YOLOv7 Model for Small Object Handling in Maritime Images

Miran Pobar
University of Rijeka
Faculty of Informatics and Digital Technologies
Radmile Matejčić 2, 51000 Rijeka, Croatia
mpobar@uniri.hr

Abstract. *Around the world many maritime surveillance cameras exist whose functionality could be expanded with computer vision-based object detection in order to monitor traffic and provide automated statistics or to increase safety. To this aim, we train two versions of the Yolov7 object detection model on a suitable custom dataset with four object categories and evaluate their detection performance. In order to handle small objects such as buoys or objects that appear visually small in the frame such as distant boats, we examine two different configurations of input to the model.*

Keywords. computer vision, neural networks, small objects, ship detection

1 Introduction

Object detection in images is a fundamental step in many realistic applications of computer vision, e.g. surveillance, autonomous driving, industrial automation etc. Although most recent research in object detection uses large datasets of common objects such as MS COCO (Lin et al. 2014) for both training and performance benchmarking, specialized tasks require training on task-specific datasets that include object categories of interest and in realistic application context. In this paper, the focus is on detection of maritime vessels and small objects in images from shore-mounted surveillance and panoramic cameras, with possible applications for increasing the safety by detecting objects that may be difficult to detect via other means such as radar, or monitoring traffic for vessels that are not equipped with the automatic identification system (AIS). Maritime surveillance cameras already exist in many places, and a whose functionality could be better utilized by automated object detection.

For object detection, we use transfer learning from an established convolutional neural network-based model to adapt the original model to the domain of interest.

To train and evaluate the model, an original dataset with annotated images of maritime scenes is used. The size of the dataset is quite small but is designed to cover a diverse set of locations and maritime scenes (e.g. large commercial port, marina, canal, small fishing boat port, shore near a tourist destination) and to enable transfer learning from models pretrained on large generic datasets.

Although great progress has been made in recent years with neural network-based methods for detecting objects in images, detection of small or tiny objects has remained challenging (Tong & Wu, 2022), especially on busy backgrounds or when objects, in addition to being small, have similar colour as the background.

Here, to deal with smallest objects such as buoys that may appear only a few pixels large in the video frame, a simple input slicing approach is evaluated. If the input image is larger than the size that the model expects, the input image is sliced into rectangular segments of the expected size and detection is performed on each slice. In this way, rescaling is avoided, which could obliterate smallest objects and make detection impossible.

The contributions of this paper include the detection performance comparison of two YOLOv7 model variants for the task of object detection on a novel maritime image dataset, and the comparison of how two image input strategies, namely rescaling or slicing affect the detection performance, especially of small objects.

The rest of the paper is organized as follows: in Section 2, related work is introduced, Section 3 describes the proposed dataset and models for maritime object detection with details of experimental setup. Results are presented and discussed in Section 4. Finally, a conclusion and suggestion for further work are given in Section 5.

2 Related Work

The majority of recent object detection methods such as SSD (Liu et al., 2016), Yolo family of detectors

(Redmon et al., 2016, Redmon & Farhadi, 2018, Bochkovskiy, Wang & Liao, 2020, Bochkovskiy & Liao, 2022) and EfficientDet (Tan, Pang & Le, 2020) are focused on detection of objects in single images and are based on convolutional neural networks (CNNs).

While universal object detectors trained on general object datasets perform well for many tasks, new application domains often require modifications and specialized datasets to adapt the models to the new tasks, such as detection in thermal imaging (Krišto, Ivašić-Kos & Pobar, 2020), or when dealing with small objects, such as detecting bees (Stojnić et al., 2021) or persons (Sambolek & Ivasic-Kos, 2021) in UAV videos.

Detection of small objects is an active topic, and some research is directed at more task-specific methods, for example (Stojnić et al., 2021), while some aim for more general solutions with modification of existing architectures such as YOLO-Z (Benjumea et al. 2023). Although in many practical applications of object detection, the true input source are not individual independent images but frames of video, relatively few methods focus on exploiting the information from multiple frames to improve object detection. For example, in (Broad, Jones & Lee, 2018) an architecture of single-image CNN-based object detector is modified with a recurrent layer so that data from multiple frames is fused on the level of extracted features and further processed in the network.

