the drop in performance when moving from $\text{Train}_{80\%}$ to $\text{Train}_{100\%}$ can be linked to the limited overall size of our labeled dataset, the randomness of single runs, and lack of hyper-parameterization. For example, Bolton et al. (2023) reported a similar phenomenon caused by the lack of hyper-parameterization. They trained DL models to identify aircraft engine types with a learning rate of 0.01, but the performance drops as the size of training data. However, when setting the learning rate to 0.001, they found the performance improvement with the increase of training dataset size.

The better performance of the SSL methods are further detailed in Table 5 for the three models fine-tuned on $\text{Train}_{100\%}$. The Baseline-F4 performs the worst in all metrics, with a substantial decrease in TP, followed by a detrimental increase in both FN and FP. Interestingly, the SwAV-Scratch-F4 method retains the highest F1-score (0.59), due to a substantially lower number of FP. The lower precision of Baseline-F4 suggests that the high-level features learned from ImageNet are not sufficiently relevant to the specific nuances of the litter detection task. Visual inspection of the predicted bounding boxes highlights that Baseline-F4 wrongly identifies waves, organic material, and the reflection of structures on banks and bridge as litter, as shown for example in Fig. 6. SwAV pre-training helps the models distinguish between the features of litter and non-litter items, as well as background characteristics. ImageNet initialization may partially hinder this process if insufficient data is available for fine-tuning, as hinted by the lower precision of SwAV-FTAL-F4 with respect to SwAV-FTAL-F2 and SwAV-Scratch-F4. Nonetheless, initializing SwAV with ImageNet weights seems useful when labeled data is particularly scarce (e.g., $\text{Train}_{5\%}$ to $\text{Train}_{20\%}$ subsets).

5.2. Out-of-domain generalization capability

The results illustrated in Section 5.1 suggest that when sufficient fine-tuning data is available, the SSL approach does not offer significant in-domain generalization advantages with respect to simple transfer of ImageNet pre-trained models. This can change by overcoming the discussed constraints on the small datasets used for SwAV pre-training and the limited computational resources. Despite these limitations, the scenario shifts favorably towards SSL when considering out-of-domain generalization, as done for zero-shot floating litter detection to the unseen locations in Amsterdam, Groningnen and Ho Chi Minh City. As shown in Fig. 7 for all models fine-tuned on $\text{Train}_{100\%}$, SwAV pre-trained methods consistently match or surpass baseline performances. For example, in the Amsterdam dataset, both SwAV-FTAL-F4 and Baseline-F2 achieved a AP50 of around 45%. In Groningen, SwAV-FTAL-F4 outperforms the best baseline model by 12.7%, reaching an AP of 49.5%. In Ho Chi Minh City, SwAV-FTAL-F2 exceeds the baseline by over 7.5% with a 20.6% AP50. Further analysis on the confusion matrices and related metrics in Tables C.1–C.3. reinforces SwAV's advantage in out-of-domain scenarios. Except for Baseline-F2 in Groningen, which exhibits high precision and fewer FPs due to subpar sensitivity, the SSL models lead in all other metrics for all case studies. These better performances are reflected also in the visual inspection of the detections, done for SwAV-FTAL-F4 and Baseline-F4 on some example images of the three unseen case studies in Fig. 8. The baseline method displays fewer correct detection and increased misdetections, especially with respect to organic material, waves and other disturbances or reflective elements on the water surface. These findings collectively suggest that SwAV pre-training notably aids in adapting to new environments, particularly when retaining high-level features. The F4 SSL models are the best overall performers, despite we did not employ the best models emerging from the Delft-Jakarta Test dataset for the evaluation of out-of-domain generalization (i.e., those fine-tuned on $\text{Train}_{80\%}$). Expectedly, performance dips in more challenging conditions, e.g., in Ho Chi Minh City. Here, factors like lower resolution at the ground due to higher sensor elevation and the introduction of drone imagery, which were not part of the training dataset, further differentiate this dataset from the Delft-Jakarta dataset used for model development.

5.3. Towards foundational models for litter detection in water bodies

The development of large-scale monitoring networks for automated quantification of litter pollution in (fresh)water bodies requires models with superior generalization capabilities (Jia et al., 2023a). These models must predict accurately in a zero-shot or few-shot manner, quantifying litter with minimal prior data on specific locations. Numerous studies demonstrated the effectiveness of fine-tuning models pre-trained on general datasets like ImageNet and COCO (Lin et al., 2014) for litter detection at specific locations (van Lieshout et al., 2020; Wolf et al., 2020). However, these models falter to retain good performances when applied to varied locations, environmental conditions, and sensor settings (van Lieshout et al., 2020; Jia et al., 2023b). This limitation underscores the need for a more robust approach rooted in the development of foundational models.

Foundational models are a recent transformative paradigm in Deep Learning. By leveraging vast amounts of data via self-supervision, these models achieve remarkable general understanding and adaptability, which allows them to reach unprecedented performances when fine-tuned for specialized tasks. This paradigm shift is exemplified by the OpenAI GPT series, a family of self-supervised foundational models that, in their latest iterations, launched the current Artificial Intelligence (AI) revolution by enabling the development of ChatGPT via specialization (Brown et al., 2020; Achiam et al., 2023). Similar remarkable examples for computer vision exists, e.g., the DINOv2 model from Meta AI and INRIA (Oquab et al., 2023), or the Prithvi foundational model for Earth Observation developed by IBM and NASA on the Harmonized Landsat and Sentinel 2 dataset (Jakubik et al., 2023).

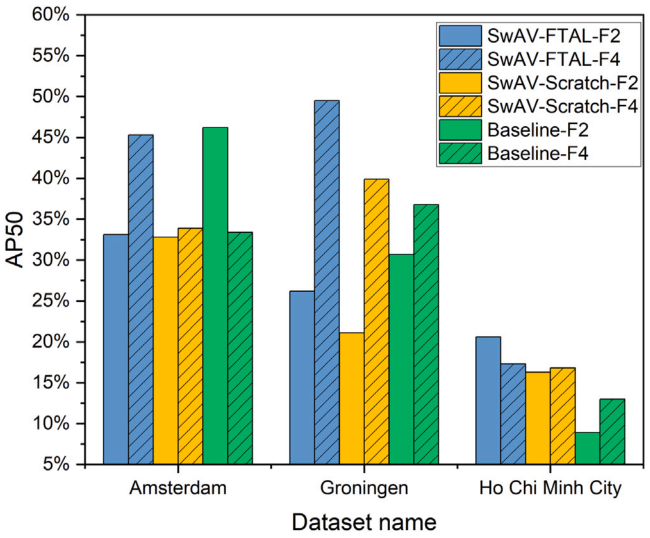

Fig. 7. Zero-shot generalization capability of the models fine-tuned on $\text{Train}_{100\%}$ for the three unseen locations: Amsterdam, Groningen, and Ho Chi Minh City.

