Usefulness of synthetic datasets for diatom automatic detection using a deep-learning approach

Aishwarya Venkataramanan, Pierre Faure-Giovagnoli, Cyril Regan, David Heudre, Cécile Figus, Philippe Usseglio-Polatera, Cedric Pradalier, Martin Laviale

Usefulness of synthetic datasets for diatom automatic detection using a deep-learning approach

Aishwarya Venkataramanan[a,b,c], Pierre Faure-Giovagnoli[*,†,b], Cyril Regan[d], David Heudre[e], Cécile Figus[‡a], Philippe Usseglio-Polatera[a], Cédric Pradalier[b,c], Martin Laviale[§,a,c]

[a] *Université de Lorraine, CNRS, LIEC, F-57000 Metz, France*
[b] *Georgia Tech Lorraine, CNRS IRL 2958, F-57070 Metz, France*
[c] *LTSER-"Zone Atelier Moselle", F-57000 Metz, France*
[d] *Université de Lorraine, CNRS, LORIA, F-54000, Nancy France*
[e] *Direction Régionale de l'Environnement, de l'Aménagement et du Logement Grand Est, Metz, France*

Abstract

Benthic diatoms are unicellular microalgae that are routinely used as bioindicators for monitoring the ecological status of freshwater. Their identification using light microscopy is a time-consuming and labor-intensive task that could be automated using deep-learning. However, training such networks relies on the availability of labelled datasets, which are difficult to obtain for these organisms. Herein, we propose a method to generate synthetic microscopy images for training. We gathered individual objects, i.e. 9230 diatoms from publicly available taxonomic guides and 600 items of debris from available real images. We collated a comprehensive dataset of synthetic microscopy images including both diatoms and debris using seamless blending and a combination of parameters such as image scaling, rotation, overlap and diatom-debris ratio. We then performed sensitivity analysis of the impact of the synthetic data parameters for training state-of-the art networks for horizontal and rotated bounding box detection (YOLOv5). We first trained the networks using the synthetic dataset and fine-tuned it to several real image datasets. Using this approach, the performance of the detection network was improved by up to 25% for precision and 23% for recall at an Intersection-over-Union(IoU) threshold of 0.5. This method will be extended in the future for training segmentation and classification networks.

[*]Present address:Univ Lyon, INSA Lyon, CNRS, UCBL, LIRIS UMR 5205, Villeurbanne, France
[†]Present address:Compagnie Nationale du Rhône, Lyon, France
[‡]Present address:Institute of Marine and Environmental Sciences, University of Szczecin, ul. Mickiewicza 18, 70-383 Szczecin, Poland
[§]Corresponding author:martin.laviale@univ-lorraine.fr

Keywords: synthetic dataset, images, diatoms, automatic detection, deep learning

1. Introduction

Diatoms are unicellular microscopic algae found in all aquatic environments, where they play a fundamental role in ecosystem functioning (Lobo et al., 2016). They are widely used in water quality monitoring due to their high sensitivity to water quality (e.g., temperature, pH, nutrients, micropollutants) and habitat conditions (Lobo et al., 2016). This is especially the case for freshwater benthic diatoms (i.e, diatoms living attached to substrates in rivers and lakes), in the context of the European Water Framework Directive implementation (Coste et al., 2009). Diatom-based biological indices are based on the occurrence of indicator species in a natural assemblage. Diatom species are identified based on morphological features (e.g., size, ornamentation) by human experts using light microscopy. This task can be time-consuming and subject to multiple biases due to the quality of the instrument or the level of expertise in diatom taxonomy. Therefore, attempts to automate the process have been ongoing since the early 90s (Du Buf et al., 1999; Pedraza et al., 2017, 2018; Kloster et al., 2020).

Automatic diatom identification consists of two sequential steps: detecting diatoms among other objects in microscopy images, followed by classification of the detected diatoms into their respective taxa. Processing typical microscopy images for diatom identification is difficult for several reasons. Firstly, individual diatoms may be difficult to distinguish from the background due to a lack of contrast. Moreover, diatoms are rarely clearly separated from each other, and debris is often observed, resulting in complex overlays and stacking, making diatoms difficult to recognize and isolate. These difficulties make the detection step challenging to automate.

Object detection procedures using microscopy images rely on both segmentation and bounding box detection. Previous work on microorganism segmentation includes methods such as thresholding, gradient-based, deformable models and feature-based approaches (Verikas et al., 2012; Kloster et al., 2014; Borges et al., 2015; Rojas Camacho et al., 2017). At best, these methods are semi-automatic and require manual fine-tuning of parameters, which is impractical. For automatic bounding box detection, Convolutional Neural Networks (CNNs) can improve performance in terms of reliability and robustness (Girshick, 2015; Redmon et al., 2016). This has been demonstrated in microscopy image detection, mainly for planktonic organisms (Shi et al., 2019; Li et al.,

2021), and also specifically for benthic diatoms (Salido et al., 2020). However, the approaches still suffer from overlapping bounding boxes when diatoms are close to each other. To overcome these difficulties, one can use rotated bounding boxes (Li et al., 2020; Zhong and Ao, 2020). By estimating the orientations of objects, rotated bounding boxes are fitted around objects, resulting in minimal overlaps and thus cleaner detection.

