Detecting the interaction between microparticles and biomass in biological wastewater treatment process with Deep Learning method

Jia, Tianlong; Peng, Zhaoxu; Yu, Jing; Piaggio, Antonella L.; Zhang, Shuo; de Kreuk, Merle K.

DOI
10.1016/j.scitotenv.2024.175813

Publication date
2024

Document Version
Final published version

Published in
Science of the Total Environment

Science of the Total Environment 951 (2024) 175813

Contents lists available at ScienceDirect

Science of the Total Environment

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

Detecting the interaction between microparticles and biomass in biological wastewater treatment process with Deep Learning method

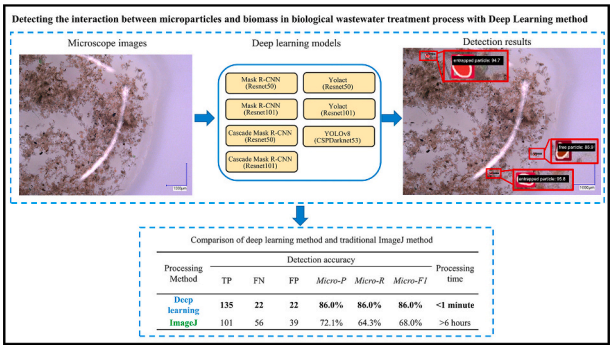
Tianlong Jia [a], Zhaoxu Peng [a,b,*], Jing Yu [c], Antonella L. Piaggio [a], Shuo Zhang [a], Merle K. de Kreuk [a]

[a] Delft University of Technology, Faculty of Civil Engineering and Geosciences, Department of Water Management, Stevinweg 1, 2628 CN Delft, the Netherlands
[b] Zhengzhou University, School of Water Conservancy and Transportation, Kexue Road 100, Zhengzhou 450001, China
[c] Erasmus University Medical Center, Department of Radiology and Nuclear Medicine, Dr Molewaterplein 40, 3015 GD Rotterdam, the Netherlands

HIGHLIGHTS

- A Deep Learning (DL) method to detect micro particles in biomass mixture is proposed.
- A new microscopic image dataset for the detection using DL is proposed.
- Detection performance of seven DL architectures is thoroughly evaluated.
- The DL method shows better detection performance compared to ImageJ method.

GRAPHICAL ABSTRACT



ARTICLE INFO

Editor: Yifeng Zhang

Keywords:
Artificial intelligence
Computer vision
Instance segmentation
Microplastics
Mask R-CNN
Yolact

ABSTRACT

Investigating the interaction between influent particles and biomass is basic and important for the biological wastewater treatment. The micro-level methods allow for this, such as the microscope image analysis method with the conventional ImageJ processing software. However, these methods are cost and time-consuming, and require a large amount of work on manual parameter tuning. To deal with this problem, we proposed a deep learning (DL) method to automatically detect and quantify microparticles free from biomass and entrapped in biomass from microscope images. Firstly, we introduced a "TU Delft-Interaction between Particles and Biomass" dataset containing labeled microscope images. Then, we built DL models using this dataset with seven state-of-the-art model architectures for a instance segmentation task, such as Mask R-CNN, Cascade Mask R-CNN, Yolact and YOLOv8. The results show that the Cascade Mask R-CNN with ResNet50 backbone achieves promising detection accuracy, with a $mAP50_{box}$ and $mAP50_{mask}$ of 90.6 % on the test set. Then, we benchmarked our results against the conventional ImageJ processing method. The results show that the DL method significantly outperforms the ImageJ processing method in terms of detection accuracy and processing cost. The DL method shows a 13.8 % improvement in micro-average precision, and a 21.7 % improvement in micro-average recall, compared to the ImageJ method. Moreover, the DL method can process 70 images within 1 min, while the

* Corresponding author at: Delft University of Technology, Faculty of Civil Engineering and Geosciences, Department of Water Management, Stevinweg 1, 2628 CN Delft, the Netherlands.
E-mail address: pzx@zzu.edu.cn (Z. Peng).

https://doi.org/10.1016/j.scitotenv.2024.175813
Received 12 May 2024; Received in revised form 19 August 2024; Accepted 24 August 2024
Available online 25 August 2024

ImageJ method costs at least 6 h. The promising performance of our method allows it to offer a potential alternative to examine the interaction between microparticles and biomass in biological wastewater treatment process in an affordable manner. This approach offers more useful insights into the treatment process, enabling further reveal the microparticles transfer in biological treatment systems.

1. Introduction

The pollution in water bodies is a worldwide issue, that affects human health and ecosystems (Lebreton et al., 2017; Borrelle et al., 2020; Jia et al., 2023a). Particulate matter commonly exist in aquatic environment, such as colloids and suspended solids (Bhat and Janaszek, 2024). Our daily life produce a large volume of wastewater, with the majority of pollutants existing in particulate form (Lan et al., 2022; Noyan et al., 2017). To remove these pollutants from sewage, wastewater-treatment plants (WWTPs) typically remove the bigger particles (millimeter range) by physical treatment units, such as coarse screens, fine screens, and sedimentation tanks. Then, they usually employ biological treatment process to remove many smaller particles (micrometer range) due to the easy operation and excellent effluent quality (Mesquita et al., 2013; Yang et al., 2024), which are based on activated sludge, biofilm or granular sludge (Zhang et al., 2024; Zhen et al., 2024). Finally, physicochemical technologies were employed to purify the wastewater further (Xu et al., 2024; Shen et al., 2023). Among the above procedures, the biological process is the most important which is designed to reduce biodegradable substrates (in solution or suspension) from sewage (Inbar et al., 2023). The dissolved substances in solution can be utilized by biomass through diffusion, whereas suspended substances need to be adsorbed by the biomass firstly, then decomposed into soluble substances. However, desorption may occur when the adhesion was weak, affecting microbial growth and deteriorating treatment performance. The factors affecting the interaction between biomass and particles include agitator power, aeration intensity, particle size, and biomass size (Sven et al., 2017; Mohamad et al., 2018). Therefore, monitoring the particle transfer process in biological treatment system and subsequently optimizing operations (e.g., mixing and aeration) are crucial for enhancing the treatment performance.

To monitor micro objects in biological wastewater treatment process, microscope image analysis is common and effective method (Inbar et al., 2023). It can offer information on the properties of microparticles, that is necessary for efficient activated sludge treatment, risk evaluation, and effluent quality. Researchers usually use image analysis software such as ImageJ to measure the particle size and investigate the morphology of the particles in microscope images (Schindelin et al., 2012; Piaggio et al., 2022; Bhat, 2024a, 2024b). ImageJ provides a variety of image processing and analysis functions (such as filtering, segmentation, measurement, and statistical analysis), and is able to process various formats of images (e.g., TIFF, PNG, GIF, JPEG, BMP, DICOM and FITS). But this processing method is time-consuming and laborious, and often relies on domain-specific knowledge to fine-tune parameters (e.g., threshold and output size), which hinders monitoring a large number of samples (Grass et al., 2014; Bhat, 2024c).

Considering the limitations associated with the ImageJ method, the development of a new automatic and efficient particle detection method for microscope image analysis is needed (Tran et al., 2023). Recently, Deep Learning (DL) approaches have garnered considerable research attention, as they provide automated alternatives to conventional methods (Zhao et al., 2020). Some studies have shown the significance of DL methods, particularly Convolutional Neural Networks (CNNs) for wastewater treatment, management, and monitoring, including works on various computer vision (CV) tasks, e.g., image classification and object detection. For example, Yurtsever and Yurtsever (2019) used the GoogLeNet model to recognize and classify micro beads in urban wastewater into five categories based on microscope images, achieving the classification accuracy of 89 %. Inbar et al. (2023) proposed a DL method to analyze and monitor the activated sludge process. They employed YOLOv5 and Faster R-CNN to detect filaments, protozoa, spherical flocs and open flocs from microscope images of flocs and microorganisms, with the mean average precision (mAP) of 67 % and 52 %, respectively. Satoh et al. (2021) used the InceptionV3 model to monitor the morphology of sludge flocs from microscope images. The model can identify and classify aggregated and dispersed flocs with a training accuracy of about 95 %, and successfully detect the 20 % morphological change in the aggregated flocs. However, most of these studies focus on detecting microorganisms (e.g., filaments, bacteria, protozoa and sludge flocs). There is a lack of DL methods for influent particle detection. Although some other methods are effective to investigate the particles transfer process between wastewater and activated sludge, such as DNA tracer encapsulated in silica particles, while the cost is super high (Grass et al., 2014).

