Implications of single-stage deep learning networks in real-time zooplankton identification

Sadaf Ansari[1,*], Dattesh V. Desai[2], Aya Saad[3] and Annette Stahl[4]

[1]Marine Instrumentation Division (Computer Vision and AI), and
[2]Biological Oceanography Division, CSIR-National Institute of Oceanography, Dona Paula, Goa 403 004, India
[3]Aquaculture Department, SINTEF Ocean AS, Trondheim, Norway
[4]Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU), Trondheim, Norway

Zooplankton are key ecological components of the marine food web. Currently, laboratory-based methods of zooplankton identification are manual, time-consuming, prone to human error and require expert taxonomists. Therefore, alternative methods are needed. In this study, we describe, implement and compare the performance of six state-of-the-art single-stage deep learning models for automated zooplankton identification. The highest prediction accuracy achieved is 99.50%. The fastest detection speed is 285 images per second, making the models suitable for real-time zooplankton classification. We validate the predictions of the generated models on unseen images. The results demonstrate the capabilities of the latest deep learning models in zooplankton identification.

Keywords: Artificial intelligence, deep learning networks, imaging, marine biology, zooplankton.

ZOOPLANKTON are tiny marine organisms that serve as food for almost all oceanic creatures, particularly consumers in the food chain. They form the link between primary producers (phytoplankton) and higher consumers (such as fish) in the food web. Detecting and classifying zooplankton is crucial for analysing their diversity and abundance, which provides insights into food-web dynamics, fishery potential and environmental health through environmental impact assessment (EIA) studies. As the composition of zooplankton groups is sensitive to environmental changes, their distribution and abundance serve as proxies for the ecological quality of aquatic environments and climate-change phenomena such as global warming[1]. Therefore, it is vital to study and monitor the spatial distribution, temporal variability and abundance of zooplankton to understand community structure, its relation to climate change, food-web dynamics and fishery potential. This information is also crucial for monitoring programmes involving EIA studies of the marine and freshwater ecosystems.

*For correspondence. (e-mail: ansari@nio.org)

The systematics of zooplankton is complex as it includes numerous species and also lacks diagnostic characters. This is considered a barrier to understanding the patterns of zooplankton biodiversity over local to global scales[2]. The zooplankton assemblage offers several challenges in the identification and analysis of species-level diversity[3–5]. The morphological features of zooplankton are highly complex and require taxonomic expertise for accurate identification. The taxonomic identification of zooplankton groups is a manual process based on their morphological observations. It involves microscopy and is labour-intensive and time-consuming. As we go down the hierarchy (from phylum to species), taxonomic details for identification become finer, and it is increasingly difficult and time-consuming to count and classify zooplankton manually. To overcome these shortcomings, there is an urgent need to develop methods for automatic detection and classification strategies that also have high accuracy and speed.

Over the past two decades, there has been a significant increase in the development of imaging equipment and devices for recording plankton images, including the video plankton recorder[6], FlowCytobot[7], FlowCam[8], ZooScan[9] and a mobile robotic explorer[10]. The growing use of imaging devices has generated a large amount of image data, leading to the development of image-processing techniques to analyse the collected images of marine organisms[9,11–13]. Convolutional neural networks (CNNs) have gained popularity for identifying marine organisms such as fish and plankton[14–20]. Many CNN-based object-detection frameworks have been proposed, grouped into two categories: one-stage and two-stage detectors. One-stage detectors perform detection and localization in a single step, making them faster. The present study focuses on zooplankton, a subcategory of plankton that belongs to the animal kingdom. A popular CNN for zooplankton classification is ZooplanktonNet, which consists of 11 layers and achieves over 93% accuracy[21].

The primary objective of this study is to describe, evaluate and compare six dominant, state-of-the-art, single-stage deep learning networks for multi-class zooplankton classification

and localization. The study also presents speed and accuracy trade-offs of the models, as well as a comparison of model size.

Materials and methods

We provide a comprehensive overview of the six dominant, state-of-the-art, single-stage deep learning networks for automated zooplankton identification. It includes detailed information about the dataset, network architecture, backbone networks, implementation, training and performance evaluation. Figure 1 illustrates the methodology adopted in this study.

Dataset

In this study, a custom dataset of zooplankton images was developed by consolidating images from the WHOI-Plankton[22,23] and Kaggle[24,25] datasets. The WHOI-Plankton dataset consists of plankton images collected by the Woods Hole Oceanographic Institution from the Martha's Vineyard Coastal Observatory in Massachusetts, USA, while the Kaggle zooplankton dataset was obtained from images collected in the Straits of Florida, USA, during May–June 2014. For this study, we selected the five most significant zooplankton categories based on their relevance to marine ecosystems in Indian aquatic habitats. The zooplankton groups are Copepoda, Chaetognatha, Siphonophora, fish larva and Echinodermata (Figure 2).