In this paper a simple input image slicing strategy is explored to avoid reducing the image size and preserve small objects in the input images, similar to (Unel, Ozkalayci & Cigla, 2019)

Datasets for detection of small object in the sea such as SeaDroneSee (Varga et al., 2022) have appeared recently. The SeaDroneSee is tailored for the search-and-rescue scenario and thus focuses on the detection of persons in the water. Reflecting the application, the images were collected using unmanned aerial vehicles with video quality being one of prime concerns.

The SPSCD dataset proposed in (Petković et al. 2023) contains images from a stationary camera in a port and is focused on detection of several classes of ships that may enter the port. It covers different times of day and weather conditions but doesn't specifically address small objects and is limited to one port.

The dataset proposed in this paper targets similar use as the SPSCD dataset but also focuses on small objects in the maritime domain, and aims to cover more variety of locations, however on a smaller scale and with fewer object classes. Since the source videos are broadcast streams from coastal cameras the video quality is highly compressed, which makes detection of small objects even more challenging.

3 Methods and Data

In this experiment, two different Yolov7-based models (Wang, Bochkovskiy & Liao, 2023) for detection of maritime objects in images were trained and evaluated on a custom dataset. Yolov7 is an iteration of the Yolo (Redmon et al., 2016) family of convolutional neural network-based object detectors that predict the bounding boxes of objects in images and their corresponding class labels in a single pass through the network. On the MS COCO object detection dataset, it achieves excellent results in terms of both speed and accuracy (Wang, Bochkovskiy & Liao, 2023). It has also recently been applied to the problem of detection of maritime objects (Zhu et al., 2023), and the authors note the mean average precision (MAP) of 75.66% at 50% intersection-over-union (IoU) threshold on a dataset for maritime search-and-rescue task (Gasienica-Jozkowy, Knapik & Cyganek, 2021). The authors proposed a modified YOLOv7 architecture, and achieved an increase small object detection mAP score of 4.2%, and an improvement of detection accuracy from 0.578 to 0.754 for the *boat* class specifically and from 0.615 to 0.643 for the buoy class in one network variant.

3.1 Dataset

The object detection dataset was prepared from 20 video streams available on the internet streaming from stationary cameras from various ports, canals, marinas, and other points of interest (Fig. 1), covering a diverse range of scenes and weather conditions. Broadly grouped, 10 streams were from small ports or marinas, where usually many smaller boats and buoys appear simultaneously in frame, with lots of occlusion due to boats being close together. Five streams were of general vistas of tourist interest, where usually fewer boats appear simultaneously but most move, and five streams were from large ports or shipping canals where normally large commercial ships appear. Most videos were streamed in 1920x1080 resolution, and a minority in 1280x720, with varying degrees of quality and compression strength. Some of the cameras were completely stationary, while some used panning or zooming motion to cover a larger area. The videos were gathered from April to May 2022.

From the videos, individual frames were extracted at time points where the scene subjectively appeared significantly changed, and saved in .jpg format. In some cases, e.g. when the camera pans or zooms, the interval between extracted images may be just a few seconds, while in some cases, e.g. in calm weather and in a port that is not busy, the interval may be several hours or days.

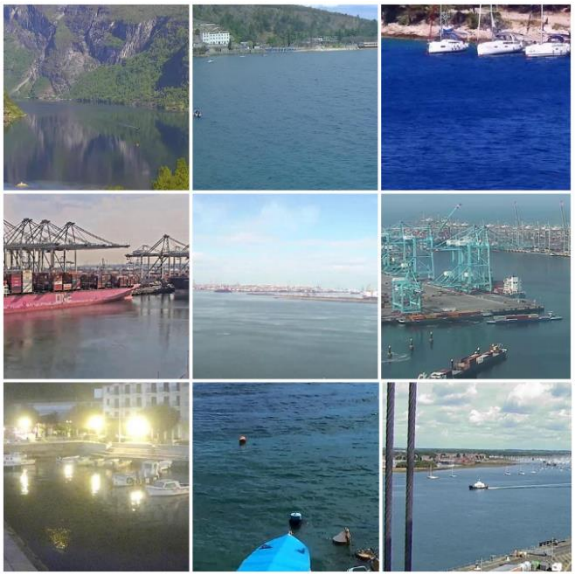


Figure 1. Examples of frames in source videos.

