Towards Real-Time Human Detection in Maritime Environment Using Embedded Deep Learning

Mostafa Rizk[1,3,4(✉)], Fatima Slim[3], Amer Baghdadi[1], and Jean-Philippe Diguet[2]

[1] IMT Atlantique, Lab-STICC, UMR CNRS 6285, 29238 Brest, France
mostfa.rizk@imt-atlantique.fr
[2] CNRS, IRL CROSSING, Adelaide, Australia
[3] Physics and Electronics Department, Lebanese University, Beirut, Lebanon
[4] CCE Department, Lebanese International University, Beirut, Lebanon

Abstract. Marine search and rescue missions necessitate a lot of effort and expenses. The use of technological advancements facilitates discovering and locating individuals and aids in the directing of rescuers and medical teams. This has the potential to save human lives while also lowering costs. The characteristics of the marine environment create additional challenges for computer vision techniques used to detect the presence of human in a scene. Currently, artificial intelligence (AI) techniques based on convolution neural networks (CNNs) provide solid solutions to detect and locate objects. In this paper, the relevance of the emergent You Only Look Once (YOLO) in detecting humans in maritime environment is investigated. The available models of YOLOv4 are trained using a custom dataset. The trained models are evaluated using recognized evaluation parameters. In addition, the inference speed is reported targeting embedded low-power hardware platforms dedicated for AI applications.

Keywords: Deep learning · YOLO · Maritime · Human detection · Man overboard · Search and rescue

1 Introduction

Maritime search and rescue (SAR) missions are crucial for most coastal states. According to the International Organization for Migration 218,062 irregular maritime migration attempts are recorded in the Mediterranean Since 2014 [1] From which, 23,939 dead and missing persons are recorded during attempted overseas crossings. Furthermore, the European Maritime Safety Agency reports in the Annual Overview of Marine Casualties and Incidents 2021 [2] that during the 2014–2020 period, 367 marine casualties resulted in a total of 550 lives lost and

This work was supported in part by the Regional Council of Bretagne through the ODESSA FEDER project.

M. Valle et al. (Eds.): SYSINT 2022, LNNS 546, pp. 583–593, 2023.
https://doi.org/10.1007/978-3-031-16281-7_55

6921 injuries in the waters of European Union (EU) Member States or involving EU ships. The ability to quickly locate missing people aids in the direction of rescuers and medical personnel, which plays an important role in increasing the chances of saving human lives while also lowering costs.

Years ago, visual surveillance in the maritime domain has been explored. However, most surveillance activities have been assigned to areas near the coasts and ports and mainly depend on human monitoring and analysis for security reasons. Computer vision techniques are also adopted in few works. However, videos and images capturing maritime environment pose challenges that are absent or less severe in other environments such as the dynamic nature of the background, unavailability of static cues, presence of small objects at distant backgrounds and illumination effects [3]. These challenges impact the efficiency of traditional computer vision techniques in detecting individuals in marine environments.

Recently, deep learning approaches have introduced efficient solutions to detect, classify and localize several objects in images and videos. In particular, the evolution of neural networks architectures has elevated the performance to a point that they are considered on par with human performance for some of these problems. However, the detection performance comes at the cost of increased hardware resources and power consumption especially for real-time scenarios with high requirements of accuracy and precision. You Only Look Once (YOLO) has been recently introduced as an efficient unified model of all phases of a CNN for doing object detection in real-time. The recent version of YOLO, so called YOLOv4, has been justified to detect objects in real-time with high level of precision. Several models of YOLOv4 exist, with different architecture specifications and consequently different detection performance in terms of accuracy and precision, detection speed and required energy budget.

The growing use of artificial intelligence (AI) based detection methods is of great interest in aiding SAR missions [4–8]. However, only few works have addressed the detection of humans in open water or for man overboard accidents [9–11]. Other available works adopting deep learning in marine environment have focused mainly on the detection of sea ships [12]. This work aims to enable efficient detection and localizing of floating humans in real-time based on AI techniques. In particular, the relevancy of YOLOv4 [13] in detecting humans in maritime environments to aid marine SAR missions is addressed. The work includes collecting a custom dataset, training different YOLOv4 available models and evaluating the trained model using mean average precision (mAP), precision, recall, and F1-score. Also, the trained models are implemented targeting Jestson Nano and Jetson Xavier development kits from Nvidia. For different power modes, the inference speed is attained while processing real-life videos. The obtained results show that YOLOv4 can achieve real-time detection when implemented on low-cost, small size embedded platforms with reduced power consumption. This paves the way to develop airborne systems or edge embedded systems mounted on shore, moving boats or floating buoys that can be exploited to facilitate search and rescue missions and in optimizing the man overboard signaling systems.