We believe foundational models tailored for floating litter detection could significantly enhance our ability to monitor and mitigate this environmental issue at scale, whether from camera imagery or satellites. While DINOv2 or Prithvi have already shown promising results for fine-tuned critical applications, e.g., medical imaging and flood inundation mapping, there is evidence that restricting the focus of the datasets used for self-supervised pre-training can be more beneficial (Li et al., 2023; Huix et al., 2024).

The SSL methods we proposed consistently outperform the standard practice of simply using ImageNet pre-trained models for out-of-domain generalization, especially in the case of the SwAV-FTAL-F4 methods. The focus of this work was not to develop models for actual deployment in monitoring networks, but to show clear evidence that the features extracted by SSL are superior to those of a comprehensive dataset such ImageNet, as already reported in other fields (Huix et al., 2024). Using more data for fine-tuning, such as it was done in existing studies leveraging pre-trained models, yields better performances (Jia et al., 2023b).

More importantly, in our preliminary explorations, we used a dataset of around 100k images for SwAV pre-training, sourced from very few, albeit different, locations. Drawing parallels from other fields, we argue that scaling this approach is necessary to yield more robust models (Goyal et al., 2022). Importantly, the state-of-the-art performances obtained by DINOv2, partly due to SwAV mechanics, indicate that careful selection of data is more important than gathering billions of images. Nevertheless, to address a problem of global scale, we must significantly expand our dataset to include millions of images from diverse geographical locations. This approach not only enhances the model's effectiveness but also aligns with the principles of equitable artificial intelligence, ensuring the model's applicability across various global contexts (Manjarrés et al., 2021). Gathering vast quantities of

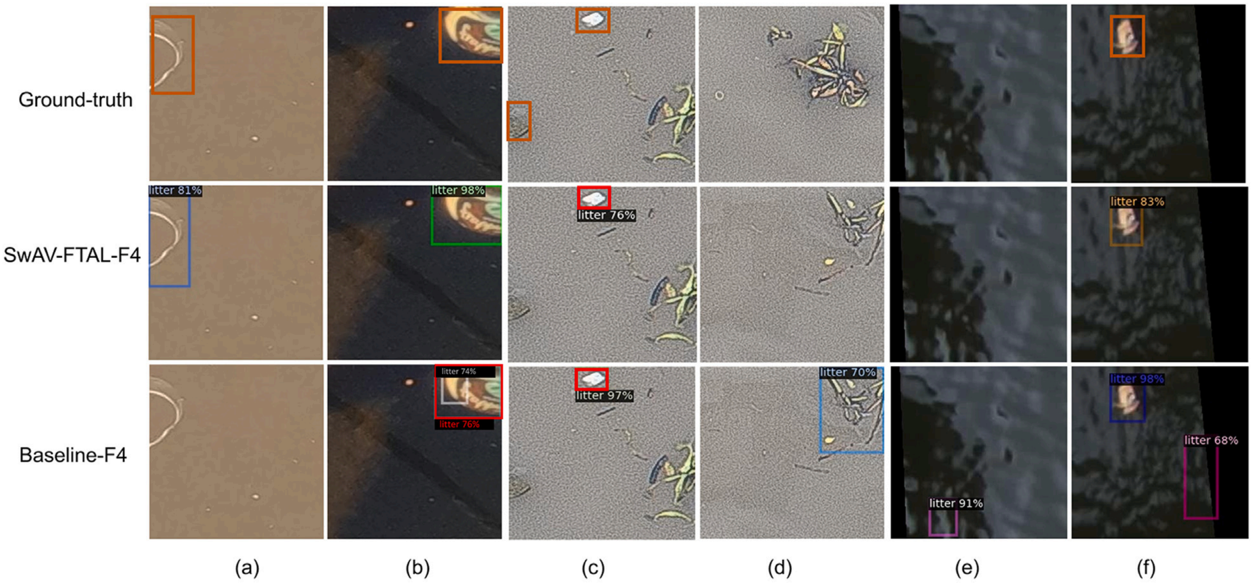

Fig. 8. Detection results of the Faster R-CNN with ResNet50 backbone on Amsterdam, Groningen, and Ho Chi Minh City subsets using SwAV-FTAL-F4 and Baseline-F4 methods. The models were fine-tuned on the Train$_{100\%}$ subset. Both methods can detect litter items in (b), (c) and (f), and only the SwAV-FTAL-F4 method can detect the litter item in (a). Common misdetection of the Baseline-F4 method includes identifying organic materials (d) and wave ((e) and (f)) as litter. Ground-truth litter is shown in red bounding boxes in the top row. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

diversified data is a necessary step, but not sufficient. Additional efforts must be directed towards implementation strategies, hyper-parameter optimization, and the selection of suitable Deep Learning architectures. For instance, considering the efficacy of transformers in state-of-the-art foundational models like GPT, DINOv2, and Prithvi, adopting similar architectures could be beneficial (Dosovitskiy et al., 2020).

6. Conclusions

Deep Learning methods for computer vision offer new opportunities to enhance floating litter detection in (fresh)water bodies. These methods process images and videos to quantify litter. However, traditional supervised learning requires extensive labeled data, a time-consuming and expensive process. Although transfer learning models trained on comprehensive datasets like ImageNet help reduce data requirements for specific locations, they lack the broader generalization essential for developing structural monitoring strategies operating at scale. To address this issue, we introduced a semi-supervised learning (SSL) approach based on SwAV, a self-supervised method that pre-trains Deep Learning models by discerning data patterns without requiring annotated images.

To demonstrate the suitability of this new approach, we carried out experiments on camera images from the Delft (the Netherlands) and Jakarta (Indonesia) using a Faster R-CNN with a ResNet50 backbone. We compared the performance of standard transfer learning from ImageNet against the use of SwAV pre-training on around 100k unlabeled images. All models were fine-tuned using a maximum of around 1.8k images from the same locations. Our results show that the SSL approach performs at par or better than the supervised learning benchmark in average precision and F1-score, when tested on unseen images gathered from the same locations of the training dataset. The improvements are more noticeable when less data (up to ≈200 images with around 300 annotated litter items) is available for fine-tuning and with respect to the prediction of false positives. More importantly, testing for zero-shot generalization capability on unseen locations in Ho Chi Minh City (Vietnam), Amsterdam and Groningen (Netherlands) shows the clear superiority of SSL. This is mainly due to the extraction of better high-level representations via SwAV pre-training on relevant unlabeled images. Better performances are reported when initializing the SSL models with ImageNet weights. While we tested this new approach only for river surfaces, it can also be applied to other freshwater bodies and extended to saltwater bodies, provided that images are captured using similar devices (i.e., static cameras).