The frames were then manually annotated with bounding boxes and class labels in four classes: *Ship*, *Boat*, *Buoy*, *Swimmer* and *Unknown* for unidentified objects in the sea. For labeling purposes, boats were roughly defined as sailboats, small fishing boats, small passenger boats, small and medium yachts while ships cargo vessels, ferries, medium and large passenger ships, large yachts etc. were labeled as ships. Since there were very few instances of the *swimmer* class, in the final dataset this class was merged with the *unknown* class. The final dataset contains 135 images with 2755 object instances (Table 1). Regarding the object sizes, boat class was ususally the largest, due to position of several cameras that were in marinas where boats passed close by. Ship class is usually visually smaller in the image because large ships were usuall much farther away from the camera. Buoy and unknown classes were the smallest regardless of the camera position, with buoy instances most often taking up under 0.01% of total image area, sometimes taking up only a few pixels.

Table 1. Number of object instances per class and median bounding box area as % of image area per class.

Class	Instances	Median area (% of image)
Boat	1537	0.93
Ship	486	0.13
Buoy	672	0.01
Unknown	60	0.02

Some examples of objects in the four classes are shown in Figure 2.

Figure 2. Example instances of object classes.

3.2 Experiment Setup

Two variants of the YOLOv7 architecture (Wang, Bochkovskiy & Liao, 2023) were used to train the models for the task of maritime object detection, with only the output number of classes modified according to the number of classes in the dataset.

The two model configurations used in this experiment correspond to the full YOLOv7 network and to the much simpler and computationally less demanding YOLOv7-tiny network. The standard YOLO7 architecture is tailored for use on desktop GPUs with 415 layers and about 37 million parameters and requires about 104.7 billion floating point computations for inference. The second variant used in this experiment is the Yolov7-tiny, which comprises 263 layers and about 6 million parameters, and is much less computationally demanding, requiring about 13.8 billion floating point operations. The input image size for both models was 640x640 pixels.

The set task is to detect objects of *ship*, *boat*, *buoy*, and *unknown* classes in images.

The training was done using transfer learning from the model weights pretrained on the MS COCO (Lin et al., 2014) dataset and available at the YOLOv7 repository (Wang, 2023).

The dataset was first randomly split into training, validation, and test sets in 60:15:25 ratio. Then, the training and validation sets were augmented with random 640x640 crops of original images, so that the training set consisted of a mixture of 81 original images and 276 crops, 357 images in total.

The training was performed for 3000 epochs on the training set with learning rate set to 0.01. In comparison to the default training settings in the Github repository (Wang, 2023), some data augmentations were turned off, namely, image translation and scaling, mosaic, mixup and paste probabilities were set to 0.

After 3000 epochs, the weights that performed best on the validation set were tested on the test set.

Models were first evaluated on the rescaled original test images that are typically sized 1920x1080 pixels (scaling strategy), and then on the extracted 640x640 slices that correspond to the native network input size (slicing strategy). To recalculate the ground truth bounding boxes in the sliced images, the yolo-tiling script was used (Neskorozhenyi, 2023)

All experiments were performed in the PyTorch implementation from the YOLOv7 repository using PyTorch 1.10.1, Python 3.8.10 and Ubuntu 20.04 operating system, on a Nvidia RTX 3090 GPU.

Standard Average precision (AP) and mean average precision (mAP) measures (Padilla, Netto & Da Silva, 2020) at 50% IoU threshold and in 50% to 95% IoU range were used to evaluate the performance of the models.

4 Results and Discussion

The results of detection with the trained YOLOv7 and YOLOv7-tiny models are shown in Table 2.

Table 2. Detection results for YOLOv7 and YOLOv7-tiny using scaled and sliced input strategies.

		YOLOv7-tiny		**YOLOv7**	
Class	mAP	Input Scaled	Input Sliced	Input Scaled	Input Sliced
Boat	IoU>0.5	0.687	**0.846**	0.287	0.699
	IoU .5:.95	0.355	**0.639**	0.111	0.488
Ship	IoU>0.5	**0.894**	0.839	0.741	0.555
	IoU .5:.95	0.6	**0.605**	0.414	0.357
Buoy	IoU>0.5	0.532	**0.725**	0.191	0.659
	IoU .5:.95	0.215	**0.426**	0.064	0.39
Un-known	IoU>0.5	**0.97**	0.666	0.187	0.335
	IoU .5:.95	0.38	**0.533**	0.119	0.302
Mean	IoU>0.5	**0.771**	0.769	0.140	0.422
	IoU .5:.95	0.388	**0.551**	0.128	0.362

Although the full YOLOv7 model is much more complex, the obtained performance was in this much better with the simpler YOLOv7-tiny model. This might be due to the quite modest dataset size, and the much larger number of parameters of the full YOLOv7 model that require more training data. Further discussion thus mostly focuses on the YOLOv7-tiny model.