Training supervised deep-learning networks requires thousands of labelled microscopy images. Most of the time, labelling is performed manually. In addition to being a time-consuming and laborious process, human intervention is also a source of error and bias (Sager et al., 2021). Thus, when available labelled data are limited, one can use synthetic datasets for pre-training the network before fine-tuning it with a real dataset (Li et al., 2021). Depending on the method adopted, images generated could be fully or partially synthetic. Learning-based approaches such as Variational Auto Encoders (Kingma and Welling, 2013) and Generative Adversarial Networks (Goodfellow et al., 2014) generate fully synthetic data, where no part of the generated images contain real images. However, to generate high-quality images, these networks require thousands of images to train on, which is impractical for applications. In the present work, we adopted a method to generate partially synthetic data, (i.e., where parts of the generated images contain real images). In the case of freshwater benthic diatoms, hundreds of digitized individual images are publicly available for various diatom species. In theory, they can also be retrieved in order to create thousands of synthetic images similar to real digital ones taken by a human expert. This offers a unique opportunity for the design of a relevant pre-labelled dataset for bounding boxes, as each individual is already boxed, allowing the training of automatic detection networks. Our method for synthetic data generation closely follows Copy-Paste augmentation as implemented for instance segmentation (Dwibedi et al., 2017; Ghiasi et al., 2021). In Copy-Paste augmentation, the generated images use real images as background, then images of individual objects of interest are pasted at random locations on the real images. However, while pasting, the global consistency of the final images is not considered. This means that an object could be placed at locations where they usually do not occur. This could be detrimental in the case of single-stage architectures such as YOLO, which predicts from full images and hence, takes into account the global context of the images while making predictions. By contrast, we generate a synthetic background and place the

individual diatoms and debris images, while considering the global consistency of the final image. To generate realistic images, we consider different parameters while pasting individual objects, such as scaling, rotation, level of overlap of diatom and diatom-debris ratio. Finally, we perform sensitivity analysis to study the impact of these parameters on the performance of the network by evaluating using real datasets.

To summarize, we evaluated the usefulness of synthetic datasets for the automatic detection of diatoms in microscopy images using deep-learning. We first propose a method to generate thousands of synthetic datasets based on an off-the-shelf seamless blending approach. This dataset is used for pre-training a state-of-the-art detection network, YOLOv5 (Jocher et al., 2020), in order to process real images with different levels of complexity (number and diversity of objects, including different diatom species and debris). Based on our experiments, fine-tuning the network previously trained on the synthetic dataset performed best in diatom detection, regardless of the type of real images. This confirms that synthetic datasets provide a useful alternative when the available labeled dataset is limited for training deep neural networks.

2. Materials and Methods

2.1. Initial Datasets

We used the following datasets for our analyses:

2.1.1. Individual diatom images

Individual diatom images were collected from several open-access taxonomic atlases (Bey and Ector, 2013; Lalanne-Cassou and Voisin, 2013; Peeters and Ector, 2018). From the atlases in PDF format, we automatically extracted diatom images using PDF scraping. Our final dataset consisted of 9230 grayscale images of individual diatoms belonging to 166 diatom species.

2.1.2. Real microscopy images

Real images were generated from two types of diatom samples: (1) natural samples collected in the field (Cleurie Stream, Vosges, France, hereafter referred to as CLEURIE) on hard mineral substrates and fixed by adding ethanol, showing natural communities consisting of various diatom species; (2) a sample collected from a lab culture of the diatom *Nitzchia palea* (hereafter referred

to as NPAL). Each liquid sample was subjected to oxidation using hot hydrogen peroxide in order to remove organic material, rinsed with deionized water, then mounted on permanent microscope slides with Naphraxinto, a high refractive index resin in order to be visualized at 1000× magnification using light microscopy. CLEURIE and NPAL datasets finally consist of 120 and 175 images, respectively, each one representing a microscope view field containing multiple diatoms and debris.

2.1.3. Debris

To create the synthetic images of microscope slides, we needed to take into account the variable occurrence of debris potentially present on microscope images. From the real microscope images, we manually extracted 600 debris objects of different shapes and sizes.

2.1.4. Synthetic Microscope Image Generation

The synthetic dataset should closely mimic the real images obtained from light microscopy.

Let us consider the following procedure to create a seamless image I of width W and height H with images from the individual diatom dataset D:

1. Init I', empty image of dimensions (W, H).
   - Pick a subset $S_i \subset D$ composed of individual diatom and debris images.
   - For each image $T_{ij} \in S_i$ of width w_{ij} and height h_{ij},
     - Apply data augmentation. Refer to Sec. 3.1 for the different augmentations and the parameters used.
     - Find a random position P_{ij} for T_{ij} on I' such that it does not overlap any previously added diatom or debris images or the borders of I' by more than a certain threshold percentage.
2. Init I, empty image of dimensions (W, H). Let B be the set of pixels $p^{I'}$ of I' comprising the borders of T_{ij}. Set the value of each pixel p_i^I of I as a weighted arithmetic mean such that:

$$p_i^I = \frac{\sum\limits_{p_j^{I'} \in B} val(p_j^{I'}) w(p_i^I, p_j^{I'})}{\sum\limits_{p_j \in B} w(p_i^I, p_j^{I'})} \tag{1}$$

$$w(p, p') = e^{-||p'-p||_2} \tag{2}$$

where $val(p)$ is the gray value of pixel p. This step generates the gray background, similar to those obtained from light microscopy.

3. Finally, place T_{ij} at P_{ij} on I using the Poisson Editing method for seamless blending (Pérez et al., 2003).

2.2. Object Detection

The deep-learning object detection networks can be distinguished into two types: one-stage, and two-stage. The one-stage family (Liu et al., 2016; Redmon et al., 2016; Lin et al., 2017) generally uses a unique network to extract and classify the bounding boxes, while two-stage networks (Girshick et al., 2014; Girshick, 2015) first use a Region Proposal Network to propose multiple Regions of Interest (RoI) which are subsequently classified using a second generic backbone architecture. One-stage detection networks are faster since they combine detection and classification in a single architecture, but are generally less accurate than two-stage networks. For our objective, we chose one stage YOLOv5 (Jocher et al., 2020) for detection due to its overall performance in terms of speed and accuracy. However, future work would incorporate Neural Architecture Search (O'Neill et al., 2021; Xue et al., 2021; Xue and Qin, 2022) to choose the optimal architecture. YOLOv5 is a real-time CNN for localizing objects on images. YOLO-based networks function by dividing the input into several grids responsible for the detection and localization of the objects they contain. The inputs to the network are images from microscopy, and the network predicts the coordinates of the bounding boxes. We used YOLOv5 for two types of detection: horizontal bounding boxes and rotated bounding boxes. While the standard YOLOv5 (Jocher et al., 2020) regresses the values of the rectangular bounding boxes, YOLOv5 for rotated bounding boxes[5] estimates the orientation in addition to the box coordinates.