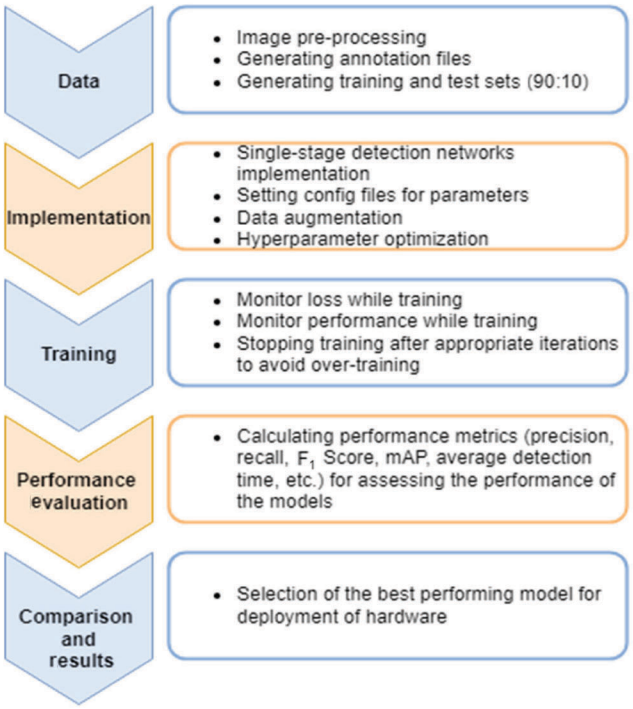

Figure 1. Methodology flow.

In this study, each zooplankton group is considered a distinct 'class' for training the neural network. To increase the size of the dataset, we employed data augmentation techniques such as rotation, horizontal flip and vertical flip, resulting in a total of 1629 original images. The dataset was divided into a training set and a validation set in the ratio of 90 : 10. To assess the model's generalization on unseen images, we utilized a test set of 221 images.

Methods

Image preprocessing: The original images in the dataset have varying sizes and a white background, and in some cases, the zooplankton (regions of interest (RoI)) touches the edge of the image. To address this, we added a white background and resized the images to 224 × 224 pixels. Figure 3 shows the original image after pre-processing. Also, pre-processing is typically not required for CNNs, as the network extracts features from the training data during the training process using a multi-layer architecture. Finally, we annotated each image according to the zooplankton group present in it.

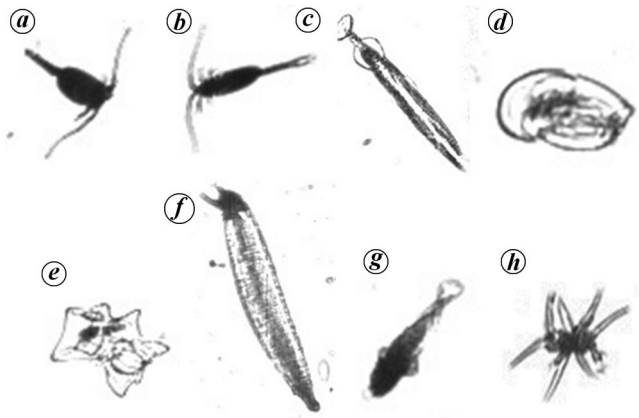

Figure 2. Zooplankton groups: (***a***) and (***b***) Copepoda, (***c***) Chaetognatha, (***d***) and (***e***) Siphonophora, (***f***) and (***g***) fish larva, (***h***) Echinodermata (source: ref. 25).

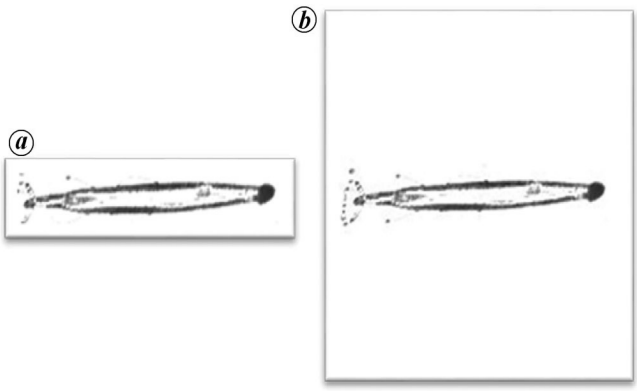

Figure 3. ***a***, Original image (ref. 25). ***b***, Image after pre-processing.