For the *Boat* and *Buoy* classes, the slicing strategy yields significantly better results for both mAP@0.5 and mAP@0.5:0.95 measures, for both models, while for the Ship class there are no significant differences in case of the YOLOv7-tiny model, and for the full model actually performs worse.

The *Unknown* class is underrepresented in the dataset yet is a catch-all class for all unrecognized objects so it may exhibit a great variance in appearance. The main purpose of the class is to prevent the classifier from assigning another class label to parts of image that have a quality of "objectness", but can't be classified into the other classes. Further investigation should be made whether this actually affects the precision of detection of other classes.

Another view of the object detection performance is shown with a confusion matrix (Tables 3 and 4).

Notably, there is very little confusion between the ship and boat classes, and the greatest source of errors are either completely missed detections (background false negative) or false positive detections (background false positive). Missed detections, or false negatives, are as expected largest for the *Buoy* class which appears smaller in the images than the other content classes (*Ship* and *Boat*), although distant ships and boats may also appear small. The false positive problem is present for both Buoy and Boat classes, possibly due to large variety of appearances of boats and small size of buoys.

Table 3. Confusion matrix for the YOLOv7-tiny model and scaling input strategy

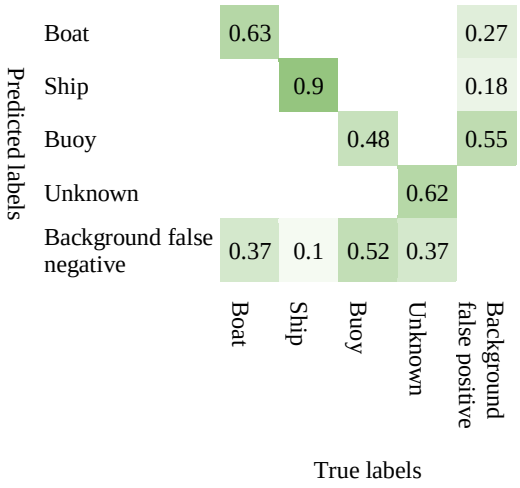
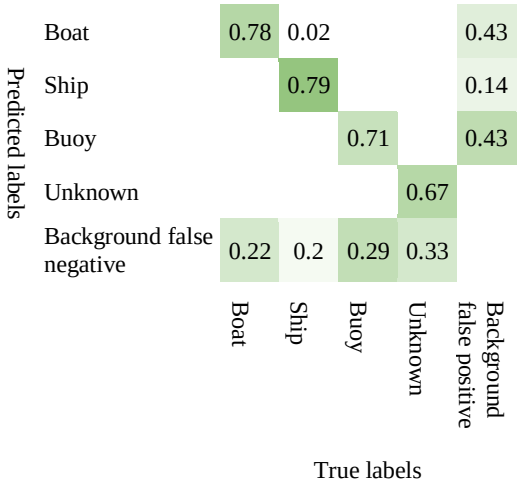


Table 4. Confusion matrix for the YOLOv7-tiny model and slicing input strategy.



Comparing the confusion matrices for the scaling and slicing input strategies (Tables 3 and 4), again it is apparent that slicing helps detection of small objects, where for the Buoy class the percent of true positive detections increased from 48 to 71, and for the Boat class slightly less dramatic from 63 to 78 percent. The true positive rate for the Ship class however fell 10 percent, possibly due to slicing off parts of ship that the network relied on for detection.

This confirms that for the task of detecting small objects, slicing the input image into tiles should be worthwhile despite the added complexity and the need to process the same image several times (equal to the

number of slices), especially if high recall (low missed detection rate) is needed.

A few examples of detection with both models using the scaling strategy are shown in Figure 3, and using the slicing strategy in Figure 4.