To bridge this gap, we proposed a DL method to detect and quantify microparticles free from biomass and entrapped in biomass. To the best of our knowledge, this work is the first to propose and assess DL-based methods to examine the interaction between microparticles and biomass in biological wastewater treatment process. This paper is structured as follows. Section 2 presents the materials and methodology used in this work. Section 3 introduces the experiments. Section 4 presents and discusses the experimental results and the limitations of this study. In Section 5, we summarized the conclusions.

2. Materials and methodology

2.1. The TUD-IPB dataset

We created the "TU Delft-Interaction between Particles and Biomass" (TUD-IPB) dataset from the experiments conducted in water lab of TU Delft (Stevinweg 1, Delft, Netherlands). Fig. 1 shows the scheme of the experiments. We collected the biomass mixture from a sewage treatment plant located in the city of Utrecht, the Netherlands. The Royal Haskoning DHV designed this plant with a treatment ability of 74,700 $m^3 \cdot d^{-1}$ and a sludge loading rate of 0.05 $kgCOD \cdot kgVSS^{-1} \cdot d^{-1}$. The biomass mixture was sampled with a specified down inflow and top outflow cylinder from the bottom of Nereda reactor after 30 min of aeration. The mixture was rinsed with distilled water three times, followed by sequential sieving through the mesh size of 3.1 mm, 2.0 mm, 1.0 mm and 0.2 mm, respectively. The biomass with the diameter of >3.1 mm (A), 2.0–3.1 mm (B), 1.0–2.0 mm (C), 0.2–1.0 mm (D) and <0.2 mm (E) were primarily collected for batch test (Fig. 1(1)). The fluorescent micro beads (Cospheric, WTW D-82362 Weilheim, Model: SEP 25. Order NO: 209503) were used to simulate particles of sewage. The diameter and density of the micro beads were 250 μm and 1.02 $g \cdot L^{-1}$, respectively. The new micro beads were grind into crushed particles (82.58 ± 47.95 μm) by a mortar, followed by the dilution of distilled water (2000–2500 $particles \cdot L^{-1}$) (Fig. 1(2)). The batch test was carried out using five conical flasks with the volume of 250 mL each. For each flask, 200 mL solution of fluorescent crushed particles was dropped in, followed by the addition of biomass. These five conical flasks were added by the biomass with varying diameter, respectively. Then the mixture was aerated with the intensity of 2 $L \cdot min^{-1}$ for 60 min (Fig. 1(3)). Samples (10 mL) were taken at 10 min, 30 min and 60 min from each flask and were observed by a digital microscope (VHX-5000). Finally, each sample was recorded by several microscope images with a resolution of 1600 × 1200 pixel to capture all the micro crushed particles (Fig. 1(4)). We annotated the micro crushed particles free from

biomass and entrapped in biomass with mask labels (Fig. 1(5)). Since this experiment focused on establishing an analytical method using collected images rather than exploring the interaction between particles and biomass through the images, no repeat experiments were conducted.

This dataset is representative and captures variability across materials found in wastewater treatment systems. In this experiment, we utilized the biomass mixture comprising both granular sludge and common activated sludge flocs, representing the full range of size distribution for usually found microbial aggregate (De Kreuk et al., 2007). Besides, microbial aggregates in actual municipal wastewater treatment plants typically appear as dark brown with minimal color variation. Moreover, we used the commonly used fluorescent microbeads as tracers (Sorasan et al., 2022). This representative dataset significantly benefits to the development of DL models with great applicability and generalization capability to various real-world biological wastewater treatment systems.

2.2. Deep learning algorithms

To detect free particles and entrapped particles and qualify the area of them, we conducted the multi-class instance segmentation task. This task is to assign category labels and instance identities to pixels in images of each particle, and qualify the number of pixels belonging to each particle for further calculating the area of each particle. Thus, it can simultaneously tackle with the problem of object detection and semantic segmentation (Jia et al., 2023a). In this work, we selected four state-of-the-art instance segmentation algorithms, including (1) Mask Region-based Convolutional Neural Network (Mask R-CNN) (He et al., 2017), (2) Cascade Mask R-CNN (Cai and Vasconcelos, 2019), (3) You Only Look At CoefficienTs (Yolact) (Bolya et al., 2019) and (4) You Only Look Once v8 (YOLOv8) (Jocher et al., 2023).

2.2.1. Mask R-CNN

Fig. 2 shows the detailed architecture of the Mask R-CNN with the ResNet backbone (He et al., 2016). The Mask R-CNN is a two-stage network, which extends the Faster R-CNN (Ren et al., 2015) by adding a branch for predicting object segmentation masks alongside the existing branch for bounding box prediction. In the first stage, the ResNet backbone extracts feature maps from the input data. Then, it uses a Region Proposal Network (RPN) to produce region proposals from the shared feature maps. These proposals and the feature maps are fed into the RoIAlign layer to extract a fixed size feature map of each RoI (Region of Interest). In the second stage, the mask head (i.e., convolutional layers) predicts pixel-wise masks in each RoI from fixed size feature maps. On the other hand, the detection head (i.e., an original part of the Faster R-CNN) uses fixed size feature maps for the bounding box and classification regression tasks.

2.2.2. Cascade Mask R-CNN

Fig. 3 shows the architecture of the Cascade Mask R-CNN with the ResNet backbone. The Cascade Mask R-CNN is a multi-stage extension of Mask R-CNN. The original Mask R-CNN comprises only a single stage to predict masks and bounding box in each RoI, i.e., the second stage of Mask R-CNN (Fig. 2). Compared with the Mask R-CNN, the Cascade Mask R-CNN comprises multiple stages, where the output of each stage is input to the next stage to obtain better refinement predictions. First, the ResNet backbone extracts feature maps, and then input them to three pooling layers. At the same time, region proposals produced by the RPN are fed into the first pooling layer (the top pooling layer). Second, the first network head (the top network head in Fig. 3) performs the first round of classification and bounding box regression. Then, these bounding boxes are considered as the input for the subsequent network head (the middle network head in Fig. 3). Similarly, the bounding boxes produced by the second network head are input to the last pooling layer (the bottom pooling layer). Finally, the last network head predicts the category, location of objects and pixel-wise masks from the output of the last pooling layer. In addition, the training data of each stage is sampled with increasing Intersection Over Union (IoU) thresholds, which addresses different training distributions (Chen et al., 2019).

In object detection and image segmentation tasks, the IoU is used to measure the deviation between ground truth and predicted area, which is calculated as follows:

$$IoU = \frac{Area(GT \cap Pred)}{Area(GT \cup Pred)} \tag{1}$$

where GT and Pred are the ground-truth and predicted bounding box or mask, respectively; A high IoU means that the predicted area does not deviate much from the ground truth bounding box or mask.