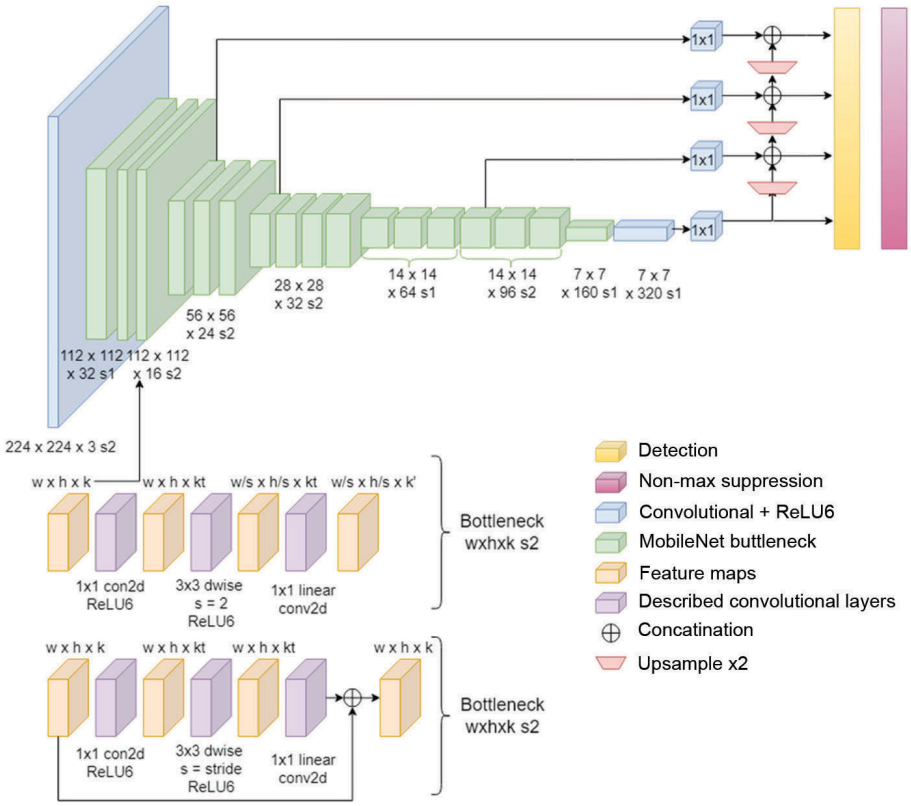

Figure 4. Single shot multibox detector with MobileNet architecture.

Single-stage detection and feature extraction: Two-stage detectors for object detection split the process into two steps. First, they propose RoI, and in the second step, the proposals are classified, and their localization is adjusted. However, this two-step approach slows down the detection process, making it unsuitable for real-time applications that require high-speed and classification accuracy, such as zooplankton analysis. In contrast, single-stage detection systems perform classification and localization in a single step, making it faster and more efficient for real-time applications. The present study compares the performance of six state-of-the-art, single-stage detectors for zooplankton classification, including the single shot multibox detector (SSD)[26], RetinaNet[27] and the you only look once (YOLO) series of algorithms[28–31].

We utilized a Keras-based MobileNet V2 feature pyramid network (FPN) feature extractor for SSD to extract feature maps from zooplankton images (Figure 4). This stack of convolutional networks is well-suited for mobile devices due to its small size and low latency, as demonstrated by Howard *et al.*[32]. Feature maps contain valuable semantic information that is localized in various regions of the zooplankton image, and they serve as a foundation for generating box predictions for zooplankton using the detection head of SSD. Non-max suppression is used by SSD to eliminate redundant predictions. It verifies the confidence score for each box prediction using IoU (Intersection over Union) and selects the box having the highest overlap with the ground-truth box.

RetinaNet, introduced by Lin *et al.*[27], is a recent addition to the one-stage detector family. This architecture shares similarities with previous object detectors, such as the use of 'feature pyramids' from SSD[26] and FPN[33] and 'anchors' from RPN[34]. It uses a ResNet architecture as the backbone with an FPN on top. Each FPN is connected to two subnetworks, one for classification and one for regression. RetinaNet addresses the class imbalance faced during training in one-stage detectors by introducing the concept of focal loss to the classification subnet. This loss bridges the accuracy gap between single-stage and state-of-the-art, two-stage detectors while maintaining comparable detection speeds to one-stage detectors. Figure 5 shows the RetinaNet architecture.

The YOLO series of algorithms are known for their high detection accuracy and fast object-detection capabilities. YOLOv4, for instance, utilizes a single neural network that partitions the zooplankton image into grids[35]. However, to ensure that the algorithm is equally efficient when run on embedded devices, a variation of the YOLO architecture was introduced called Tiny-YOLO[28]. For the present study, we employed the DarkNet-53 feature extractor for Tiny-YOLOv4. We compared the performances of Tiny-YOLOv4, YOLOv5 (ref. 29), YOLOv6 (ref. 30), and YOLOv7 (ref. 31). Figure 6 shows the architecture of YOLOv5.

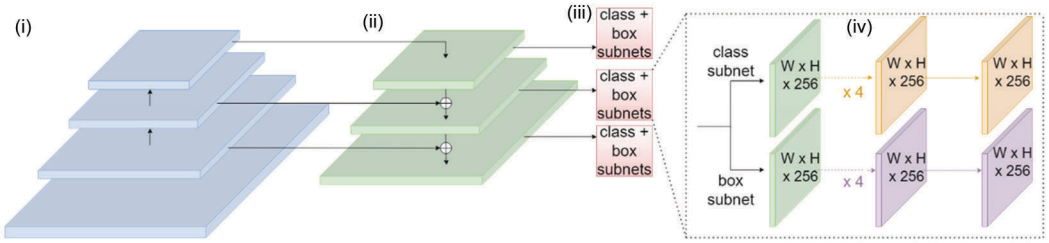

Figure 5. RetinaNet architecture. (i) ResNet, (ii) Feature pyramid net, (iii) Class subnet (top) and (iv) Box subnet (bottom).