This paper aims to contribute to pave the way for the development of self-supervised foundational models specifically for litter detection. This transformative approach can yield substantial impact as seen in other fields through foundational models like the GPT series, DINOv2, and Prithvi. Achieving this goal will involve a collective effort to gather a much broader range of images across the globe. Additionally, more focus is needed on thorough hyper-parameter optimization and effective implementation strategies, as well as exploring advanced Deep Learning architectures, e.g., transformers. Apart from the development of foundational models, other explorations could focus on a multi-class object detection downstream task aimed at identifying various litter categories of interests, e.g., bags and nets.

CRediT authorship contribution statement

Tianlong Jia: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Rinze de Vries:** Writing – review & editing, Supervision, Data curation. **Zoran Kapelan:** Writing – review & editing, Supervision, Funding acquisition. **Tim H.M. van Emmerik:** Writing – review & editing, Data curation. **Riccardo Taormina:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Author Rinze de Vries is employed by Noria Sustainable Innovators. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Data availability statement

The code for this study is available on https://github.com/TianlongJia/deep_plastic_SSL. The data used in this study including images and bounding box annotations are available for download at: (1) TU Delft - Green Village, https://doi.org/10.5281/zenodo.13730228; (2) Oostpoort, https://doi.org/10.5281/zenodo.13730298; (3) Amsterdam, https://doi.org/10.5281/zenodo.13730370; (4) Groningen, https://doi.org/10.5281/zenodo.13730384; and (5) Ho Chi Minh City, https://doi.org/10.4121/78bb4822-7b70-4632-887a-7cacd344024e.

Acknowledgments

The work is supported by China Scholarship Council (No. 202006160032) and the Directorate-General for Public Works and Water Management of The Netherlands (Rijkswaterstaat), The Netherlands. We are grateful to Paolo Tasseron for the Amsterdam dataset, and Tim Janssen, Louise Schreyers, and Khiet Bui for the Ho Chi Minh City dataset. We would like to thank Xueqin Chen, Jing Yu and Jingmin Long for fruitful scientific discussions. We acknowledge the use of computational resources of DelftBlue supercomputer (https://www.tudelft.nl/dhpc).

Appendix A. Dataset information and image examples

Appendix B. Methodology and evaluation metrics

B.1. Swapping Assignments between multiple Views of the same image (SwAV)

Two major core components of SwAV are clustering assignment and multi-crop augmentation strategy. SwAV's clustering assignment avoids the direct comparison of negative and positive pairs in contrastive learning. That reduces the computational overhead and potential noise introduced by large sets of negative samples, leading to more efficient and robust model training compared to other contrastive learning methods. The multi-crop augmentation strategy improves performance of self-supervised methods with only a small increase in the memory and computational cost. These allow SwAV outperform other recent and successful contrastive learning methods (e.g., the Simple framework for Contrastive Learning of visual Representations and Momentum Contrast) on the ImageNet classification benchmark (Caron et al., 2020).

Fig. 2 shows the "swapped" prediction mechanism in SwAV. Given two image views (x_1 and x_2), we computed their code Q_1 and Q_2 by mapping their feature vectors (z_1 and z_2) to a set of prototypes C comprising K prototype vectors, as detailed in Section 3.2. Then, the "swapped" prediction problem is solved using the following loss function:

$$L(z_1, z_2) = l(z_1, Q_2) + l(z_2, Q_1) \tag{6}$$

where $l(z, Q)$ measures the fit between the feature z and the code Q. It can be computed as follows:

$$l(z_1, Q_2) = -\sum_k Q_2^{(k)} \log p_1^{(k)} \tag{7}$$

$$l(z_2, Q_1) = -\sum_k Q_1^{(k)} \log p_2^{(k)} \tag{8}$$

$$p_1^{(k)} = \frac{\exp\left(\frac{1}{\tau}\mathbf{z}_1^\top \mathbf{c}_k\right)}{\sum_{k'} \exp\left(\frac{1}{\tau}\mathbf{z}_1^\top \mathbf{c}_{k'}\right)} \tag{9}$$

$$p_2^{(k)} = \frac{\exp\left(\frac{1}{\tau}\mathbf{z}_2^\top \mathbf{c}_k\right)}{\sum_{k'} \exp\left(\frac{1}{\tau}\mathbf{z}_2^\top \mathbf{c}_{k'}\right)} \tag{10}$$

where C_k is the kth prototype vector in C, and τ denotes the temperature parameter that controls the sharpness of the probability distribution (Caron et al., 2020).

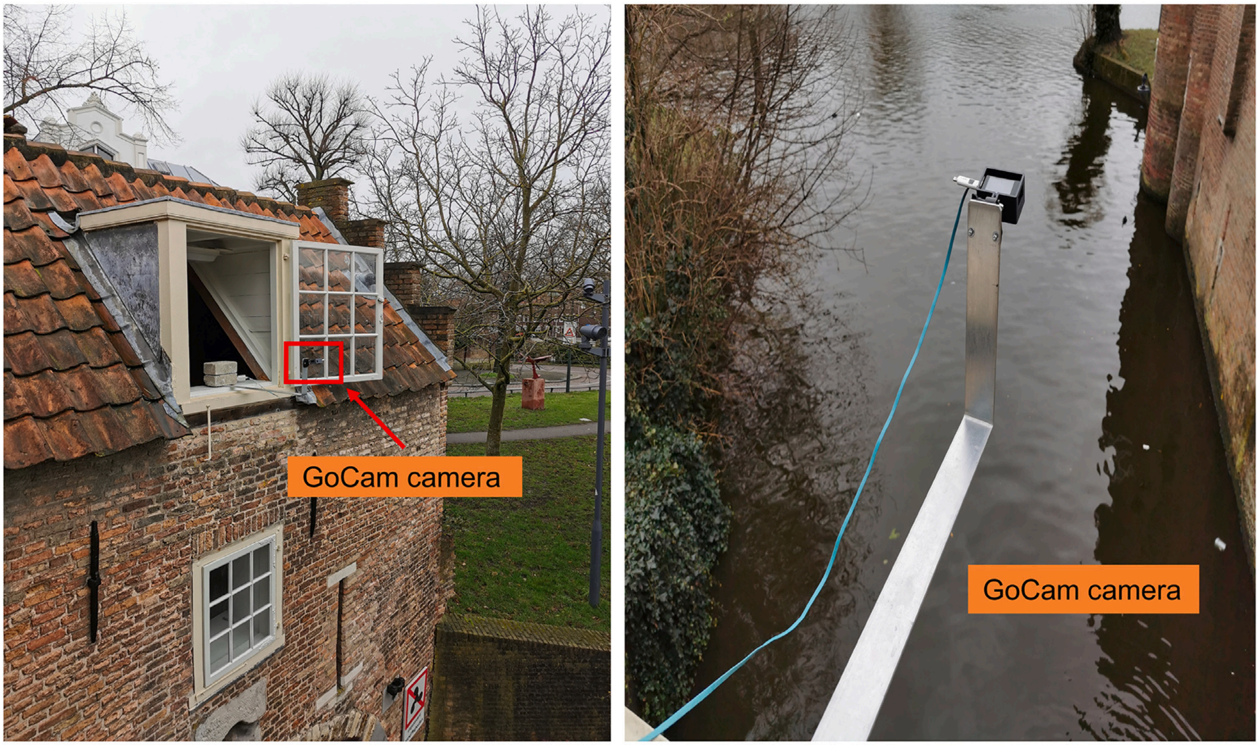

Fig. A.1. Monitoring setups at the Oostpoort.