3. Experiments

3.1. Baseline

Four augmentation techniques were used in order to generate the synthetic dataset: scaling, rotation, overlap between diatoms and debris, and diatom to debris ratio. To determine the

[5]https://github.com/XinzeLee/RotateObjectDetection

sensitivity of each of the augmentation methods while evaluating using real microscopy images, and to choose the optimal parameter values, we performed a grid-search on the parameters without changing the neural network hyperparameters as summarized in Table 1. The definition and significance of each of these techniques and the parameter values are given below.

1. **Scaling**: Scaling refers to resizing the images of individual diatoms and debris while creating the synthetic dataset. This was carried out to represent the real microscopy images where diatoms belonging to a species can occur at varying scales. For an accurate scaling value, one could consider the actual size range in which the diatoms occur naturally. However, for our analysis, we randomly chose a scaling value from an exponential distribution to consider a wide range of scales for each diatom species. For the grid-search, we considered situations with and without scaling to generate the data.
2. **Rotation**: The individual diatom images were randomly rotated at any angle between 0 and 360°. Our grid-search considered two conditions: with and without rotating the images.
3. **Overlap**: For each individual diatom on the synthetic image, this refers to the level of overlap allowed between other diatoms or debris. We varied the level of overlap between the objects from 0 to 25%. The maximum threshold value was chosen as 25% to avoid hidden objects that would bias training and evaluation. For our parameter-tuning, we considered two cases: no overlap between the objects and overlap between objects.
4. **Diatom-debris ratio**: This refers to the ratio between the number of diatoms and the number of debris objects while generating the images. We considered the following range of values:(1) diatom-debris ratio between 2 and 5; (2) diatom-debris ratio between 0.1 and 0.5. For the above two cases, the number of diatoms was between 6 and 10, and we varied the number of debris objects to achieve the desired ratio.

To study the impact of the synthetic dataset while training on the real datasets, we performed the following experiments: (1) in the 'Synthetic' experiment, we trained and evaluated the network on the synthetic images; (2) in the 'Zero-shot learning' experiment, we trained the network with the synthetic dataset and evaluated using the available real dataset; (3) in the 'Training from scratch' experiment, we trained the network with only the real dataset and evaluated using the real dataset; (4) in the 'Fine-tuning' experiment, we used the model trained on the synthetic

Table 1: Augmentation methods and the parameters used to generate synthetic dataset

Dataset name	Parameters
D1	No rotation; Scaling factor=1; Diatom-debris ratio=[2,5]; Overlap=0;
D2	No rotation; Scaling factor=1; Diatom-debris ratio=[2,5]; Overlap=0.25;
D3	No rotation; Scaling factor=1; Diatom-debris ratio=[0.1,0.5]; Overlap=0;
D4	No rotation; Scaling factor=1; Diatom-debris ratio=[0.1,0.5]; Overlap=0.25;
D5	No rotation; Scaling factor=exponential; Diatom-debris ratio=[2,5]; Overlap=0;
D6	No rotation; Scaling factor=exponential; Diatom-debris ratio=[2,5]; Overlap=0.25;
D7	No rotation; Scaling factor=exponential; Diatom-debris ratio=[0.1,0.5]; Overlap=0;
D8	No rotation; Scaling factor=exponential; Diatom-debris ratio=[0.1,0.5]; Overlap=0.25;
D9	With rotation; Scaling factor=1; Diatom-debris ratio=[2,5]; Overlap=0;
D10	With rotation; Scaling factor=1; Diatom-debris ratio=[2,5]; Overlap=0.25;
D11	With rotation; Scaling factor=1; Diatom-debris ratio=[0.1,0.5]; Overlap=0;
D12	With rotation; Scaling factor=1; Diatom-debris ratio=[0.1,0.5]; Overlap=0.25;
D13	With rotation; Scaling factor=exponential; Diatom-debris ratio=[2,5]; Overlap=0;
D14	With rotation; Scaling factor=exponential; Diatom-debris ratio=[2,5]; Overlap=0.25;
D15	With rotation; Scaling factor=exponential; Diatom-debris ratio=[0.1,0.5]; Overlap=0;
D16	With rotation; Scaling factor=exponential; Diatom-debris ratio=[0.1,0.5]; Overlap=0.25;