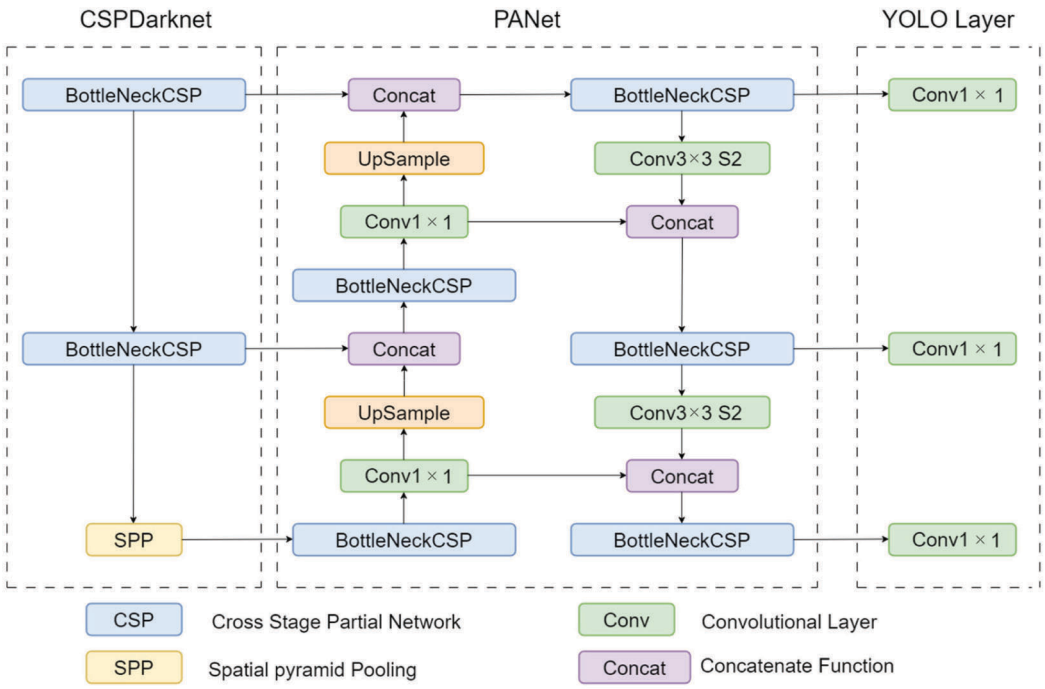

Figure 6. YOLOv5 architecture.

Implementation and training procedure: The TensorFlow2 deep learning framework was utilized to conduct experiments on the dataset. The training was carried out on the NVIDIA RTX A6000 GPU with 64 GB RAM, where the backbone networks were pre-trained on the COCO dataset[36]. To avoid lengthy training times, transfer learning was employed by reusing the weights from the COCO dataset and fine-tuning them while training on our custom dataset. To ensure efficient training and accurate predictions, we adjusted the training parameters, and the model was trained on the training set, while the validation set was used for evaluation during training. Figure 7 shows the progression of loss functions during the training of SSD MobileNet, RetinaNet ResNet101, Tiny-YOLOv4, YOLOv5, YOLOv6 and YOLOv7 models. The loss function value decreases with training time and eventually stabilizes. Upon completion of the training process, the test set was used to assess the generalization ability of the models.

For the training process, we set an initial learning rate of 0.001 and a burn-in period of 1000 while utilizing a batch size of 64 with subdivisions of 16. Our chosen hyperparameters include a momentum of 0.9 and a weight decay of 0.0005.

Performance evaluation: For the evaluation of zooplankton classification and localization models, we employed various performance indicators, including precision, recall, F_1 score, mean average precision (mAP), and average detection time (sec). The precision and recall are defined as follows

$$\text{Precision} = \text{True positives}/(\text{True positives} + \text{false positives}), \quad (1)$$

$$\text{Recall} = \text{True positives}/(\text{True positives} + \text{false negatives}). \quad (2)$$

The 'true positives' are positive outcomes that the model predicts correctly, 'false positives' are positive outcomes