This threshold is usually set as 0.5 (Jia et al., 2023a). However, using only a single IoU threshold for training usually causes a problem. On the one hand, a low IoU threshold (e.g., 0.5) typically creates noisy detection in positive samples. On the other hand, a high threshold (e.g., 0.7) normally results in model overfitting due to vanishing positive samples

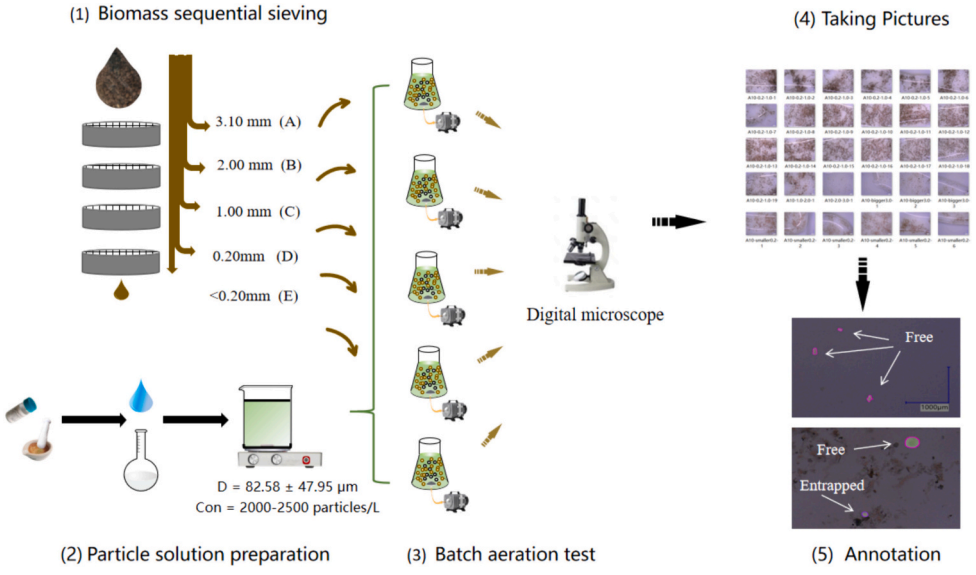


Fig. 1. Experimental scheme.

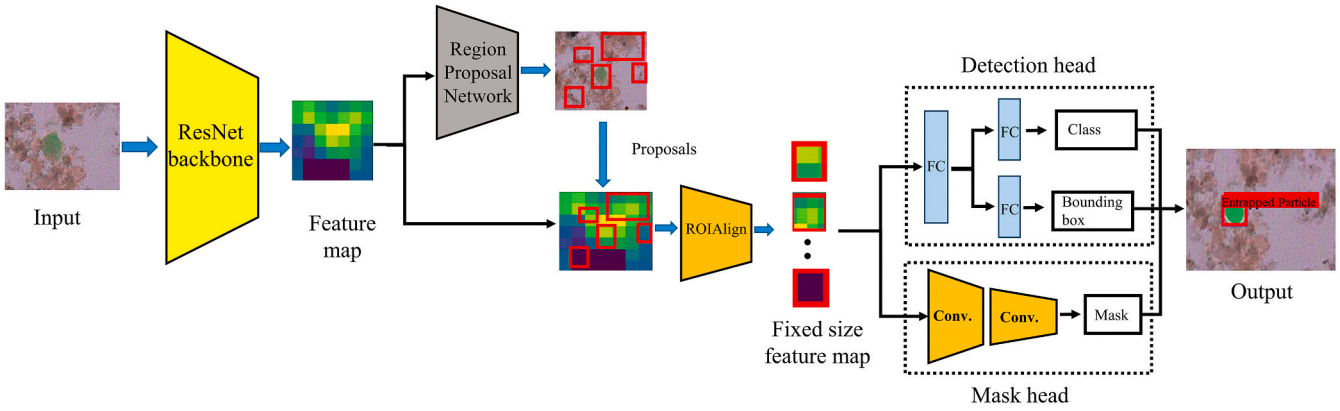


Fig. 2. Outline of the Mask R-CNN with ResNet backbone architecture for instance segmentation (He et al., 2017). First, the ResNet backbone derives feature maps from the input. Then, the region proposal network creates region proposals from these feature maps. Furthermore, these proposals and the feature maps are fed into the RoIAlign layer, that extract fixed size feature maps of each proposal. Finally, the mask head predicts pixel-wise masks, and the detection head simultaneously predicts the category and precise location of objects from the fixed size feature maps. Acronyms used: Convolutional layer (Conv.), Fully connected layer (FC).

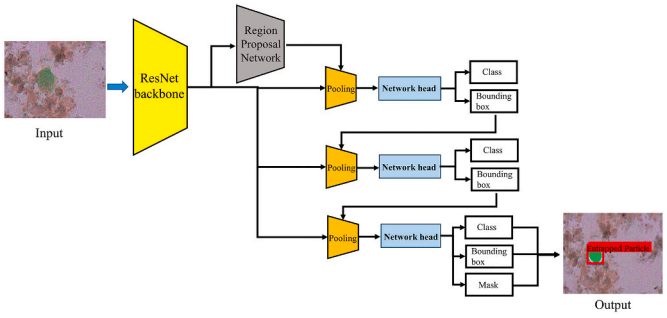


Fig. 3. Outline of the Cascade Mask R-CNN with ResNet backbone architecture for instance segmentation (Cai and Vasconcelos, 2019). First, the ResNet backbone extracts feature maps from the input, and then input these feature maps to three pooling layers. At the same time, the region proposal network generates region proposals from these feature maps. These proposals are fed into the first pooling layer (the top pooling layer). Second, the first network head predicts the category and location of objects with bounding box from the output of the pooling layer. Then, the predicted bounding boxes are input to the second pooling layer (the middle pooling layer) for the next round of classification and bounding box regression. Similarly, the predicted bounding boxes in this stage are input to the last pooling layer (the bottom pooling layer). Finally, the last network head predicts the category, location of objects and pixel-wise masks from the output of the last pooling layer.

for large thresholds. The Cascade Mask R-CNN addresses this problem by using multi-stage detectors trained with increasing IoU thresholds (Cai and Vasconcelos, 2019).

2.2.3. Yolact

Fig. 4 shows the architecture of the Yolact with the ResNet backbone.

The Yolact is a one-stage real-time instance segmentation algorithm, which divides the segmentation process into two parallel tasks (Bolya et al., 2019). Firstly, it produces a set of prototype masks of the entire input image using a protonet network. At the same time, it predicts the mask coefficient of each instance using a prediction head. Then, it linearly combines the mask coefficients with the prototype masks to obtain the final segmentation mask of each instance. Compared with two-stage instance segmentation algorithms (e.g., Mask R-CNN), the Yolact achieves faster processing speed by avoiding feature pooling and proposal generation, which are common used in two-stage algorithms (Cao et al., 2020). For example, Yolact enables to process images at an FPS of 30 on the COCO dataset (Lin et al., 2014) with the competitive accuracy (Bolya et al., 2019).

2.2.4. YOLOv8

YOLOv8 network was released in 2023, which belongs to the famous YOLO series networks (Jocher et al., 2023). These networks are commonly used in object detection and instance segmentation tasks due to the high detection accuracy and fast processing speed. YOLOv8 mainly includes the backbone and the head modules. The YOLOv8 uses the CSPDarknet53 as its backbone. The backbone, a convolutional neural network, is used to extract feature maps from the input data. Then, these feature maps are processed by the head module to predict object locations, object segmentation masks and object category. YOLOv8 adopts an anchor-free method for object detection. It involves predicting the center of objects before constructing the corresponding bounding boxes, rather than predicting offsets from predefined anchor boxes. This method enables YOLOv8 to achieve faster processing speed.