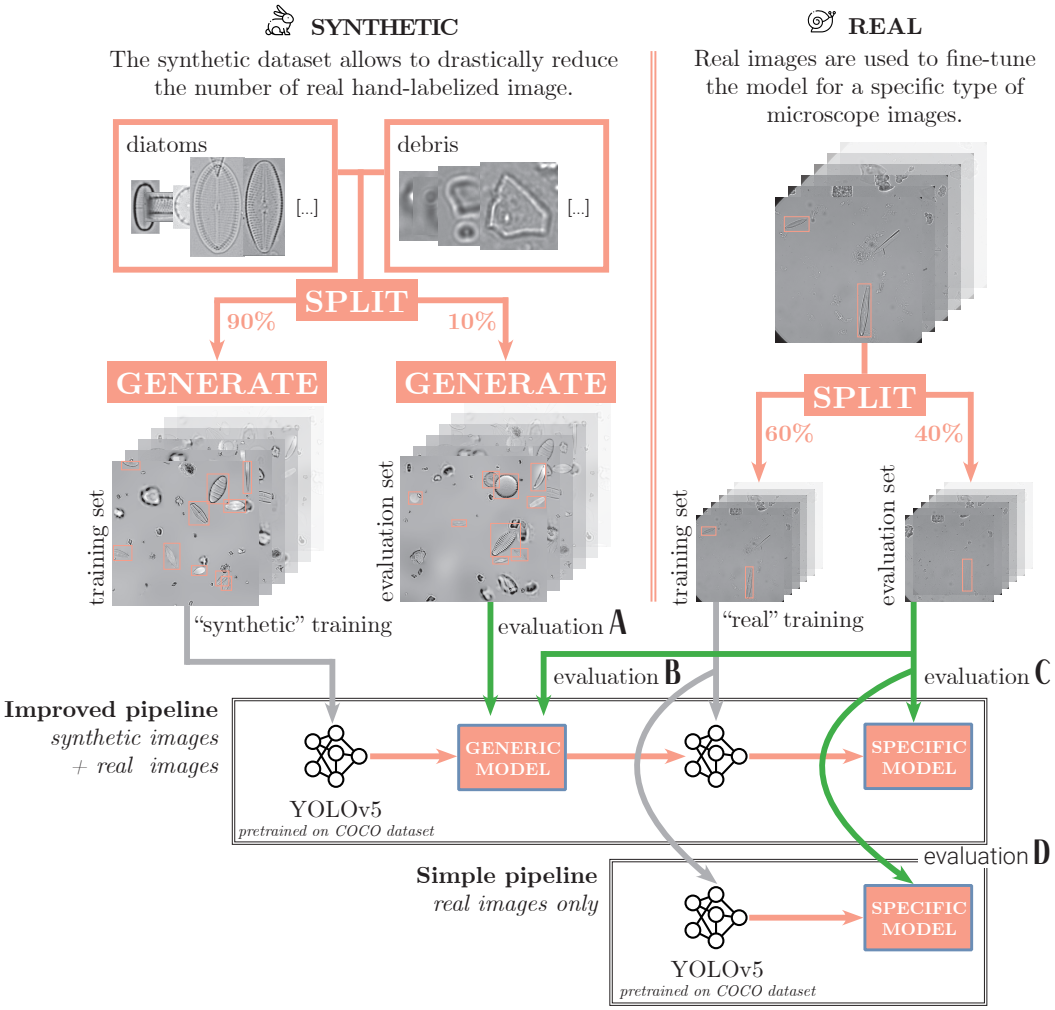

Figure 1: Pipelines for evaluating networks. First, individual diatoms and debris images are used to generate synthetic datasets for training the detection networks. There are four evaluation pipelines: Evaluation A – the network is trained and evaluated on synthetic datasets. Evaluation B – the network is trained on synthetic images and evaluated on real images (zero-shot detection). Evaluation C – the network is trained on a synthetic dataset, fine-tuned to real images, and evaluated on real images. Evaluation D – the network is trained and evaluated on real datasets

dataset to fine-tune to the real dataset and evaluated on the real dataset. Figure. 1 illustrates the pipeline for training and evaluation.

3.2. Evaluation Metrics

Detection is concluded when the intersection over union (IoU), which measures the extent of overlap between the predicted box and the ground truth box, is above a certain threshold. For a fixed threshold of IoU, precision gives the percentage of detections that are correct.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{3}$$

For a fixed IoU threshold, recall gives the number of positive detections out of all the positives in the ground truth.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{4}$$

In addition to these metrics, some Object Detection challenges (VOC, COCO, Open Images) define their own metrics that have become standard evaluation indicators. In this work, we also calculated mean Average Precision for IoU at 0.5 (mAP@0.5) and AP for IoU from 0.50 to 0.95 with a step size of 0.05 (mAP@0.5:0.95) as defined in COCO (2019). We used the same metrics for evaluating the horizontal and rotated bounding boxes.

3.3. Implementation Details

We used Adam optimizer (Kingma and Ba, 2014) and the learning rate was 0.0002. The batch size for training YOLOv5 (horizontal and rotated) was 16. For every category of synthetic dataset (D1-D16 in Table. 1), we generated 4000 images. While training the networks using synthetic and real datasets, we used 80% of the images for training, 10% for validation and 10% for testing. We trained our networks on GeForce GTX 2080 with 8 Gb RAM and GeForce GTX 1080 with 12 Gb RAM. The total training time for sensitivity analysis was 24 days.

4. Results

4.1. Horizontal bounding box detection

Figure 2 shows some examples of the horizontal bounding box detections from YOLOv5.