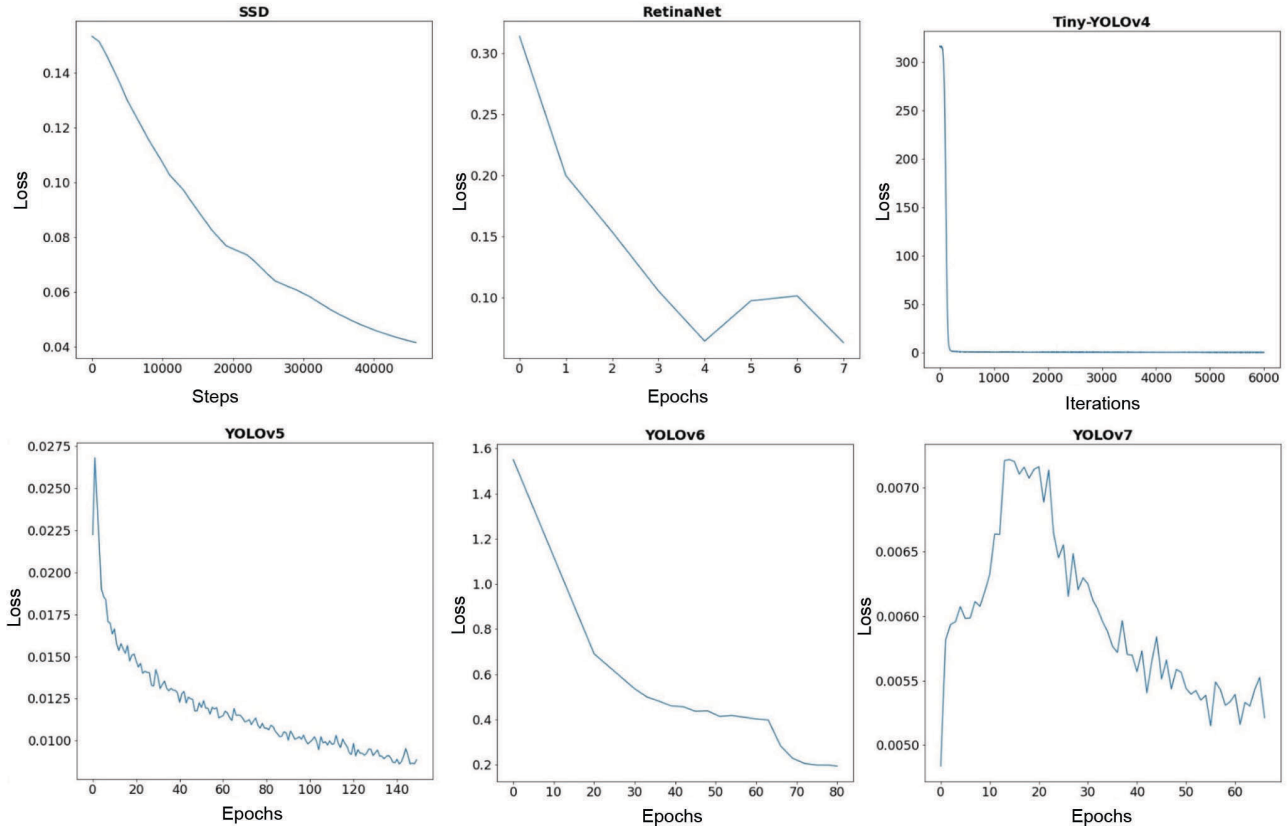

Figure 7. Loss function graphs of deep learning models.

that it predicts incorrectly, and 'false negatives' are negative outcomes that the model predicts incorrectly.

The F_1 score is the weighted harmonic average of precision and recall values.

$$F_1 = \frac{2(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}. \quad (3)$$

mAP is the average precision of all zooplankton classes.

$$\text{mAP} = \frac{1}{c}\sum_{i=1}^{c} \text{AP}_i, \quad (4)$$

where AP_i is the average precision of class i and c is the number of classes.

In addition, detection speed is a crucial metric for assessing model performance, measured in frames per second (FPS). It represents the number of images processed per second, and a higher detection speed indicates better real-time performance. Typically, a detection speed of 30 FPS or greater is considered to achieve real-time detection.

Results and discussion

In this study, laboratory-based manual analysis of zooplankton samples for taxonomic identification was automated using six state-of-the-art, dominant categories of single-stage object detectors: SSD, RetinaNet and the YOLO series of algorithms (Tiny-YOLOv4, YOLOv5, YOLOv6 and YOLOv7) for high speed and accuracy. Lightweight backbone networks have been used in all the object detectors to make them suitable for mobile applications. The performance of the six deep learning models was evaluated using popular performance evaluation metrics like precision, recall, F_1 score, mAP and FPS to identify the best-performing single-stage network for taxonomic identification of zooplankton for use in real-time applications (Figure 8). Table 1 presents the performance metrics of different object detection models, including SSD, RetinaNet, Tiny-YOLOv4, YOLOv5, YOLOv6 and YOLOv7. Precision, recall, F_1 score and accuracy were the evaluation metrics used to assess the deep learning models. The YOLOv5 model achieved the highest F_1 score of 0.997 and the highest accuracy of 99.50%, indicating that it is the most accurate and precise object detection model compared to the other models. Among the YOLO family of algorithms, Tiny-YOLOv4 and YOLOv7 achieved slightly lower scores than YOLOv5 with an F_1 score of 0.93 and 0.924, respectively, and mAP of 95.03% and 95.5% respectively. YOLOv6 showed the lowest performance. The SSD model achieved a precision of 0.719, a recall of 0.775 and an F_1 score of 0.746, with an accuracy of 92.50%. The RetinaNet model achieved a precision of