2.3. ImageJ processing method

Researchers have frequently used ImageJ processing software to

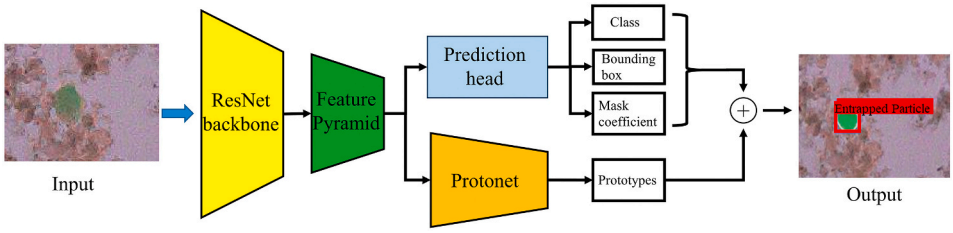


Fig. 4. Outline of the Yolact with ResNet backbone architecture for instance segmentation (Bolya et al., 2019). First, it uses a protonet network to produce prototype masks of the entire input image, and uses a prediction head to predict the mask coefficient of each instance. Then, it obtains the final segmentation mask of each instance by linearly combining the mask coefficients with the prototype masks.

measure and analyze the particle characteristics in the images (Schindelin et al., 2012; Piaggio et al., 2022; Bhat et al., 2024). The high definition images were usually taken by digital microscope. The recognition procedure usually consisted of 3 steps: (1) Converting the original image to 8-bit type; (2) Adjusting the threshold (e.g., minimum and maximum value) and output size for each image by subjective judgement; (3) Analyzing the particles (e.g., particle size, circularity, and display results). Every step is operated manually. After analyzing the images, this software outputs the area, perimeter, angle, circularity and other characteristics of each recognized particle. However, the adjusted threshold and output size can seriously influence the analyzed results. To improve the analysis accuracy, it is often necessary to try several times to achieve the optimal result. Sometimes, the original images need to be polished to reduce the impact on the output.

3. Experimental procedure

In this work, we aimed to demonstrate the potential of DL methods in comparison to the conventional ImageJ processing method for detection and quantification of microparticles free from biomass and entrapped in biomass in biological wastewater treatment process. To achieve this, we carried out two experiments. Experiment 1 involves evaluating seven state-of-the-art DL model architectures to identify the best performing model architecture on the TUD-IPB dataset (see Section 2.1). Experiment 2 involves benchmarking performance of the best performing model architecture against ImageJ processing method.

3.1. Experiment 1: DL architectures comparison

Table 1 shows the subsets used for Experiment 1. Firstly, we randomly selected 702 images from the TUD-IPB dataset described in Section 2.1. Then, we annotated the free particles (1081 items) and entrapped particles (616 items) in images with mask labels. Finally, we randomly divided these images into train, validation and test sets with the ratio of 80:10:10 (Table 1). The train and validation subsets are used to train the models, and evaluate their performance during the training process, respectively. The test subset is employed to assess the models' generalization capability on an "unseen" dataset. In this experiment, we compared the detection and segmentation performances of seven selected DL architectures on validation and test subsets: (1) Mask R-CNN with ResNet50 backbone, (2) Mask R-CNN with ResNet101 backbone, (3) Cascade Mask R-CNN with ResNet50 backbone, (4) Cascade Mask R-CNN with ResNet101 backbone, (5) Yolact with ResNet50 backbone, (6) Yolact with ResNet101 backbone, and (7) YOLOv8.

3.2. Experiment 2: DL versus ImageJ processing method

In this experiment, we compared the detection and segmentation accuracy and cost of the DL method and ImageJ processing method on test set in Table 1. We selected the best performing model architecture on test set emerging in Experiment 1.

3.3. Implementation of DL architectures

We firstly initialized the DL architectures with weights pre-trained on COCO dataset (Lin et al., 2014), and then fine-tuned it on the train set. We performed random flipping data augmentation during training (Jia et al., 2023b). We trained the models for 100 epochs using the SGD optimizer with a batch size of 4, a momentum of 0.9 and a weight decay of 0.0001 (Loshchilov and Hutter, 2016). Due to the limited computational resources available, we only fine-tuned learning rate hyperparameter in the set of {0.00001, 0.0001, 0.001, 0.01} for each DL architecture, since this parameter significantly affect the convergence rate of training scheme and model performance (Nakama, 2009). We only employed the best learning rate in Experiments 2. To avoid overfitting, we employed the model with parameters obtained from the epoch that produces the highest validation accuracy. Then, we assessed the model performance using the test set. The experiments are implemented on the Delft Blue supercomputer with NVIDIA Tesla V100S GPU (32 GB) (Delft High Performance Computing Centre (DHPC), 2022) using the Python programming language (version 3.8.16) and the PyTorch DL framework (version 1.13.1).

Table 1
Subsets for Experiment 1.

Subset	No. images	No. masks per class	
		Entrapped particle	Free particle
Train	562	478	927
Validation	70	67	68
Test	70	71	86

3.4. Performance evaluation metric

To evaluate the overall performances of DL models across all classes, we employed five metrics: (1) box-level mAP50 ($mAP50_{box}$), (2) mask-level mAP50 ($mAP50_{mask}$), (3) micro-average precision (*micro-P*), (4) micro-average recall (*micro-R*) and (5) micro-average F1 (*micro-F1*). The mAP50 is the mean Average Precision (mAP) at the IoU threshold of 50 % (Chian et al., 2021). The other metrics are also computed using the same threshold. The mAP is a comprehensive evaluation metric of all classes for a given IoU threshold, which is expressed as follows:

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{2}$$

where AP_i is the average precision (AP) of the i-th class for a given IoU threshold; n is the number of classes.

The AP value is the average of the precision P values at different recalls R, i.e., the area under the precision-recall curve (Padilla et al., 2020). These three values are defined as follows:

$$AP_i = \int_0^1 P_i(R_i)dR_i \tag{3}$$

$$P_i = \frac{TP_i}{TP_i + FP_i} \tag{4}$$

$$R_i = \frac{TP_i}{TP_i + FN_i} \tag{5}$$

where $P_i(R_i)$ is the precision at the recall of R_i; TP_i (True Positive) represents the number of detected bounding boxes or masks that belong to class i and have an IoU equal to or above the threshold; FP_i (False Positive) represent the number of detected bounding boxes or masks that either do not belong to class i or have an IoU below the threshold; FN_i (False Negative) means the number of ground truths belonging to class i that remain undetected.

The *micro-P*, *micro-R* and *micro-F1* metrics are computed as follows:

$$micro - P = \frac{\sum_i^n TP_i}{\sum_i^n TP_i + \sum_i^n FP_i} \tag{6}$$

$$micro - R = \frac{\sum_i^n TP_i}{\sum_i^n TP_i + \sum_i^n FN_i} \tag{7}$$

$$micro - F1 = \frac{2 \times micro - P \times micro - R}{micro - P + micro - R} \tag{8}$$

4. Results and discussion

4.1. Experiment 1: DL architectures comparison

Table 2 shows the training time and validation accuracy of seven model architectures with different learning rates on the instance segmentation task. Models with higher learning rates (i.e., 0.01) usually obtain the best validation accuracy across most models architectures, such as YOLOv8, Mask R-CNN and Cascade Mask R-CNN regardless of the backbone used. But for those model architectures, a too low learning rate (i.e., 0.00001) generally leads to poor validation accuracy. For Yolact architectures, a learning rate of 0.0001 (ResNet50 backbone) and 0.001 (ResNet101 backbone) yield the best accuracy among the tested learning rates. A higher learning rate (i.e., 0.01) may cause the deviation from the optimal value due to large updates to the model weights, resulting in the difficulty of model convergence. A lower learning rate (i.e., 0.00001) may cause the learning process to take a long time to converge or become stuck at local minimum (Dutta et al., 2018). Due to the fast processing speed of one-stage algorithms, Yolact and YOLOv8 takes less training time than two-stage algorithms (i.e., Mask R-CNN and Cascade Mask R-CNN). Especially, the training speed of YOLOv8 is 30 min per 100 epochs, which is 7–10 times faster than Mask R-CNN (198–270 min) and Cascade Mask R-CNN (240–306 min). However, YOLOv8 performs the worst in validation accuracy, with a $mAP50_{box}$ ranging from 48.7 % to 70.7 % and a $mAP50_{mask}$ between 41.9 % and 51.3 % with the learning rates ranging from 0.01 to 0.0001. Other model architectures achieve higher accuracy compared to YOLOv8, obtaining a $mAP50_{box}$ ranging from 44.6 % to 87.0 % and a $mAP50_{mask}$ between 47.4 % and 87.9 % under the same learning rate range.