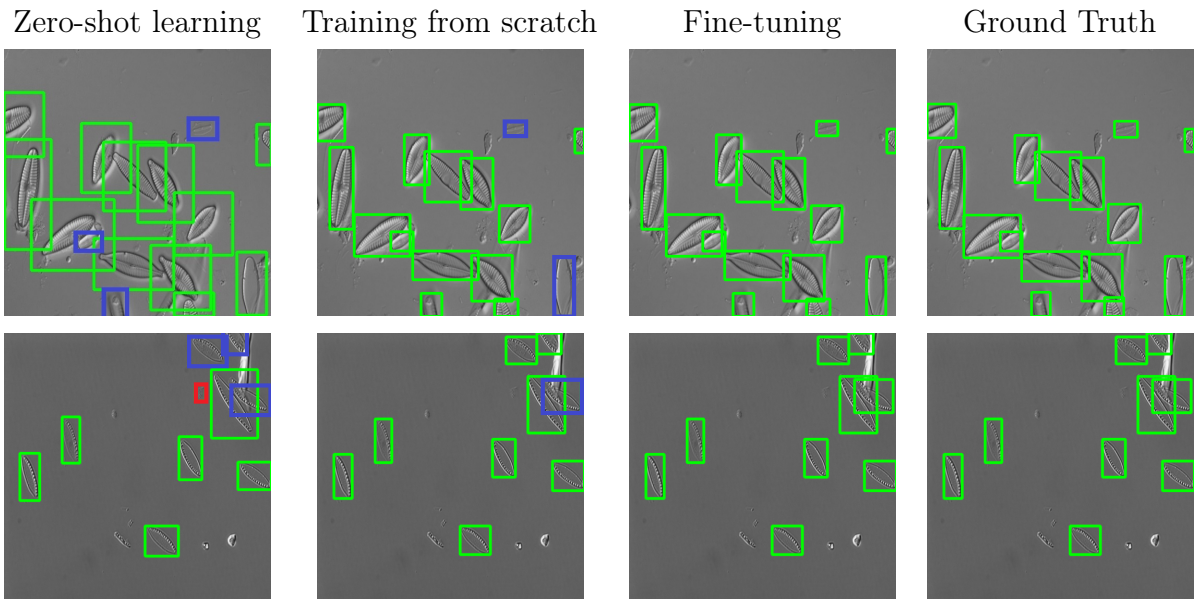

Figure 2: Horizontal bounding box detection obtained from YOLOv5 using CLEURIE (top row) and NPAL (bottom row) real images. Green boxes represent the correct detection; blue indicates false negatives and red boxes indicate false positives.

Table 2: Quantitative evaluation of horizontal bounding box detection. Higher values indicate better performance.

M1-Precision; M2-Recall; M3-mAP@.5; M4-mAP@.5:.95.

Datasets	Synthetic				Zero-shot learning								Fine-tuning							
					CLEURIE				NPAL				CLEURIE				NPAL			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
D1	**1.000**	**1.000**	0.995	**0.990**	0.422	0.461	0.406	0.174	0.848	0.549	0.694	0.413	0.930	0.917	0.974	0.797	0.816	**0.958**	0.918	0.756
D2	0.998	0.998	0.995	0.981	0.632	0.453	0.501	0.224	0.533	0.690	0.603	0.405	0.925	0.923	0.974	0.804	0.842	0.901	0.909	0.745
D3	0.999	**1.000**	0.995	0.973	0.573	0.458	0.512	0.222	0.543	0.620	0.567	0.310	0.939	0.920	0.975	0.802	**0.936**	0.829	**0.933**	**0.786**
D4	0.998	0.999	0.995	0.967	0.605	0.605	0.614	0.254	0.639	0.775	0.683	0.257	0.921	0.934	0.976	0.802	0.837	0.944	0.924	0.775
D5	0.993	0.996	0.995	0.971	0.543	0.535	0.451	0.107	**0.861**	0.795	0.788	**0.308**	0.928	**0.963**	0.972	0.791	0.747	0.915	0.812	0.705
D6	0.995	0.991	0.995	0.968	0.579	0.404	0.444	0.195	0.475	0.423	0.464	0.275	0.933	0.923	0.979	0.791	0.805	0.928	0.903	0.747
D7	0.995	0.992	0.995	0.967	0.512	0.596	0.568	0.308	0.800	0.450	0.547	0.373	0.919	0.946	0.978	0.797	0.812	0.915	0.920	0.779
D8	0.995	0.991	0.995	0.965	0.802	0.513	0.645	0.353	0.766	0.507	0.531	0.284	0.907	0.948	0.971	0.786	0.840	0.887	0.892	0.770
D9	1.000	0.999	0.995	0.955	0.666	0.610	0.652	**0.364**	0.646	0.436	0.461	0.265	0.960	0.900	0.977	0.798	0.793	0.971	0.879	0.757
D10	0.998	0.991	0.995	0.915	0.791	0.642	0.675	0.173	0.805	0.761	0.741	0.253	0.910	0.954	0.977	0.791	0.767	0.972	0.850	0.723
D11	0.999	1.000	0.995	0.932	0.783	0.613	0.621	0.121	0.753	0.690	0.669	0.210	0.916	0.940	0.977	0.787	0.775	0.972	0.871	0.712
D12	0.999	**1.000**	**0.999**	0.946	0.695	0.642	0.670	0.190	0.762	0.859	0.804	0.175	0.910	0.954	0.974	0.784	0.802	0.915	0.899	0.744
D13	0.988	0.990	0.995	0.954	0.681	0.484	0.546	0.194	0.844	0.789	0.861	0.199	0.931	0.928	0.977	0.788	0.841	0.901	0.890	0.728
D14	0.982	0.987	0.994	0.931	**0.800**	**0.700**	**0.776**	0.300	0.710	0.901	0.763	0.235	**0.967**	0.911	**0.981**	**0.806**	0.805	0.930	0.886	0.749
D15	0.982	0.984	0.996	0.943	0.651	0.648	0.630	0.157	0.849	**0.958**	**0.913**	0.244	0.917	0.948	0.975	0.771	0.779	0.943	0.873	0.753
D16	0.980	0.974	0.993	0.922	0.766	0.685	0.750	0.211	0.766	0.924	0.858	0.275	0.933	0.920	0.979	0.789	0.800	**0.958**	0.878	0.738

Table 3: Quantitative evaluation of horizontal bounding box detection when the network is trained using only the real dataset.

Dataset	Precision	Recall	mAP@0.5	mAP@0.5:0.95
CLEURIE	0.840	0.894	0.864	0.611
NPAL	0.744	0.864	0.847	0.594

Tables 2 and 3 show the quantitative metrics for horizontal bounding box detection using YOLOv5 (Jocher et al., 2020). The following observations could be made based on the results:

Scaling: When evaluated on synthetic dataset, the precision, and recall (mIoU=0.50 and mIoU from 0.50 to 0.95) was higher when the images were not scaled (D1-D4; D9-D12 in Table. 2). This is because when there is no scaling, the diatoms are of uniform size, which leads to network overfitting. When evaluated on the real microscopy images, the CLEURIE dataset contained diatoms of varying size due to the presence of different species, and hence the presence of scaling in the synthetic dataset leads to better metric values on both zero-shot learning and fine-tuning. NPAL images contained diatoms of the same species, and all diatoms were of rather uniform size. Hence, the network generalized better when the synthetic dataset had no scaling.