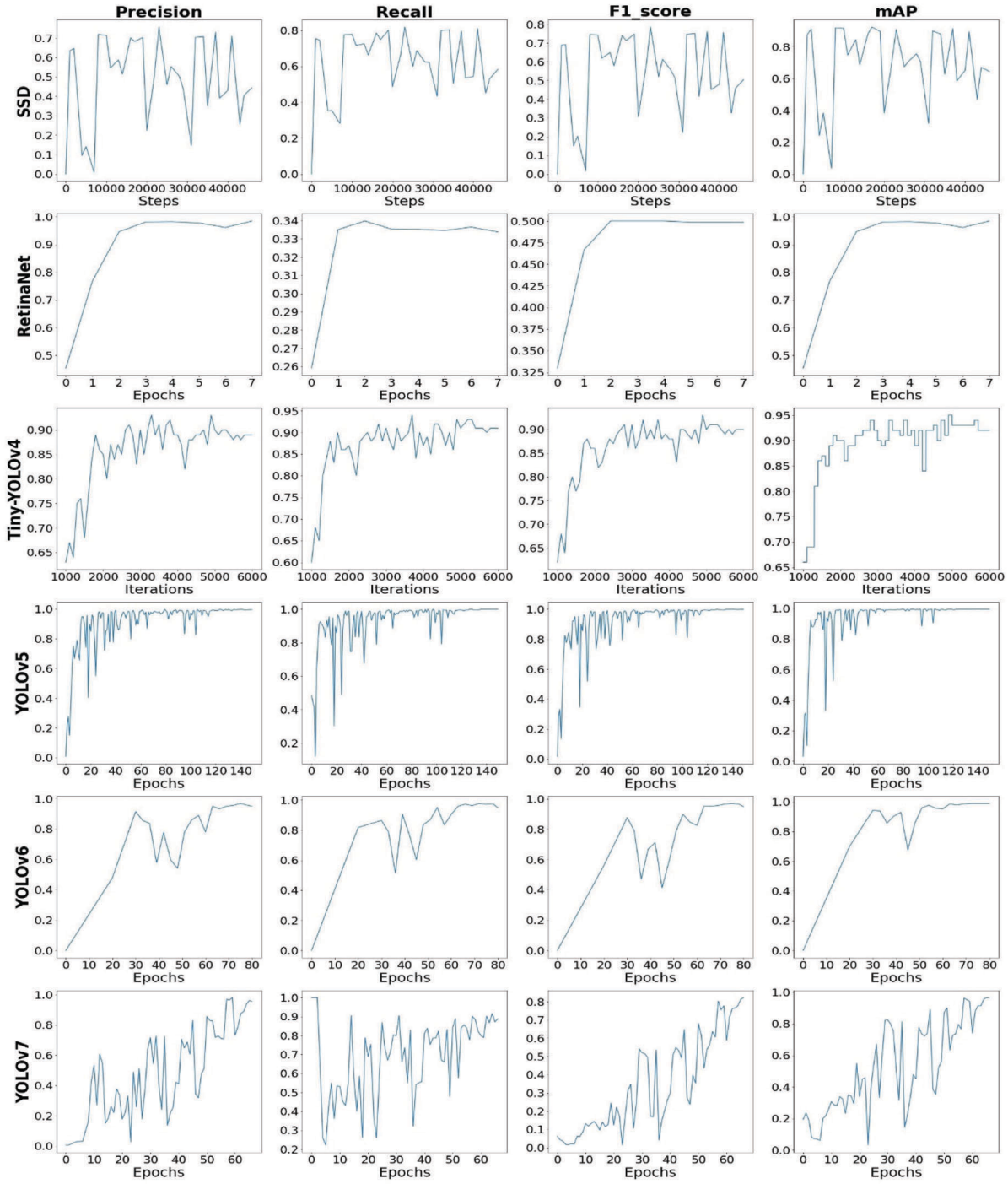

Figure 8. Performance graphs of deep learning models.

0.982, a recall of 0.335 and an F_1 score of 0.5, with an accuracy of 98.22%. Although mAP of the SSD and RetinaNet models was high, their F_1 scores were low.

Table 2 shows the performance of the deep learning-based zooplankton classification models in terms of their FPS and size (megabytes, MB). The YOLOv5 model performed exceptionally well, achieving an impressive FPS of 120 and a relatively small model size of 14.3 MB. The YOLOv6 and YOLOv7 models further improved FPS, achieving 215 and 285 respectively, but had larger model sizes of 40.6 and 70.8 MB respectively. The SSD model achieved an FPS of 9 with a model size of 6.5 MB. The RetinaNet model achieved an FPS of 8 with a significantly larger model size of 437 MB. Figure 9 reveals that each model exhibits trade-offs between accuracy and speed. Overall, these results indicate that the YOLOv5 model

outperforms the other models in terms of F_1 score, mAP, FPS and model size, making it a promising option for real-time applications with limited computing resources.

Table 1. Performance metrics for deep learning models

Model	Precision	Recall	F_1 score	mAP (%)
SSD	0.719	0.775	0.746	92.50
RetinaNet	0.982	0.335	0.500	98.22
Tiny-YOLOv4	0.930	0.930	0.930	95.03
YOLOv5	0.996	1.000	0.997	99.50
YOLOv6	0.987	0.889	0.972	97.17
YOLOv7	0.916	0.934	0.924	95.50

Table 2. Speed and size of deep learning models

Model	Frames per second	Model size (MB)
SSD	9	6.5
RetinaNet	8	437
Tiny-YOLOv4	2	260
YOLOv5	120	14.3
YOLOv6	215	40.6
YOLOv7	285	70.8

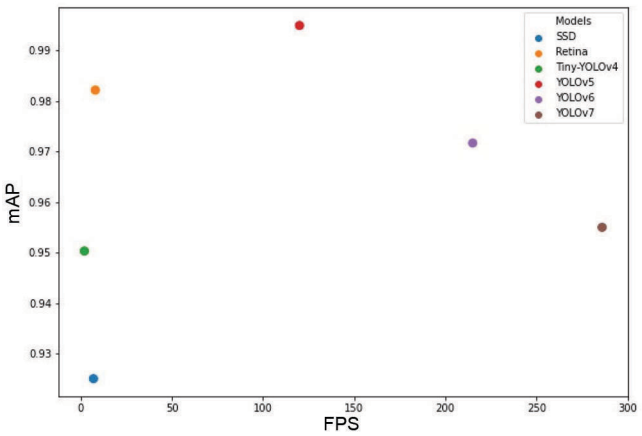

Figure 9. Speed and accuracy comparison of deep learning models.