Figure 3. Example detection results for the scaling input strategy. Top row: Ground truth, middle row: YOLOv7, bottom row: YOLOv7-tiny.

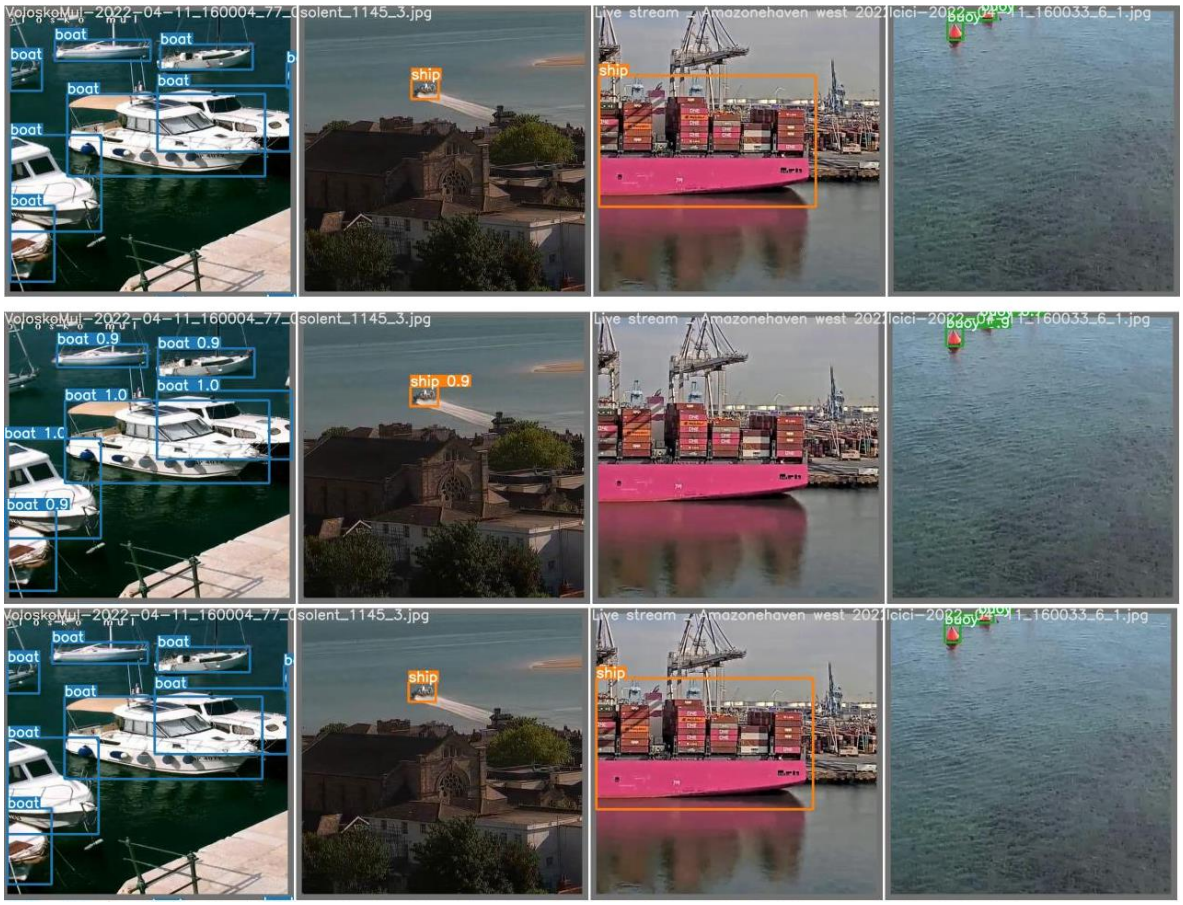


Figure 4. Example detection results for the slicing input strategy. Top row: Ground truth, middle row: YOLOv7, bottom row: YOLOv7-tiny.

5 Conclusion

In this paper, we explored the task of object detection in maritime images on a custom dataset prepared from surveillance cameras. The simpler and faster YOLOv7-tiny performed better than full YOLOv7 model in this experiment, pointing to both suitability of that model for the task and the likely necessity of gathering more training data to fully utilize the more complex model architecture. However, the YOLOv7-tiny architecture is well-suited to the task as is less computationally demanding than the full model and is aimed at the edge GPU devices, that may conceivably be deployed on site near the already installed cameras, avoiding the penalty of high video compression if streaming video is used as input for object detection.