Rotation: When the dataset contained no rotated diatoms (D1-D8), the $\text{mIoU}_{0.5:0.95}$ on the synthetic dataset was superior, but it did not generalize well to the CLEURIE and NPAL datasets. This is because networks detecting horizontal bounding boxes are designed to fit non-rotated objects better, hence the network generalizes well. The network draws tighter and more accurate bounding boxes around these objects, resulting in better mIoU values. However, real datasets contain diatoms present in random orientations. Thus, training datasets D9-D16 generalized better to CLEURIE and NPAL datasets compared with D1-D8.

Overlap: From the results, having overlaps had no significant impact on the precision and recall scores (mIoU=0.50 and mIoU from 0.50 to 0.95) of the network for synthetic and real datasets. However, the best metric values were obtained when a maximum overlap of 25% was allowed. This means that although overlap helps improve network performance, the network was able to generalize well even when the training datasets do not have overlap of diatoms.

Diatom-debris ratio: The network had better metrics when the diatom-debris ratio was in the range [2,5] for both the synthetic and real datasets. For the synthetic datasets, this could

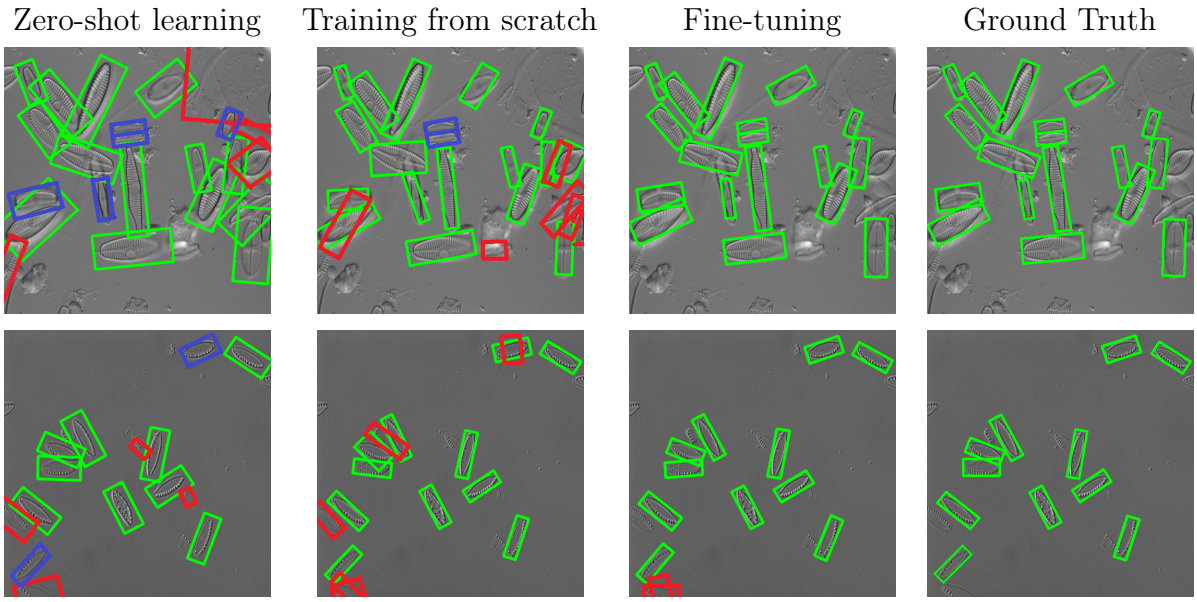

Figure 3: Rotated bounding box detection from YOLOv5 using CLEURIE (top row) and NPAL (bottom row) real images. Green boxes indicate correct detection; blue indicates false negatives, and red boxes indicate false positives.

be explained by the reduced complexity of the images for the network to detect, since the debris objects were widely distributed. For both the CLEURIE and NPAL datasets, the diatom-debris ratio was greater than 1. Hence, when the synthetic dataset was characterized by a higher diatom-debris ratio, the network generalized better for these datasets.

Table 3 shows the results when the network was trained using only the real dataset. Compared with the performance when the network was trained only on real data, there was an improvement of 15% in precision (mIoU=0.50) and 8% in recall (mIoU=0.50) for the CLEURIE dataset and an improvement of 25% in precision (mIoU=0.50) and 11% in recall (mIoU=0.50) for the NPAL dataset when the network was trained on synthetic data and fine-tuned to real data.

4.2. Rotated Bounding Box Detection

Figure 3 shows some examples of the rotated bounding box detection from YOLOv5. Tables 4 and 5 show the quantitative metrics for rotated bounding box detection. Since no rotation was included in D1-D8, we considered datasets D9-D16.

Table 4: Quantitative evaluation of rotated bounding box detection. Higher values indicate better performance.

M1-Precision; M2-Recall; M3-mAP@.5; M4-mAP@.5:.95.