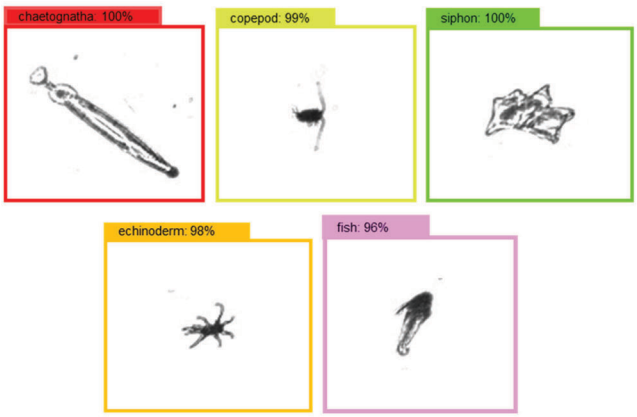

Figure 10. Examples of deep learning model predictions.

Figure 10 shows the predictions of the YOLOv5 model on zooplankton test images.

Conclusion

The primary objective of this study was to evaluate the potential usefulness, practical implementation and comparative performance of six state-of-the-art, single-stage deep learning models for identifying zooplankton, with a specific emphasis on their suitability for real-time mobile applications.

We implemented six deep learning models, namely SSD, RetinaNet, Tiny-YOLOv4, YOLOv5, YOLOv6 and YOLOv7, to detect and classify zooplankton from different taxonomic groups. The performance of these models was evaluated using various indicators, such as precision, recall, F_1 score, mAP, detection speed and model size. SSD MobileNet and RetinaNet achieved an mAP of 92.5% and 98.22% respectively. However, their detection speed of 9 and 8 FPS respectively, and large model size are inadequate for real-time mobile applications. The YOLO series of algorithms exhibited the highest F_1 score, accuracy and detection speed, while featuring some of the lightest models, making them suitable for deployment on hardware for real-time mobile applications. YOLOv7 has a mAP of 95.50% and an F_1 score of 0.927, slightly lower than that of YOLOv5, but it had the highest detection speed of 285 FPS. Overall, YOLOv5 outperformed all the other models with an F_1 score of 0.997, mAP of 99.50% and a high detection speed of 120 images per second, making it ideal for real-time applications. Additionally, with a lightweight model size of 14.3 MB, YOLOv5 could meet the requirements for deployment on embedded platforms. In a forthcoming study, our collected data will be utilized. We intend to optimize the model to ensure its suitability for deployment on embedded platforms specifically designed for field use.

1. Richardson, A. J., In hot water: zooplankton and climate change. *ICES J. Mar. Sci.*, 2008, **65**(3), 279–295.
2. Bucklini, A., Lindeque, P. K., Rodgrigues-Ezpeleta, Albaina, A. and Lehtiniemi, M., Metabarcoding of marine zooplankton: prospects, progress and pitfalls. *J. Plankton Res.*, 2016, **38**(3), 393–400.
3. Knowlton, N., Molecular genetic analyses of species boundaries in the sea. *Hydrobiology*, 2000, **420**, 73–90.
4. Lindsay, D. J. *et al.*, The perils of online biogeographic databases: a case study with the 'monospecific' genus *Aegina* (Cnidaria, Hydrozoa, Narcomedusae). *Mar. Biol. Res.*, 2017, **13**, 494–512.
5. Snelgrove, P. *et al.*, Global patterns in marine biodiversity. In *United Nations World Ocean Assessment I*, Cambridge University Press, UK, 2017, pp. 501–524.
6. Davis, C. S., Gallager, S. M., Berman, M. S., Haury, L. R. and Strickler, J. R., The video plankton recorder (VPR): design and initial results. *Arch. Hydrobiol. Beih. Ergebn. Limnol.*, 1992, **36**, 67–81.
7. Olson, R. J. and Sosik, H. M., A submersible imaging-in-flow instrument to analyze nano-and microplankton: imaging FlowCytobot. *Limnol. Oceanogr. Methods*, 2007, **5**, 195–203.
8. Sieracki, C. K., Sieracki, M. E. and Yentsch, C. S., An imaging-in-flow system for automated analysis of marine microplankton. *Mar. Ecol. Prog. Ser.*, 1998, **168**, 285–296.