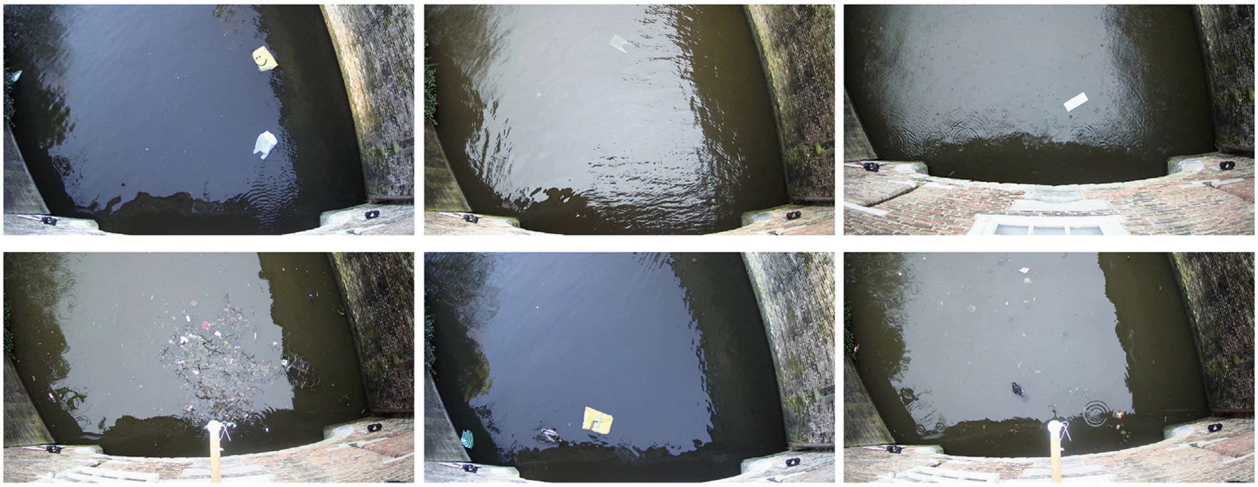

Fig. A.2. Examples of Oostpoort images.

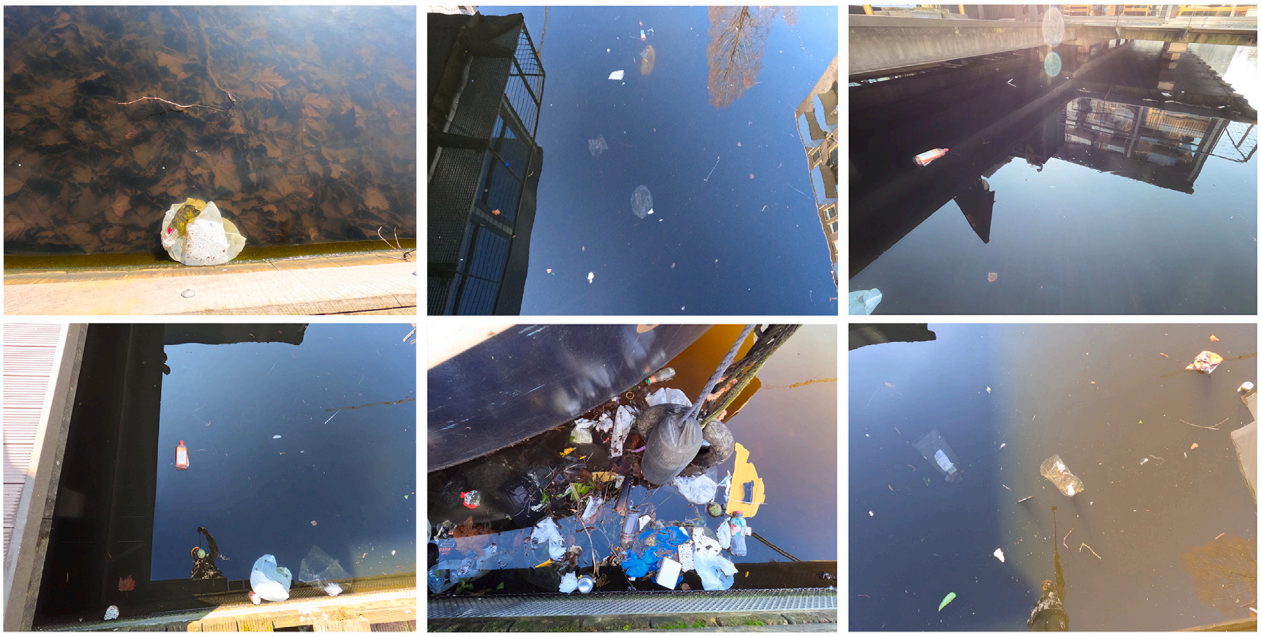

Fig. A.3. Examples of Amsterdam images.

B.2. Faster R-CNN

The Faster R-CNN is a two-stage detection network. In the first stage of the Faster R-CNN, the backbone extracts relevant feature maps from the input data. Then, the region proposal network, a fully convolutional network, generates region proposals from the shared feature maps. These region proposals together with the feature maps are fed into the RoI pooling layer, performs the pooling operation to integrate feature maps of region proposals with different scales into fixed size feature maps. In the second stage, the extra-network predicts the category with a confidence level and the precise location of objects from each region proposal in the fixed size feature maps.

B.3. Bounding box refinement with Non-Maximum Suppression

Non-Maximum Suppression (NMS) is a post-processing technique often applied after object detection to eliminate redundant bounding boxes, and ensure that each detected object is represented by the single most probable box (Hosang et al., 2017). It compares the overlap of boxes using Intersection over Union (IoU) and suppresses all boxes except the one with the highest confidence score when the overlap exceeds a specific threshold. The IoU NMS threshold of 0.5 is a common value that balances the need to reduce box overlap against the risk of missing closely spaced objects.