In this experiment, we also compared the performance of different model architectures on unseen images from the test subset. Table 3 shows the performance of different model architectures on the test subset. In this table, we only reported the learning rate corresponding to the highest $mAP50_{mask}$ achieved on the validation set for every architecture. The results shows that both the Cascade Mask R-CNN and Mask R-CNN with ResNet50 backbones achieve the best performance, with $mAP50_{box}$ of 90.6 % and 89.2 %, and $mAP50_{mask}$ of 90.6 % and 90.3 % on the test subset, respectively. The Yolact performs worse, achieving a $mAP50_{box}$ varying between 78.2 % and 80.3 % and $mAP50_{mask}$ varying between 85.3 % and 85.9 % depending on the backbone. For example, with the same ResNet50 backbone, Cascade Mask R-CNN yields a significant improvement of 12.4 % in $mAP50_{box}$ and 4.7 % in $mAP50_{mask}$, compared to Yolact. One main reason for the poor performance of Yolact is that it generates insufficient prototype masks of the input images. The YOLOv8 performs the worst, with a $mAP50_{box}$ of 74.6 % and a $mAP50_{mask}$ of 56.4 %. It is not surprising since most one-stage model architectures usually perform worse than two-stage architectures in accuracy (Jia et al., 2023a).

While the Cascade Mask R-CNN and Mask R-CNN with ResNet50 backbone exhibit similar overall accuracy (i.e., $mAP50_{box}$ and $mAP50_{mask}$ metric values), the Cascade Mask R-CNN outperforms the Mask R-CNN in achieving a balanced accuracy between the two classes. For example, the Cascade Mask R-CNN obtains similar accuracy for free particles (AP_{box} = 91.0 % and AP_{mask} = 91.0 %) and entrapped particles (AP_{box} = 90.1 % and AP_{mask} = 90.1 %). However, the Mask R-CNN exhibits a 3.8 % difference in AP_{box} and a 3.1 % difference in AP_{mask} between free particles and entrapped particles. Thus, we selected the Cascade Mask R-CNN with ResNet50 backbone as the best model architecture among the tested architectures, and used it in Experiment 2.

Table 4 shows the confusion matrix of the Cascade Mask R-CNN model for particle detection. We found that most particles are correctly detected in test set, including 59 entrapped particle cases and 76 free particle cases, while some particles are undetected (15 cases), some background noise in images is falsely detected as particles (15 cases), and few particles are wrongly classified (7 cases). In addition, the model demonstrates a strong capability to distinguish these two particle

Table 2
Training time and validation accuracy of seven architectures.

Model architecture	Backbone	Learning rate	Training time (min)	Validation accuracy	
				$mAP50_{box}$	$mAP50_{mask}$
Mask R-CNN	ResNet50	0.01	207	**84.0 %**	**86.6 %**
		0.001	198	83.4 %	85.9 %
		0.0001	200	44.6 %	47.4 %
		0.00001	227	0.0 %	0.0 %
	ResNet101	0.01	261	**86.2 %**	86.5 %
		0.001	270	82.8 %	**86.8 %**
		0.0001	269	49.7 %	55.1 %
		0.00001	266	0.0 %	0.0 %
Cascade Mask R-CNN	ResNet50	0.01	256	**87.0 %**	**87.8 %**
		0.001	240	86.4 %	86.5 %
		0.0001	247	65.8 %	67.5 %
		0.00001	244	0.0 %	0.0 %
	ResNet101	0.01	300	86.8 %	87.9 %
		0.001	306	85.1 %	85.9 %
		0.0001	301	69.5 %	72.9 %
		0.00001	303	0.0 %	0.0 %
Yolact	ResNet50	0.01	–	–	–
		0.001	183	78.6 %	72.9 %
		0.0001	183	**84.1 %**	**83.8 %**
		0.00001	201	81.3 %	81.5 %
	ResNet101	0.01	–	–	–
		0.001	258	**83.4 %**	**83.5 %**
		0.0001	247	79.1 %	82.0 %
		0.00001	258	80.0 %	82.2 %
YOLOv8	CSPDarknet53	0.01	30	**70.7 %**	**51.3 %**
		0.001	31	63.1 %	49.5 %
		0.0001	31	48.7 %	41.9 %
		0.00001	31	14.0 %	10.0 %

Note: the bold entities are the best results of each model architecture on validation subset.

Table 3
Performance of different model architectures on the test set.

Model architecture	Backbone	Learning rate	$mAP50_{box}$	$mAP50_{mask}$	$AP50_{box}$ per class		$AP50_{mask}$ per class	
					Entrapped particle	Free particle	Entrapped particle	Free particle
Mask R-CNN	Resnet50	0.01	89.2 %	90.3 %	87.3 %	91.1 %	88.8 %	91.9 %
	Resnet101	0.001	84.9 %	88.0 %	78.4 %	91.5 %	84.4 %	91.5 %
Cascade Mask R-CNN	Resnet50	0.01	**90.6 %**	**90.6 %**	90.1 %	91.0 %	90.1 %	91.0 %
	Resnet101	0.01	86.5 %	88.0 %	84.8 %	88.3 %	85.6 %	90.4 %
Yolact	Resnet50	0.0001	78.2 %	85.9 %	75.0 %	81.4 %	86.6 %	85.2 %
	Resnet101	0.001	80.3 %	85.3 %	78.5 %	82.1 %	84.8 %	85.7 %
YOLOv8	CSPDarknet53	0.01	74.6 %	56.4 %	67.2 %	81.9 %	54.0 %	58.9 %

Note: the bold entities are the best results of model architectures.

Table 4
Confusion matrix of the Cascade Mask R-CNN model for particle detection.

		Predicted class		
		Entrapped particle	Free particle	Background
True class	Entrapped particle	**59**	4	8
	Free particle	3	**76**	7
	Background	4	11	–

Note: the bold entities are the number of particles detected correctly.

classes. For example, the model only detects 4 entrapped particles as free particles, and identifies 3 free particles as entrapped particles. These results shows that the DL model can provide an alternative method for automatically detecting and quantifying free particles and entrapped particles. Furthermore, it could be used to investigate the interaction between microparticles and biomass in biological wastewater treatment process.

4.2. Experiment 2: DL versus ImageJ processing method

Table 5 shows the detection performance and processing time of the DL and ImageJ methods on the test subset. The DL method consistently outperforms the ImageJ method across all evaluation metrics. Out of 157 ground-truth particles in test subset, the DL method (TP = 135) correctly detects 34 more particles than the ImageJ method (TP = 101). This results in an improvement of 21.7 % in *micro-R* (86 % vs 64.3 %), due to a higher number of TPs. In addition, the DL method (FP = 22) yields 17 fewer misdetections compared to the ImageJ method (FP = 39). Thus, the DL method achieves an improvement of 13.8 % in *micro-P*, and an improvement of 18.0 % in *micro-F1*, compared to the ImageJ method. Furthermore, the DL method can accurately classify 135 particles into "free particles" and "entrapped particles" categories, thanks to its feature-learning ability. The DL models, especially CNNs, can learn complex feature representations from input data by multiple layers of nonlinear information processing (Jia et al., 2023a). This enables them to effectively handle complex computer vision tasks. In contrast, the ImageJ method, which does not involve leaning from data, can only identify particles as a single generic "particle" category. It requires additional post-processing to distinguish between "free particles" and "entrapped particles".