Datasets	Synthetic				Zero-shot learning								Fine-tuning							
					CLEURIE				NPAL				CLEURIE				NPAL			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
D9	**0.983**	0.992	**0.991**	**0.811**	0.302	0.320	0.172	0.037	0.634	0.706	0.618	0.198	0.868	0.958	**0.928**	0.679	0.862	0.955	0.939	0.724
D10	0.975	**0.987**	0.988	0.793	0.536	0.414	0.346	0.027	**0.813**	**0.806**	**0.804**	**0.234**	0.826	0.819	0.865	0.615	0.880	0.991	0.940	0.730
D11	0.968	0.958	0.985	0.750	0.520	0.475	0.372	0.027	0.755	0.838	0.781	0.176	0.868	**0.957**	0.925	**0.699**	0.879	0.982	0.942	0.737
D12	0.972	0.993	0.981	0.781	0.387	0.260	0.162	0.028	0.731	0.675	0.656	0.176	**0.944**	0.823	0.890	0.662	**0.909**	0.991	**0.968**	**0.769**
D13	0.957	0.949	0.969	0.722	0.349	0.199	0.122	0.062	0.677	0.644	0.605	0.154	0.910	0.839	0.857	0.603	0.872	0.982	0.955	0.733
D14	0.953	0.941	0.958	0.721	0.393	0.398	0.268	0.049	0.640	0.665	0.605	0.154	0.927	0.823	0.857	0.626	0.880	**1.000**	0.937	0.715
D15	0.936	0.889	0.900	0.709	0.466	0.434	0.317	0.072	0.699	0.709	0.670	0.191	0.892	0.806	0.817	0.589	0.879	0.982	0.928	0.708
D16	0.936	0.876	0.903	0.680	**0.541**	**0.531**	**0.448**	**0.091**	0.736	0.803	0.748	0.221	0.846	0.887	0.875	0.647	0.908	0.982	0.952	0.743

Table 5: Quantitative evaluation of rotated bounding box detection when the network is trained using only the real dataset.

Dataset	Precision	Recall	mAP@0.5	mAP@0.5:0.95
CLEURIE	0.804	0.783	0.834	0.528
NPAL	0.843	0.811	0.871	0.642

Scaling: The evaluation metrics follow a similar trend to the horizontal box evaluation. Evaluation on the synthetic dataset achieved the best performance when there was no scaling, since the network overfits to the uniform diatom sizes. Scaling helps the network generalize better to the CLEURIE dataset due to the presence of different taxa of diatoms with varying scales. The NPAL dataset consists of diatoms of the same species with uniform sizes, hence the network generalizes best when trained with no scaling.

Overlap: When evaluated on the synthetic dataset, the network performed best when there was no overlap between the diatoms and debris. This is because, when there are no overlaps, there is no ambiguity due to occlusions for the network while learning the rotated bounding box parameters. For the CLEURIE and NPAL datasets, overlap helps the network generalize better.

Diatom-debris ratio: The network performed better when the diatom-debris ratio was in the range [2,5] for the synthetic and real datasets. This is because fewer debris objects means a sparser distribution compared with the diatoms, making them easier to discard.

Table 5 shows the results when the network was trained using only the real dataset. Compared with the performance when the network was trained only on real data, there was an improvement of 17% in precision and 22% in recall for the CLEURIE dataset and an improvement of 8% in precision and 23% in recall for the NPAL dataset, for an mIoU threshold of 0.5 when the network was trained on synthetic data and fine-tuned to real data.

5. Conclusion

In this work, we assessed the use of a synthetic dataset for diatom detection using deep-learning. When the labelled dataset available for training the network is limited, one can use a synthetic dataset for training and fine-tune it to the real dataset. We proposed a method to generate thousands of synthetic datasets and performed a comprehensive analysis on the impact of the synthetic

dataset when training horizontal and rotated object detection networks. Our results show that when the available labeled dataset is limited, fine-tuning from a network trained on synthetic dataset can significantly improve the performance of the network. This approach will thus be of interest for future work, which will focus on the implementation of automatic diatom segmentation and classification networks.

Funding

The PhD scholarship for AV was funded by ANR [ANR-20-THIA-0010] and Région Grand-Est. Additional financial support was provided by CNRS (ZAM LTSER Moselle, PNRIA) and Université de Lorraine (LUE).

Data availability

Image datasets and code used to generate synthetic dataset are available on the institutional repository DOREL (`https://doi.org/10.12763/UADENQ`).

Author contributions

AV: Conceptualization, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing original draft. PFG: Conceptualization, Formal analysis, Methodology, Software, Validation, Writing original draft. CR: Data curation (code), Software, Validation, Reviewing, and editing the final draft. DH: Data Curation (images), Review and editing the final draft CF: Data Curation (images), Review and Editing the final draft. PUP: Funding acquisition, Supervision, Writing-Reviewing and editing the final draft. CP: Project administration, Funding acquisition, Conceptualization, Supervision, Writing- Review and Editing. ML: Project administration, Funding acquisition, Conceptualization, Supervision, Writing-Reviewing and Editing

References

Bey, M.Y., Ector, L., 2013. Atlas des diatomées des cours d'eau de la région rhône-alpes. tome 1. Centriques, Monoraphidées. tome 2. Araphidées, Brachyraphidées. tome 3. Naviculacées: Naviculoidées. tome 4. Naviculacées: Naviculoidées. tome 5. Naviculacées: Cymbelloidées, Gomphonematoidées. tome 6. Bacillariacées, Rhopalodiacées, Surirellacées, Direction Régionale de l'Environnement, de l'Aménagement et du Logement Rhône-Alpes, ISBN:978-2-11-129817-0, p. 1182.

Borges, V.R.P., Hamann, B., Silva, T.G., Vieira, A.A., Oliveira, M.C.F., 2015. A highly accurate level set approach for segmenting green microalgae images, in: 2015 28th SIBGRAPI Conference on Graphics, Patterns and Images, IEEE. pp. 87–94.

COCO, 2019. Coco - evaluate - metrics. http://cocodataset.org/#detection-eval. Accessed: 2020-05-05.

Coste, M., Boutry, S., Tison-Rosebery, J., Delmas, F., 2009. Improvements of the biological diatom index (bdi): Description and efficiency of the new version (bdi-2006). Ecological Indicators 9, 621 – 650. doi:https://doi.org/10.1016/j.ecolind.2008.06.003.