B.4. Evaluation metrics

The average precision (AP) is calculated by integrating the area under the precision–recall curve. For object detection, the precision–recall curve is computed by (i) sorting all detections in descending order based on their confidence level, (ii) accumulating all TPs and FPs, (iii) and computing p and r for each cumulative detection

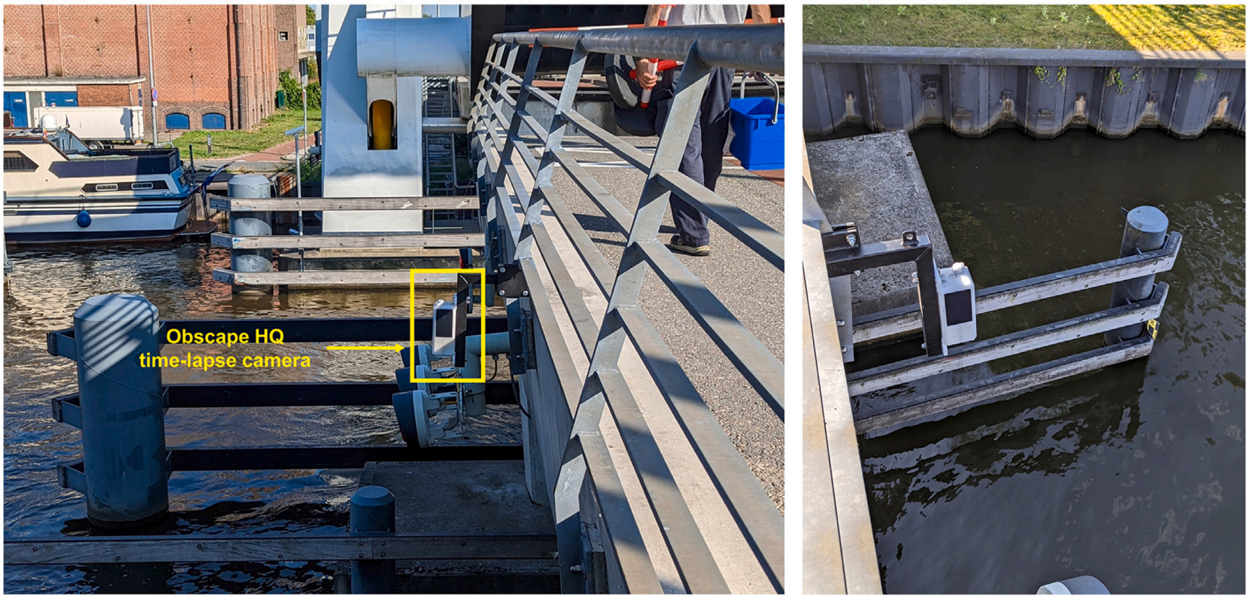

Fig. A.4. Monitoring setups at Groningen.

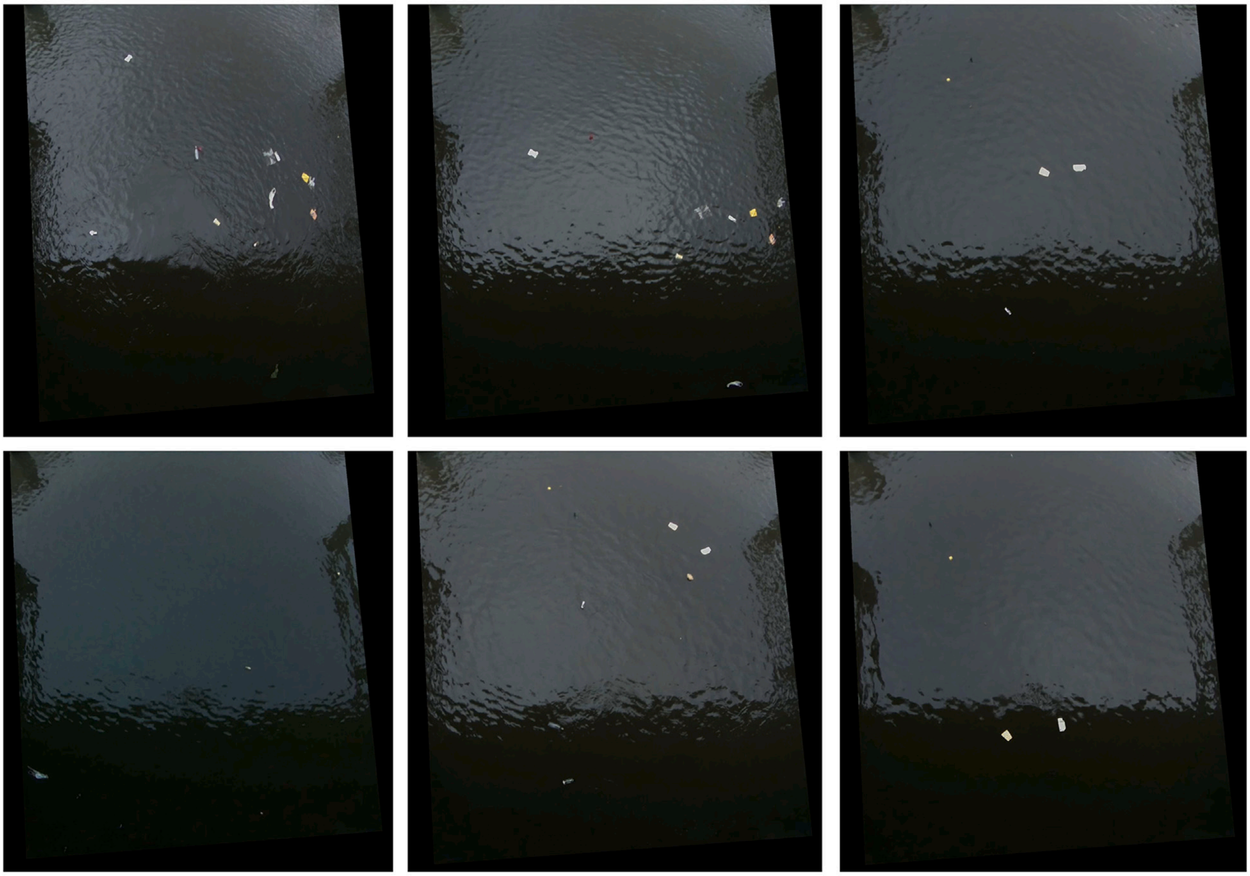

Fig. A.5. Examples of Groningen images. The images used in the experiments are cropped to omit the structure.