Table 5
Detection performance of deep learning and ImageJ methods on the test subset.

Processing method	Detection accuracy						Processing time
	TP	FN	FP	Micro-P	Micro-R	Micro-F1	
Deep learning	**135**	**22**	**22**	**86.0 %**	**86.0 %**	**86.0 %**	**<1 min**
ImageJ	101	56	39	72.1 %	64.3 %	68.0 %	>6 h

Note: the bold entities are the best results of processing methods.

In terms of processing time, manual parameter tuning for ImageJ processing usually costs 5–10 min for one image. Thus, manually processing all 70 images in test subset costs at least 6 h. Nevertheless, the DL method can automatically process all 70 images within 1 min without manual parameter tuning. The DL method significantly reduce the manual effort and processing time, compared to conventional ImageJ processing method.

Fig. 5 shows particle detection results of the DL method and ImageJ processing method on four typical images. The results show that the DL method significantly outperforms the ImageJ method on detecting particles in microscope images. The accuracy of ImageJ processing method usually decreases due to the infection of sludge flocs, small and big granules. Fig. 5(a) includes some irregular particles and biomass in a clean background. We observed that the DL method accurately identifies four free particles, while the ImageJ method only correctly detects three items (threshold 79–136, output size 8000–45,000 μm^2). The free particle close to biomass is not detected, and the ImageJ method erroneously detects four small biomass as particles. Fig. 5(b) includes particles and sludge flocs (Diameter < 0.2 mm). The DL method correctly recognizes two of three ground-truth entrapped particles, but the ImageJ method only detects one particle (threshold 87–105, output size 9800–9900 μm^2). Furthermore, if the output size is adjusted to a wider range, many small sludge flocs are detected as particles erroneously. Fig. 5(c) includes particles and small granules (diameter is 0.2–1.0 mm). The DL method correctly detects two of three ground-truth entrapped particles and one free particle. The ImageJ method (threshold 123–137, output size 5100–5200 μm^2) only detects one entrapped particle and a free particle correctly. Similar to Fig. 5(b), many small biomass are wrongly detected as particles if the analyzed size range becomes bigger. Fig. 5(d) represents the mixture of particles and big granules (diameter is 2.0–3.1 mm). The DL method accurately identifies all the ground-truth entrapped particles, while the ImageJ method only identifies one of them (threshold 33–74, output size 21,000–22,000 μm^2). The lower detection accuracy of the ImageJ method can be mainly attributed to the large number of sludge flocs or small granules present in the background of images. Their presence significantly affects the measurement results of ImageJ (without post-polishment). If the output size is adjusted bigger, the ImageJ method can yield more "particles", while most of them are biomass. Thus, ImageJ method yields more FPs than the DL method (see Table 5).

Fig. 6 compares the particle areas predicted by the DL method (135 TPs), with their ground-truth measurements. The results show a strong correlation, with an R^2 value of 0.9944, indicating an excellent fit between the particle areas predicted by the DL method and those manually measured. Furthermore, the DL method demonstrates consistent accuracy in predicting both large (25,000 μm^2 < area < 50,000 μm^2) and small (area < 25,000 μm^2) particles and measuring their areas, with

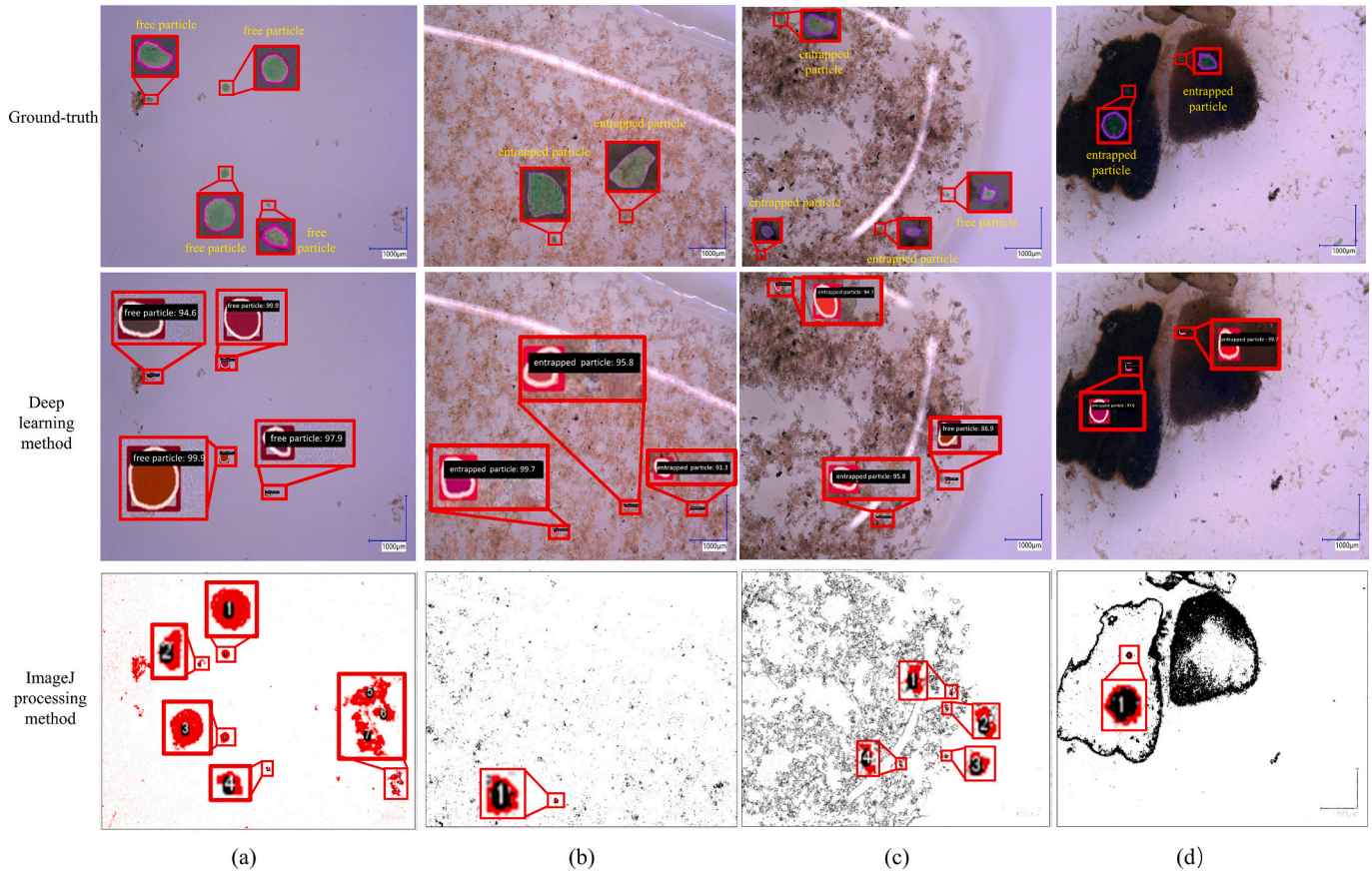


Fig. 5. Detection results of the DL method (middle) and ImageJ processing method (bottom). The objects in these four example images are (a) microparticles and biomass, (b) microparticles and sludge flocs (D < 0.2 mm), (c) microparticles and small granules (D = 0.2–1.0 mm), and (d) microparticles and big granules (D = 2.0–3.1 mm).