Du Buf, J., Shahbazkia, H., Ciobanu, A., Bayer, M., Droop, S., Head, R., Juggins, S., Fischer, S., Bunke, H., Wilkinson, M., Roerdink, J., Pech Pacheco, J.L., Cristobal, G., 1999. Diatom identification: A double challenge called adiac. Proceedings - International Conference on Image Analysis and Processing, ICIAP 1999 , 734–739doi:10.1109/ICIAP.1999.797682.

Dwibedi, D., Misra, I., Hebert, M., 2017. Cut, paste and learn: Surprisingly easy synthesis for instance detection, in: Proceedings of the IEEE international conference on computer vision, pp. 1301–1310.

Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.Y., Cubuk, E.D., Le, Q.V., Zoph, B., 2021. Simple copy-paste is a strong data augmentation method for instance segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2918–2928.

Girshick, R., 2015. Fast r-cnn, in: Proceedings of the IEEE international conference on computer vision, pp. 1440–1448.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 580–587.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. Advances in neural information processing systems 27.

Jocher, G., Stoken, A., Borovec, J., NanoCode012, ChristopherSTAN, Changyu, L., Laughing, tkianai, Hogan, A., lorenzomammana, yxNONG, AlexWang1900, Diaconu, L., Marc, wanghaoyang0106, ml5ah, Doug, Ingham, F., Frederik, Guilhen, Hatovix, Poznanski, J., Fang, J., , L.Y., changyu98, Wang, M., Gupta, N., Akhtar, O., PetrDvoracek, Rai, P., 2020. ultralytics/yolov5: v3.1 - Bug Fixes and Performance Improvements. doi:10.5281/zenodo.4154370.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

Kingma, D.P., Welling, M., 2013. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 .

Kloster, M., Kauer, G., Beszteri, B., 2014. Sherpa: an image segmentation and outline feature extraction tool for diatoms and other objects. BMC bioinformatics 15, 218.

Kloster, M., Langenkämper, D., Zurowietz, M., Beszteri, B., Nattkemper, T.W., 2020. Deep learning-based diatom taxonomy on virtual slides. Scientific reports 10, 1–13.

Lalanne-Cassou, C., Voisin, J.F., 2013. Atlas des diatomées d'ile de france, Direction Régionale et Interdépartmentale de l'Environnement et de l'Energie d'Île-de-France, p. 734.

Li, L., Zhou, Z., Wang, B., Miao, L., Zong, H., 2020. A novel cnn-based method for accurate ship detection in hr optical remote sensing images via rotated bounding box. IEEE Transactions on Geoscience and Remote Sensing 59, 686–699.

Li, Y., Guo, J., Guo, X., Hu, Z., Tian, Y., 2021. Plankton detection with adversarial learning and a densely connected deep learning model for class imbalanced distribution. Journal of Marine Science and Engineering 9, 636.

Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, pp. 2980–2988.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C., 2016. Ssd: Single shot multibox detector, in: European conference on computer vision, Springer. pp. 21–37.

Lobo, E.A., Heinrich, C.G., Schuch, M., Wetzel, C.E., Ector, L., 2016. Diatoms as bioindicators in rivers, in: River Algae. Springer, pp. 245–271.

O'Neill, D., Xue, B., Zhang, M., 2021. Evolutionary neural architecture search for high-dimensional skip-connection structures on densenet style networks. IEEE Transactions on Evolutionary Computation 25, 1118–1132.

Pedraza, A., Bueno, G., Deniz, O., Cristóbal, G., Blanco, S., Borrego-Ramos, M., 2017. Automated diatom classification (part b): a deep learning approach. Applied Sciences 7, 460.

Pedraza, A., Bueno, G., Deniz, O., Ruiz-Santaquiteria, J., Sanchez, C., Blanco, S., Borrego-Ramos, M., Olenici, A., Cristobal, G., 2018. Lights and pitfalls of convolutional neural networks for diatom identification, in: Optics, Photonics, and Digital Technologies for Imaging Applications V, International Society for Optics and Photonics. p. 106790G.

Peeters, V., Ector, L., 2018. Atlas des diatomées des cours d'eau du territoire bourguignon. volume 2: Monoraphidées, brachyraphidées, Direction Régionale de l'Environnement, de l'Aménagement et du Logement, Bourgogne-Franche-Comté. Dijon, p. 271.

Pérez, P., Gangnet, M., Blake, A., 2003. Poisson image editing. ACM Trans. Graph. 22, 313–318. doi:10.1145/882262.882269.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779–788.

Rojas Camacho, O., Forero, M.G., Menéndez, J.M., 2017. A tuning method for diatom segmentation techniques. Applied Sciences 7, 762.

Sager, C., Janiesch, C., Zschech, P., 2021. A survey of image labelling for computer vision applications. Journal of Business Analytics 4, 91–110.

Salido, J., Sánchez, C., Ruiz-Santaquiteria, J., Cristóbal, G., Blanco, S., Bueno, G., 2020. A low-cost automated digital microscopy platform for automatic identification of diatoms. Applied Sciences 10, 6033.

Shi, Z., Wang, K., Cao, L., Ren, Y., Han, Y., Ma, S., 2019. Study on holographic image recognition technology of zooplankton. International Conference on Computer Intelligent Systems and Network Remote Control (CISNRC 2019) , 580–594.

Verikas, A., Gelzinis, A., Bacauskiene, M., Olenina, I., Olenin, S., Vaiciukynas, E., 2012. Phase congruency-based detection of circular objects applied to analysis of phytoplankton images. Pattern Recognition 45, 1659–1670.

Xue, Y., Qin, J., 2022. Partial connection based on channel attention for differentiable neural architecture search. IEEE Transactions on Industrial Informatics .

Xue, Y., Wang, Y., Liang, J., Slowik, A., 2021. A self-adaptive mutation neural architecture search algorithm based on blocks. IEEE Computational Intelligence Magazine 16, 67–78.

Zhong, B., Ao, K., 2020. Single-stage rotation-decoupled detector for oriented object. Remote Sensing 12, 3262.