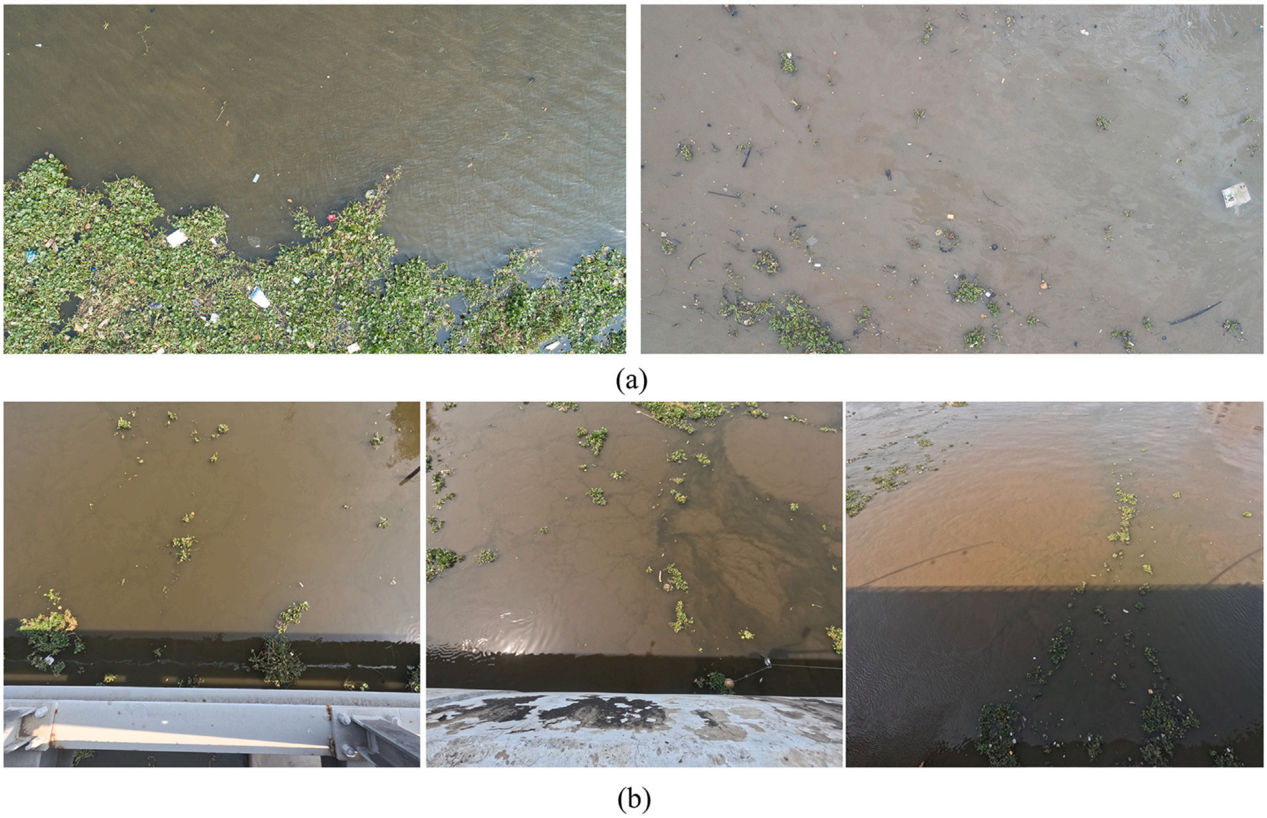

Fig. A.6. Examples of Ho Chi Minh City images collected by (a) drones and (b) cameras.

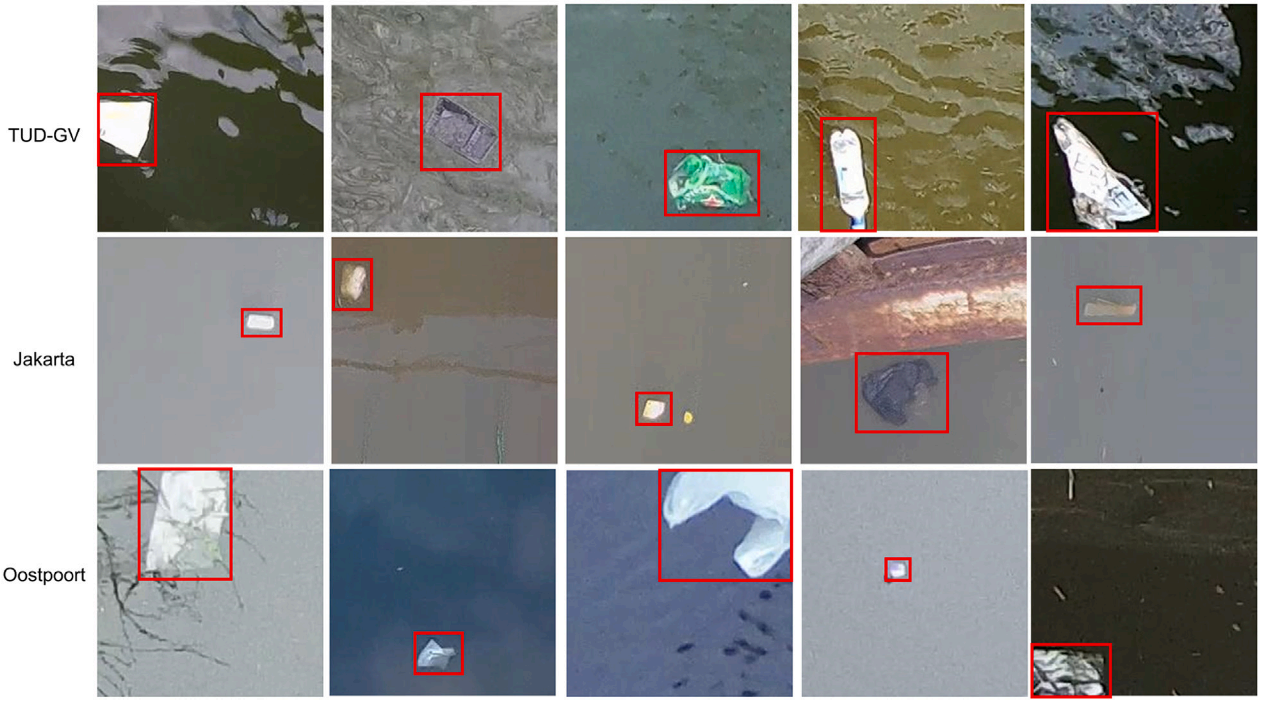

Fig. A.7. Examples of images tiles (224 × 224 pixels) from TUD-GV, Jakarta and Oostpoort dataset.