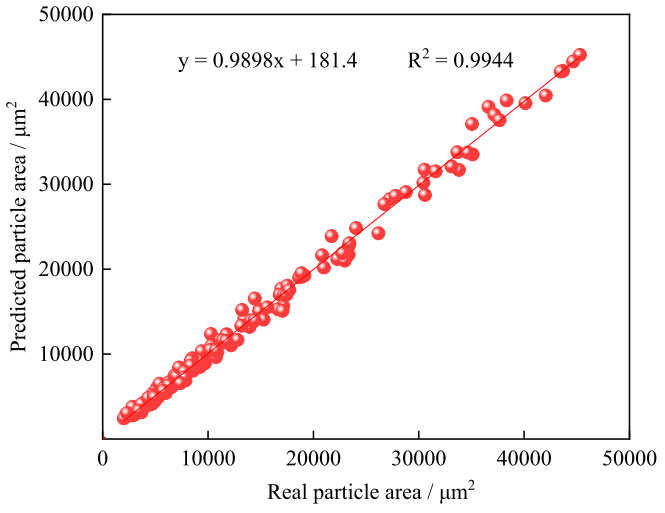


Fig. 6. Comparison of the areas of 135 particles correctly detected by the deep learning method in the test subset with their corresponding ground-truth measurements.

minimal variance observed in model performances across different particle sizes.

4.3. Limitations

DL methods are increasingly applied in various aspects of wastewater treatment, including energy conservation and sludge bulking control (Wang et al., 2024; Zheng et al., 2022). Organic matter is crucial for biological wastewater treatment systems. Many organic matter in actual sewage existed in the form of particles, represented by particle chemical oxygen demand (pCOD). In municipal wastewater, the pCOD typically constitutes >50 % of the total COD, with particle sizes ranging from 0.45 to 250.00 μm (Alondra et al., 2021). The interaction (e.g., attach and detach) between pCOD and biomass (e.g., activated sludge and granular sludge) plays a critical role in the efficiency of sewage treatment. Therefore, this study focuses on developing a new method to analyze the interaction between particles and biomass effectively and efficiently.

Although the above experiment results indicate the cost-effectiveness of DL methods, we acknowledged several limitations in our dataset and methodology that require further improvement for real-world applications. Firstly, the dataset used in Experiment 1 contains a limited amount of data for model training and validation. While the detection and segmentation accuracy of the models developed in Experiment 1 is acceptable ($mAP50_{box} \approx 90$ %, $mAP50_{mask} \approx 90$ %), more images are needed to develop a sufficiently robust model. Arya et al. (2020) suggest that image classification typically requires over 5000 labeled images per class to develop a robust and reliable detection model. Similar requirement may be applicable to the instance segmentation task in this study. Secondly, training a model in a supervised learning manner requires a large amount of labeled data. However, labeling small particles in microscope images is expensive and laborious. This hinders obtaining a robust model for monitoring the particle transfer process in the activated sludge system. Researchers can enhance efforts to bridge this gap by exploiting self-supervised machine learning techniques (Liu et al., 2021). These techniques learns the data feature

representations from unlabeled data without any supervision. Third, future work should focus on a more comprehensive hyper-parameter optimization analysis (e.g., batch size and momentum). Besides, an evaluation of various data augmentation techniques to improve model performance is needed, such as cropping, rotation, noise addition, changes in color or brightness and copy-paste augmentation (Shorten and Khoshgoftaar, 2019). Literature demonstrates benefits for the use of these techniques for improving DL model performance (Jia et al., 2023b; Wu et al., 2024). Finally, this study lacks understanding of the specific features that may drive the detection of microparticles free from biomass and entrapped in biomass. This lack of interpretability may limit the use of DL models in real-life scenarios, where understanding DL models' decision-making process is important. Future work may focus on building safer, and more reliable models by understanding models' decision-making process using some techniques, such as Gradient-weighted Class Activation Mapping (Grad-CAM) (Selvaraju et al., 2017).

5. Conclusion

Monitoring the particle transport process between wastewater and activated sludge is significant for improving the treatment performance. While the traditional micro-level methods (such as microscope image analysis with the conventional ImageJ processing software) are time-consuming and laborious, to address this challenge we proposed a DL method to automatically detect and quantify microparticles free from biomass and entrapped in biomass. Firstly, we introduced a novel dataset including about 700 labeled microscope images with free and entrapped microparticles; then, we evaluated the performance of various DL architectures in an instance segmentation task. The results show that the Cascade Mask R-CNN with ResNet50 backbone performs best, achieving a $mAP50_{box}$ and a $mAP50_{mask}$ of 90.6 %. It also obtains a balanced accuracy for free particles (AP_{box} = 91.0 %) and entrapped particles (AP_{box} = 90.1 %). Furthermore, we observed that the DL method obtained an improvement of 13.8 % in micro-average precision and 21.7 % in micro-average recall, compared against the ImageJ method. The DL method can automatically classify free particles and entrapped particles, while the ImageJ method can only detect particles as a single generic "particle" category. In addition, the DL method processes 70 images within 1 min without manual parameter tuning, which is significantly faster than the ImageJ method with manual parameter tuning (over 6 h).

Researchers could use the proposed DL method to further evaluate the ratio of the number (or area) of free particles to entrapped particles, and the number (area) distribution of each particle. These can provide more insights to explore new phenomena. This study is an initial step in developing a DL-based framework, that can be possibly used in wastewater treatment plants as an intelligent tool to analyze and monitor the interaction between microparticles and biomass in biological wastewater treatment process. We also believe our proposed novel dataset is a great resource for the community to develop such tool.

CRediT authorship contribution statement

Tianlong Jia: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Conceptualization. **Zhaoxu Peng:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Data curation, Conceptualization. **Jing Yu:** Software, Methodology. **Antonella L. Piaggio:** Software. **Shuo Zhang:** Data curation. **Merle K. de Kreuk:** Supervision, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The code for this study is available on https://github.com/TianlongJia/deep_pollutant. The TUD-IPB dataset is available for download from Zenodo at https://doi.org/10.5281/zenodo.13374998.

Acknowledgements

The work is supported by China Scholarship Council (Nos. 202006160032; 202107045010). The authors acknowledge the computing resources provided by Delft High Performance Computing Centre (https://www.tudelft.nl/dhpc).