To deal with detection of small objects, which may be important for hazard avoidance or rescue situations, an alternative input strategy to simply rescaling the input image at the network input was considered. Here, the input image was sliced into segments of exact size of network input, without first resizing the image. The experimental results show that this greatly improves detection of small objects such as buoys, raising the true positives from 41 to 71 percent on the test set using the YOLOv7-tiny model.

The detection performance of small objects is still far from perfect, and additional improvements should be considered in future work, for example integration of detections from multiple consecutive frames.

In addition, the dataset will be expanded to include even more variety of weather conditions and locations, and the method tested on other maritime datasets such as SPSCD or SeaDroneSee.

Acknowledgments

This work was fully supported by the EU Horizon project "INNO2MARE: Strengthening the Capacity for Excellence of Slovenian and Croatian Innovation Ecosystems to Support the Digital and Green Transitions of Maritime Regions" (101087348) and University of Rijeka project uniri-drustv-18-288.

References

Benjumea, A., Teeti, I., Cuzzolin, F., & Bradley, A. (2023). *YOLO-Z: Improving small object detection in YOLOv5 for autonomous vehicles* (arXiv:2112.11798). arXiv. http://arxiv.org/abs/2112.11798

Broad, A., Jones, M., & Lee, T. Y. (2018, September). Recurrent Multi-frame Single Shot Detector for Video Object Detection. In *BMVC* (p. 94).

Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

Gasienica-Jozkowy, J., Knapik, M., & Cyganek, B. (2021). An ensemble deep learning method with optimized weights for drone-based water rescue and surveillance. *Integrated Computer-Aided Engineering*, *28*(3), 221-235.

Krišto, M., Ivasic-Kos, M., & Pobar, M. (2020). Thermal object detection in difficult weather conditions using YOLO. *IEEE access*, *8*, 125459-125476.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13* (pp. 740-755). Springer International Publishing.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14* (pp. 21-37). Springer International Publishing.

Neskorozhenyi, R. (2023, June). yolo-tiling [GitHub repository]. GitHub. https://github.com/slanj/yolo-tiling

Padilla, R., Netto, S. L., & Da Silva, E. A. (2020, July). A survey on performance metrics for object-detection algorithms. In 2020 international conference on systems, signals and image processing (IWSSIP) (pp. 237-242). IEEE.

Petković, M., Vujović, I., Lušić, Z., & Šoda, J. (2023). Image Dataset for Neural Network Performance Estimation with Application to Maritime Ports. *Journal of Marine Science and Engineering*, *11*(3), 578.

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).

Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Sambolek, S., & Ivasic-Kos, M. (2021). Automatic person detection in search and rescue operations using deep CNN detectors. *Ieee Access*, *9*, 37905-37922.

Stojnić, V., Risojević, V., Muštra, M., Jovanović, V., Filipi, J., Kezić, N., & Babić, Z. (2021). A Method for Detection of Small Moving Objects in UAV

Videos. *Remote Sensing*, *13*(4), 653. https://doi.org/10.3390/rs13040653

Tan, M., Pang, R., & Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10781-10790).

Tong, K., & Wu, Y. (2022). Deep learning-based detection from the perspective of small or tiny objects: A survey. *Image and Vision Computing*, *123*, 104471.

Unel, F. O., Ozkalayci, B. O., & Cigla, C. (2019). The Power of Tiling for Small Object Detection. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 582–591. https://doi.org/10.1109/CVPRW.2019.00084

Varga, L. A., Kiefer, B., Messmer, M., & Zell, A. (2022). SeaDronesSee: A Maritime Benchmark for Detecting Humans in Open Water. *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 3686–3696. https://doi.org/10.1109/WACV51458.2022.00374

Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 7464-7475).

Wang, C. Y. (2023, June). Yolov7 (Version 0.1) [GitHub repository]. GitHub. https://github.com/WongKinYiu/yolov7/

Zhu, Q., Ma, K., Wang, Z., & Shi, P. (2023). YOLOv7-CSAW for maritime target detection. *Frontiers in neurorobotics*, *17*, 1210470. https://doi.org/10.3389/fnbot.2023.1210470