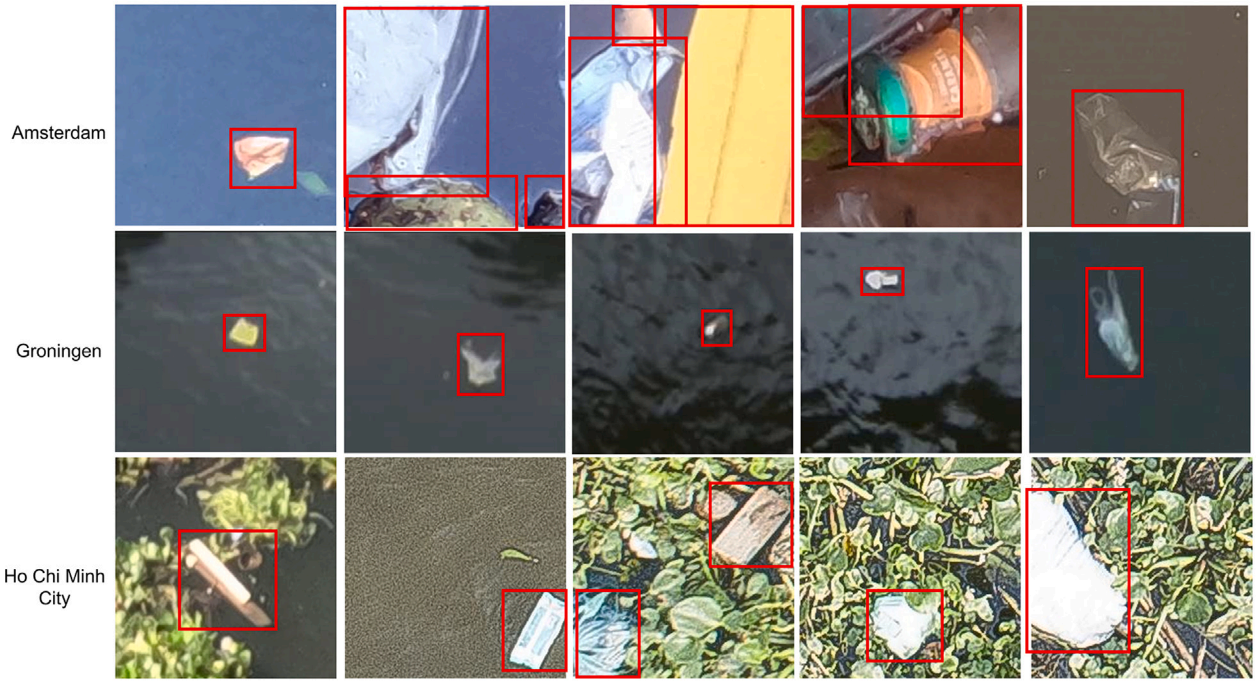

Fig. A.8. Examples of images tiles (224 × 224 pixels) from Amsterdam, Groningen and Ho Chi Minh City dataset.

Table C.1
Model performances of the Faster R-CNN with ResNet50 backbone using various methods on Amsterdam images. The model was fine-tuned on the Train100% dataset.

Method	Images with litter annotated						Images without litter
	TP	FN	FP	Precision	Recall	F1-score	FP
SwAV-FTAL-F2	81	123	**90**	**0.47**	0.40	0.43	**1530**
SwAV-Scratch-F2	83	121	109	0.43	0.41	0.42	1839
Baseline-F2	114	90	160	0.42	0.56	**0.48**	2617
SwAV-FTAL-F4	**138**	**66**	241	0.36	**0.68**	0.47	2249
SwAV-Scratch-F4	104	100	129	0.45	0.51	**0.48**	1695
Baseline-F4	95	109	180	0.35	0.47	0.40	2115

Table C.2
Model performances of the Faster R-CNN with ResNet50 backbone using various methods on Groningen images. The model was fine-tuned on the Train100% dataset.

Method	Images with litter annotated						Images without litter
	TP	FN	FP	Precision	Recall	F1-score	FP
SwAV-FTAL-F2	143	382	53	0.73	0.27	0.40	430
SwAV-Scratch-F2	117	408	56	0.68	0.22	0.34	401
Baseline-F2	165	360	**28**	**0.85**	0.31	0.46	**67**
SwAV-FTAL-F4	**283**	**242**	137	0.67	**0.54**	**0.60**	151
SwAV-Scratch-F4	227	298	99	0.70	0.43	0.53	219
Baseline-F4	208	317	167	0.55	0.40	0.46	468

(Dollár and Lin, 2014; Padilla et al., 2020). In the computation of r for the accumulated detections, the denominator term is constant and equal to the total amount of ground-truth boxes.

AP is an average measure that can sometimes obscure model weaknesses, e.g., a model might achieve good AP through a few highly accurate detections but perform poorly on others. The computation method for the precision–recall curve can also introduce challenges since the precision at each recall level can be subject to fluctuations due to the model's varying confidence levels across different detections (Padilla et al., 2020).

The F1-score captures a model's accuracy in detecting objects (recall) while minimizing incorrect detections (precision), making it crucial for contexts where false positives and false negatives have significant implications. Thus, combining AP50 and F1-score allows for a more thorough assessment of both localization accuracy and overall detection efficacy.

Appendix C. Confusion matrices and performance metrics for out-of-domain generalization

See Tables C.1–C.3.

Table C.3
Model performances of the Faster R-CNN with ResNet50 backbone using various methods on Ho Chi Minh City images. The model was fine-tuned on the Train100% dataset.

Method	Images with litter annotated						Images without litter
	TP	FN	FP	Precision	Recall	F1-score	FP
SwAV-FTAL-F2	**340**	**751**	1128	0.23	**0.31**	0.27	5889
SwAV-Scratch-F2	268	823	613	0.30	0.25	0.27	5291
Baseline-F2	254	837	1436	0.15	0.23	0.18	7326
SwAV-FTAL-F4	310	781	954	0.25	0.28	0.26	4009
SwAV-Scratch-F4	272	819	**434**	**0.39**	0.25	**0.30**	**2946**
Baseline-F4	236	855	929	0.20	0.22	0.21	4300

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al., 2023. Gpt-4 technical report. arXiv preprint arXiv:2303.08774.

Bellou, N., Gambardella, C., Karantzalos, K., Monteiro, J.G., Canning-Clode, J., Kemna, S., Arrieta-Giron, C.A., Lemmen, C., 2021. Global assessment of innovative solutions to tackle marine litter. Nat. Sustain. 4 (6), 516–524.

Bolton, S., Dill, R., Grimaila, M.R., Hodson, D., 2023. ADS-B classification using multivariate long short-term memory–fully convolutional networks and data reduction techniques. J. Supercomput. 79 (2), 2281–2307.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al., 2020. Language models are few-shot learners. Adv. Neural Inf. Process. Syst. 33, 1877–1901.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A., 2020. Unsupervised learning of visual features by contrasting cluster assignments. Adv. Neural Inf. Process. Syst. 33, 9912–9924.

Chen, T., Kornblith, S., Norouzi, M., Hinton, G., 2020. A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning. PMLR, pp. 1597–1607.

Cuturi, M., 2013. Sinkhorn distances: Lightspeed computation of optimal transport. Adv. Neural Inf. Process. Syst. 26.

Delft High Performance Computing Centre (DHPC), 2022. DelftBlue supercomputer (Phase 1). https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase1.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. Ieee, pp. 248–255.

Dollár, P., Lin, T.-Y., 2014. Detectron2. https://github.com/cocodataset/cocoapi.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

van Emmerik, T.H., Frings, R.M., Schreyers, L.J., Hauk, R., de Lange, S.I., Mellink, Y.A., 2023. River plastic transport and deposition amplified by extreme flood. Nat. Water 1–9.

van Emmerik, T., Mellink, Y., Hauk, R., Waldschläger, K., Schreyers, L., 2022. Rivers as plastic reservoirs. Front. Water 3, 212.