9. Grosjean, P., Picheral, M., Warembourg, C. and Gorsky, G., Enumeration, measurement, and identification of net zooplankton samples using the ZOOSCAN digital imaging system. *ICES J. Mar. Sci*., 2004, **61**, 518–525.
10. Saad, A. *et al.*, Advancing ocean observation with an AI-driven mobile robotic explorer. *Oceanography*, 2021, **33**(3), 50–59.
11. Jeffries, H. P., Berman, M. S., Poularikas, A. D., Katsinis, C., Melas, I., Sherman, K. and Bivins, L., Automated sizing, counting and identification of zooplankton by pattern recognition. *Mar. Biol.*, 1984, **78**, 329–334.
12. Tang, X., Stewart, W. K., Huang, H., Gallager, S. M., Davis, C. S., Vincent, L. and Marra, M., Automatic plankton image recognition. *Artif. Intellig. Rev*., 1998, **12**, 177–199.
13. Gorsky, G. *et al*., Digital zooplankton image analysis using the ZooScan integrated system. *J. Plankton Res.*, 2010, **32**, 285–303.
14. Py, O., Hong, H. and Zhongzhi, S., Plankton classification with deep convolutional neural networks. In Proceedings of the IEEE Information Technology, Networking, Electronic and Automation Control Conference, Chongqing, China, 2016, pp. 132–136.
15. Sung, M., Yu, S. C. and Girdhar, Y., Vision based real-time fish detection using convolutional neural network. In Proceedings of IEEE OCEANS 2017, Aberdeen, United Kingdom, 2017.
16. Cheng, K., Cheng, X., Wang, Y., Bi, H. and Benfield, M. C., Enhanced convolutional neural network for plankton identification and enumeration. *PLoS ONE*, 2019, **14**(7), e0219570.
17. Ansari, S., Saad, A., Stahl, A. and Rajachandran, M., Vision-based real-time zooplankton detection and classification using faster R-CNN. In AGU ASLO Ocean Sciences Meeting, San Diego, California, USA, 2020, doi:10.1002/essoar.10502404.1.
18. Bergum, S., Saad, A. and Stahl, A., Automatic *in situ* instance and semantic segmentation of planktonic organisms using Mask R-CNN. In Proceedings of Global Oceans 2020: Singapore – US Gulf Coast, 2020, pp. 1–8.
19. Cheng, X., Ren, Y., Cheng, K., Cao, J. and Hao, Q., Method for training convolutional neural networks for *in situ* plankton image recognition and classification based on the mechanisms of the human eye. *Sensors*, 2020, **20**(9), 2592.
20. Saad, A., Bergrum, S. and Stahl, A., An instance segmentation framework for *in situ* plankton taxa assessment. In Proceedings of Thirteenth International Conference on Machine Vision, SPIE, 2021, vol. 11605, pp. 294–303.
21. Dai, J., Wang, R., Zheng, H., Ji, G. and Qiao, X., ZooplanktonNet: deep convolutional network for zooplankton classification. In Proceedings of IEEE OCEANS 2016, Shanghai, China, 2016.
22. Sosik, H. M. and Olson, R. J., Automated taxonomic classification of phytoplankton sampled with imaging-in-flow cytometry. *Limnol. Oceanogr. Meth*., 2007, **5**, 204–216.
23. Sosik, H. M., Peacock, E. E. and Brownlee, E. F., Annotated Plankton Images – Data Set for Developing and Evaluating Classification Methods. 2015; https://hdl.handle.net/10.1575/1912/7341, doi:10.1575/1912/7341.
24. Cowen, R. K. and Guigand, C. M., *In situ* ichthyoplankton imaging system (ISIIS): system design and preliminary results. *Limnol. Oceanogr. Meth.*, 2008, **6**, 126–132.
25. Cowen, R. K., Sponaugle, S., Robinson, K. and Luo, J., Planktonset 1.0: plankton imagery data collected from F.G. Walton Smith in Straits of Florida from 2014-06-03 to 2014-06-06 and used in the 2015 National Data Science Bowl (NCEI Accession 0127422), Oregon State University, Hatfield Marine Science Center, USA, 2015.
26. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y. and Berg, A. C., SSD: Single Shot MultiBox Detector. In Proceedings of European Conference of Computer Vision, Amsterdam, The Netherlands, 2016, pp. 21–37.
27. Lin, T. Y. *et al.*, Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 2017, pp. 2980–2988.
28. Jiang, Z., Zhao, L., Li, S. and Jia, Y., Real-time object detection method based on improved YOLOv4-tiny, 2020, arXiv preprint arXiv:2011.04244.
29. Jung, Hyun-Ki and Choi, Gi-Sang, Improved YOLOv5: efficient object detection using drone images under various conditions. *Appl. Sci.*, 2022, **12**(14), 7255.
30. Li, C. *et al.*, YOLOv6: a single-stage object detection framework for industrial applications, 2022, arXiv preprint arXiv:2209.02976.
31. Wang, Chien-Yao, Bochkovskiy, A. and Liao, Hong-Yuan, YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, 2022, arXiv preprint arXiv:2207.02696.
32. Howard, A. G. *et al.*, MobileNets: efficient convolutional neural networks for mobile vision applications, 2017, arXiv preprint arXiv:1704.04861.
33. Lin, T. Y., Dollar, P., Girshick, R., He, K., Hariharan, B. and Belongie, S., Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, Hawaii, USA, 2017, pp. 2117–2125.
34. Ren, S., He, K., Girshick, R. and Sun, J., Faster R-CNN: towards real-time object detection with region proposal networks. *Adv. Neural Inf. Proces. Syst.*, 2015, **28**, 91–99.
35. Redmon, J., Divvala, S., Girshick, R. and Farhadi, A., You only look once: unified, real-time object detection. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016, pp. 779–788.
36. Lin, T Y. *et al.*, Microsoft COCO: common objects in context. In Proceedings of the European Conference of Computer Vision, Zurich, Switzerland, 2014, pp. 740–755.

ACKNOWLEDGEMENTS. We thank CSIR-National Institute of Oceanography (CSIR-NIO), Goa, India and the Norwegian University of Science and Technology, Norway, for continuous support and encouragement throughout this study. This is CSIR-NIO contribution 7088.

Received 8 February 2023; re-revised accepted 20 September 2023

doi: 10.18520/cs/v125/i11/1259-1266