References

Alondra, A., Stephanie, W., Gudrun, A., Harald, H., 2021. Hydrolysis of particulate organic matter from municipal wastewater under aerobic treatment. Chemosphere 263, 128329.
Arya, D., Maeda, H., Ghosh, S.K., Toshniwal, D., Mraz, A., Kashiyama, T., Sekimoto, Y., 2020. Transfer Learning-based Road Damage Detection for Multiple Countries arXiv preprint arXiv:2008.13101.426.
Bhat, M.A., 2024a. Airborne microplastic contamination across diverse university indoor environments: a comprehensive ambient analysis. Air Qual. Atmos. Health 1–16.
Bhat, M.A., 2024b. A comprehensive characterization of indoor ambient microplastics in households during the Covid-19 pandemic. Air Qual. Atmos. Health 1–17.
Bhat, M.A., 2024c. Unravelling the microplastic contamination: a comprehensive analysis of microplastics in indoor house dust. Indoor Built Environ. 1420326X241248054.
Bhat, M.A., Janaszek, A., 2024. Evaluation of potentially toxic elements and microplastics in the water treatment facility. Environ. Monit. Assess. 196, 475.
Bhat, M., Gedik, K., Gaga, E., 2024. A preliminary study on the natural aging behavior of microplastics in indoor and outdoor environments. Int. J. Environ. Sci. Technol. 21, 1923–1936.
Bolya, D., Zhou, C., Xiao, F., Lee, Y.J., 2019. Yolact: real-time instance segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9157–9166.
Borrelle, S.B., Ringma, J., Law, K.L., Monnahan, C.C., Lebreton, L., McGivern, A., Murphy, E., Jambeck, J., Leonard, G.H., Hilleary, M.A., et al., 2020. Predicted growth in plastic waste exceeds efforts to mitigate plastic pollution. Science 369, 1515–1518.
Cai, Z., Vasconcelos, N., 2019. Cascade r-cnn: high quality object detection and instance segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 43, 1483–1498.
Cao, J., Anwer, R.M., Cholakkal, H., Khan, F.S., Pang, Y., Shao, L., 2020. Sipmask: spatial information preservation for fast image and video instance segmentation. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16. Springer, pp. 1–18.
Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Shi, J., Ouyang, W., et al., 2019. Hybrid task cascade for instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4974–4983.
Chian, E., Fang, W., Goh, Y.M., Tian, J., 2021. Computer vision approaches for detecting missing barricades. Autom. Constr. 131, 103862.
De Kreuk, M., Kishida, N., Van Loosdrecht, M., 2007. Aerobic granular sludge–state of the art. Water Sci. Technol. 55, 75–81.
Delft High Performance Computing Centre (DHPC), 2022. DelftBlue Supercomputer (Phase 1). https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase1.
Dutta, A., Gitahi, J.M., Ghimire, P., Mink, R., Peteinatos, G., Engels, J., Hahn, M., Gerhards, R., 2018. Weed detection in close-range imagery of agricultural fields using neural networks. Publ. DGPF 27, 633–645.
Grass, R.N., Schalchli, J., Paunescu, D., Soellner, J.O., Kaegi, R., Stark, W.J., 2014. Tracking trace amounts of submicrometer silica particles in wastewaters and activated sludgeusing silica-encapsulated dna barcodes. Environ. Sci. Technol. Lett. 1, 484–489.
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.
He, K., Gkioxari, G., Doll'ar, P., Girshick, R., 2017. Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2961–2969.
Inbar, O., Shahar, M., Gidron, J., Cohen, I., Menashe, O., Avisar, D., 2023. Analyzing the secondary wastewater-treatment process using faster r-cnn and yolov5 object detection algorithms. J. Clean. Prod. 416, 137913.
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., 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.
Jocher, G., Chaurasia, A., Qiu, J., 2023. Ultralytics YOLO. https://github.com/ultralytics/ultralytics.
Lan, T., Hu, Y., Cheng, L., Chen, L., Guan, X., Yang, Y., Guo, Y., Pan, J., 2022. Floods and diarrheal morbidity: evidence on the relationship, effect modifiers, and attributable risk from Sichuan province, China. J. Glob. Health 12.

Lebreton, L.C., Van Der Zwet, J., Damsteeg, J.W., Slat, B., Andrady, A., Reisser, J., 2017. River plastic emissions to the world's oceans. Nat. Commun. 8, 1–10.
Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Doll'ar, 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, 857–876.
Loshchilov, I., Hutter, F., 2016. Sgdr: Stochastic Gradient Descent With Warm Restarts arXiv preprint arXiv:1608.03983.
Mesquita, D.P., Amaral, A.L., Ferreira, E.C., 2013. Activated sludge characterization through microscopy: a review on quantitative image analysis and chemometric techniques. Anal. Chim. Acta 802, 14–28.
Mohamad, A.F., Maria, N., Azmi, A., Khalida, M., 2018. Mass transfer kinetics of biosorption of nitrogenous matter from palm oil mill effluent by aerobic granules in sequencing batch reactor. Environ. Technol. 39, 2151–2161.
Nakama, T., 2009. Theoretical analysis of batch and on-line training for gradient descent learning in neural networks. Neurocomputing 73, 151–159.
Noyan, K., Allı, B., Okutman Taş, D., S¨ozen, S., Orhon, D., 2017. Relationship between cod particle size distribution, cod fractionation and biodegradation characteristics in domestic sewage. J. Chem. Technol. Biotechnol. 92, 2142–2149.
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.
Piaggio, A.L., Soares, L.A., Balakrishnan, M., Guleria, T., de Kreuk, M.K., Lindeboom, R. E., 2022. High suspended solids removal of indian drain water with a down-scaled dissolved air flotation (daf) for water recovery. Assessing water-type dependence on process control variables. Environ. Chall. 8, 100567.
Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: towards real-time object detection with region proposal networks. Adv. Neural Inf. Proces. Syst. 28, 91–99.
Satoh, H., Kashimoto, Y., Takahashi, N., Tsujimura, T., 2021. Deep learning-based morphology classification of activated sludge flocs in wastewater treatment plants. Environ. Sci.: Water Res. Technol. 7, 298–305.
Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch, S., Rueden, C., Saalfeld, S., Schmid, B., et al., 2012. Fiji: an open-source platform for biological-image analysis. Nat. Methods 9, 676–682.
Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D., 2017. Grad-cam: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 618–626.
Shen, Y., Sun, P., Ye, L., Xu, D., 2023. Progress of anaerobic membrane bioreactor in municipal wastewater treatment. Sci. Adv. Mater. 15, 1277–1298.
Shorten, C., Khoshgoftaar, T.M., 2019. A survey on image data augmentation for deep learning. J. Big Data 6, 60. https://doi.org/10.1186/s40537-019-0197-0.
Sorasan, C., Edo, C., González-Pleiter, M., Fernández-Piñas, F., Leganés, F., Rodríguez, A., Rosal, R., 2022. Ageing and fragmentation of marine microplastics. Sci. Total Environ. 827, 154438.
Sven, S., Christiane, Z., Olal, A., Creta, F., Peter, G., 2017. Shift in mass transfer of wastewater contaminants from microplastics in the presence of dissolved substances. Environ. Sci. Technol. 51, 12254–12263.
Tran, H.T., Hadi, M., Nguyen, T.T.H., Hoang, H.G., Nguyen, M.K., Nguyen, K.N., Vo, D.V. N., 2023. Machine learning approaches for predicting microplastic pollution in peatland areas. Mar. Pollut. Bull. 194, 115417.
Wang, Z., Zhou, X., Wang, H., Huang, Z., Ji, J., Peng, Z., Jiang, K., 2024. XGB-SEGA coupled energy saving method for wastewater treatment plants. Appl Water Sci 14, 29.
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. 261, 121999.
Xu, D., Wu, S., Yan, A., Chen, Z., Xu, J., Gu, C., Qi, Y., Wu, S., 2024. Efficient recycling of sewage water in a polyester integrated industry: a case study. Desalin. Water Treat. 100508.
Yang, G., Tang, Y., Liu, X., Wang, L., Qin, L., Li, D., Shen, X., Kong, C., Zhai, W., Fodjo, E. K., et al., 2024. Determination of free glycidol and total free monochloropropanediol in fish and krill oil with simple aqueous derivatization and high-performance liquid chromatography–tandem mass spectrometry. Foods 13, 2340.
Yurtsever, M., Yurtsever, U., 2019. Use of a convolutional neural network for the classification of microbeads in urban wastewater. Chemosphere 216, 271–280.
Zhang, L., Jiang, S., Jia, Y., Zhang, M., Guo, J., 2024. Effects of Na^{+}/H_2O_2 on nitrogen removal and sludge activity: performance and mechanism. J. Environ. Chem. Eng. 113194.
Zhao, L., Dai, T., Qiao, Z., Sun, P., Hao, J., Yang, Y., 2020. Application of artificial intelligence to wastewater treatment: a bibliometric analysis and systematic review of technology, economy, management, and wastewater reuse. Process Saf. Environ. Prot. 133, 169–182.
Zhen, J., Wang, Z.B., Ni, B.J., Ismail, S., El-Baz, A., Cui, Z., Ni, S.Q., 2024. Synergistic integration of anammox and endogenous denitrification processes for the simultaneous carbon, nitrogen, and phosphorus removal. Environ. Sci. Technol. 58, 10632–10643.
Zheng, Y., Peng, Z., Xia, H., Zhang, W., 2022. Prediction of Sludge Settleability Through Artificial Neural Networks With Optimized Input Variables, 36, pp. 694–703.