Goyal, P., Duval, Q., Reizenstein, J., Leavitt, M., Xu, M., Lefaudeux, B., Singh, M., Reis, V., Caron, M., Bojanowski, P., Joulin, A., Misra, I., 2021. VISSL. https://github.com/facebookresearch/vissl.

Goyal, P., Duval, Q., Seessel, I., Caron, M., Misra, I., Sagun, L., Joulin, A., Bojanowski, P., 2022. Vision models are more robust and fair when pretrained on uncurated images without supervision. arXiv preprint arXiv:2202.08360.

Güldenring, R., Nalpantidis, L., 2021. Self-supervised contrastive learning on agricultural images. Comput. Electron. Agric. 191, 106510.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778.

Hosang, J., Benenson, R., Schiele, B., 2017. Learning non-maximum suppression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4507–4515.

Huix, J.P., Ganeshan, A.R., Haslum, J.F., Söderberg, M., Matsoukas, C., Smith, K., 2024. Are natural domain foundation models useful for medical image classification? In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 7634–7643.

Hurley, R., Braaten, H.F.V., Nizzetto, L., Steindal, E.H., Lin, Y., Clayer, F., van Emmerik, T., Buenaventura, N.T., Eidsvoll, D.P., Økelsrud, A., et al., 2023. Measuring riverine macroplastic: Methods, harmonisation, and quality control. Water Res. 119902.

Jaiswal, A., Babu, A.R., Zadeh, M.Z., Banerjee, D., Makedon, F., 2020. A survey on contrastive self-supervised learning. Technologies 9 (1), 2.

Jakubik, J., Roy, S., Phillips, C.E., Fraccaro, P., Godwin, D., Zadrozny, B., Szwarcman, D., Gomes, C., Nyirjesy, G., Edwards, B., Kimura, D., Simumba, N., Chu, L., Mukkavilli, S.K., Lambhate, D., Das, K., Bangalore, R., Oliveira, D., Muszynski, M., Ankur, K., Ramasubramanian, M., Gurung, I., Khallaghi, S., Li, H.S., Cecil, M., Ahmadi, M., Kordi, F., Alemohammad, H., Maskey, M., Ganti, R., Weldemariam, K., Ramachandran, R., 2023. Foundation models for generalist geospatial artificial intelligence. Preprint Available on arxiv:2310.18660.

Jia, T., Kapelan, Z., de Vries, R., Vriend, P., Peereboom, E.C., Okkerman, I., Taormina, R., 2023a. Deep learning for detecting macroplastic litter in water bodies: a review. Water Res. 119632.

Jia, T., Peng, Z., Yu, J., Piaggio, A.L., Zhang, S., de Kreuk, M.K., 2024. Detecting the interaction between microparticles and biomass in biological wastewater treatment process with Deep Learning method. Sci. Total Environ. (ISSN: 0048-9697) 175813.

Jia, T., Vallendar, A.J., de Vries, R., Kapelan, Z., Taormina, R., 2023b. Advancing deep learning-based detection of floating litter using a novel open dataset. Front. Water 5, 1298465.

Kaandorp, M.L., Lobelle, D., Kehl, C., Dijkstra, H.A., van Sebille, E., 2023. Global mass of buoyant marine plastics dominated by large long-lived debris. Nat. Geosci. 16 (8), 689–694.

Lebreton, L., Slat, B., Ferrari, F., Sainte-Rose, B., Aitken, J., Marthouse, R., Hajbane, S., Cunsolo, S., Schwarz, A., Levivier, A., et al., 2018. Evidence that the Great Pacific Garbage Patch is rapidly accumulating plastic. Sci. Rep. 8 (1), 1–15.

Li, W., Lee, H., Wang, S., Hsu, C.-Y., Arundel, S.T., 2023. Assessment of a new GeoAI foundation model for flood inundation mapping. In: Proceedings of the 6th ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery. pp. 102–109.

van Lieshout, C., van Oeveren, K., van Emmerik, T., Postma, E., 2020. Automated river plastic monitoring using deep learning and cameras. Earth Space Sci. 7 (8), e2019EA000960.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., 2014. Microsoft coco: Common objects in context. In: European Conference on Computer Vision. Springer, pp. 740–755.

Liu, X., Zhang, F., Hou, Z., Mian, L., Wang, Z., Zhang, J., Tang, J., 2021. Self-supervised learning: Generative or contrastive. IEEE Trans. Knowl. Data Eng. 35 (1), 857–876.

Loshchilov, I., Hutter, F., 2016. Sgdr: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983.

Manjarrés, Á., Fernández-Aller, C., López-Sánchez, M., Rodríguez-Aguilar, J.A., Castañer, M.S., 2021. Artificial intelligence for a fair, just, and equitable world. IEEE Technol. Soc. Mag. 40 (1), 19–24.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al., 2023. Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193.

Padilla, R., Netto, S.L., Da Silva, E.A., 2020. A survey on performance metrics for object-detection algorithms. In: 2020 International Conference on Systems, Signals and Image Processing. IWSSIP, IEEE, pp. 237–242.

Pham, H.H., Le, T.T., Tran, D.Q., Ngo, D.T., Nguyen, H.Q., 2021. Interpreting chest X-rays via CNNs that exploit hierarchical disease dependencies and uncertainty labels. Neurocomputing 437, 186–194.

Punjani, A., Fleet, D.J., 2021. 3D variability analysis: Resolving continuous flexibility and discrete heterogeneity from single particle cryo-EM. J. Struct. Biol. 213 (2), 107702.

Reddy, Y., Viswanath, P., Reddy, B.E., 2018. Semi-supervised learning: A brief review. Int. J. Eng. Technol. 7 (1.8), 81.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Adv. Neural Inf. Process. Syst. 28, 91–99.

Renfei, C., Jian, W., Yong, P., Zhongwen, L., Hua, S., 2023. Detection and tracking of floating objects based on spatial-temporal information fusion. Expert Syst. Appl. 225, 120185.

Wolf, M., van den Berg, K., Garaba, S.P., Gnann, N., Sattler, K., Stahl, F., Zielinski, O., 2020. Machine learning for aquatic plastic litter detection, classification and quantification (APLASTIC-Q). Environ. Res. Lett. 15 (11), 114042.

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., Girshick, R., 2019. Detectron2. https://github.com/facebookresearch/detectron2.

Wu, Y., Ma, X., Guo, G., Jia, T., Huang, Y., Liu, S., Fan, J., Wu, X., 2024. Advancing deep learning-based acoustic leak detection methods towards application for water distribution systems from a data-centric perspective. Water Res. 121999.

Xu, Y., Ou, Q., van der Hoek, J.P., Liu, G., Lompe, K.M., 2024. Photo-oxidation of micro-and nanoplastics: physical, chemical, and biological effects in environments. Environmental Science & Technology 58 (2), 991–1009.

Yosinski, J., Clune, J., Bengio, Y., Lipson, H., 2014. How transferable are features in deep neural networks? Adv. Neural Inf. Process. Syst. 27.