2 Background

2.1 Object Detection

Previously, object detection has been achieved using computer vision techniques based on feature extraction such as histogram of oriented gradients (HOG) [14] and scale-invariant feature transform (SIFT) [15]. Currently, artificial intelligence (AI) techniques based on convolution neural networks (CNNs) are the dominant methods for object detection, which compromise both classification and localizing of objects within the image by determining bounding boxes (coordinates and size) around the objects of interest. Several techniques based on CNN are developed targeting object detection. Two-stage models such as region-convolutional neural network (R-CNN) [16] apply classification of objects based on pre-selected regions. The post-processing operations required to refine the bounding boxes, eliminate duplicates and adjust the detection scores increase the complexity and impact the speed of detection. Despite the introduction of R-CNN enhanced versions [17,18], real-time detection has not been granted.

You Only Look Once (YOLO) has been proposed in [19] as an efficient unified model of all phases of a CNN for doing object detection in real-time. Several versions of YOLO have been developed by modifying the network architecture. In YOLOv2 [20], the fully connected layers at the end have been eliminated and Darknet-19 architecture has been adopted. YOLOv3 [21] uses Darknet-53 architecture and inherits the concept of residual networks. The detections are made at 3 different scales which enables the detection of small objects. Recently, YOLOv4 [13] object detection method has been introduced. It outperforms other available methods in terms of speed and accuracy performance. The experiments targeting Microsoft Common object in context (COCO) dataset [22] show that YOLOv4 is faster and more accurate than real-time neural networks EfficientDet [23] and RetinaNet [24] provided by Google and Facebook respectively.

The architecture of YOLOv4 consists of the backbone, neck and dense prediction so-called the head. The backbone is in charge of extracting features. The neck aggregates the features and delivers them to the detection head. Based on several experiments and comparisons [13], CSPDarknet53 is selected for the backbone. Spatial Pyramid Pooling block (SPP) is added to the PANet path-aggregation neck. The anchor based YOLOv3 is adopted as detection head in YOLOv4.

YOLOv4 exploits a set of universal methods that are assumed to improve CNN accuracy for majority of models, tasks, and datasets. These universal methods are data augmentation (DA), Weighted-Residual-Connections (WRC), Cross-Stage-Partial-connections (CSP), Cross mini-Batch Normalization (CmBN), Self-adversarial-training (SAT) and Mish-activation. These universal methods are implemented in combination with new devised methods such DropBlock regularization, and Complete-IoU loss. YOLOv4 employs these available methods in two ways in order to create a more efficient and powerful object detection model: *Bag-of-Freebies* and *Bag-of-Specials*. *Bag-of-Freebies* compromises training strategies and pre-processing methods. Adopting these strategies

enhance the training without impacting the inference performance as training is done offline. Data augmentation is used to alleviate the degree of variability of training images in order to increase the robustness of the detection during inference against unknown environments. Data augmentation includes pixel-wise computer vision techniques such cutmix, mosaic, image resizing, blurring, image rotating, random scaling, flipping, cropping and changing the exposure, saturation and hue. Focal loss is also adopted to address the issue of data imbalance existing between various classes. Label smoothing is used to convert hard labels into soft labels leading to improving the robustness of the model. *Bag-of-Specials* contains architecture-related plug-in modules and post-processing methods introduced. Mish activation is used for both backbone and detector. CSP and Multi-input weighted residual connections are selected for the backbone. SPP-block, SAM-block, PAN path-aggregation block are added to the neck/detector.

2.2 Human Detection Using Deep Learning Methods

Several works have adopted deep learning techniques to detect individuals for several applications such as social distancing [25], crowd detection, security and search and rescue missions [4–8]. However, few works have addressed human detection in marine environment. In [9], the authors have exploited YOLOv3 Tiny to detect human swimming in open water via areal images. The authors have deployed the trained network on NVIDIA Jetson TX1 platform to enable real-time detection of human in search and rescue missions using UAVs. In [26], SSD and YOLOv3 have been examined to detect man overboard event detection. The authors have not presented the performance results. In [10], Faster R-CNN has been employed to locate the person in water using thermal images. In [11], YOLOv3 has been utilized to detect and localize human in marine environment using images captured by UAVs for search and rescue missions. The authors have focused on analyzing the effects of flight altitude on the detection performance. The used dataset for training, validating and testing includes 450 images only, which are collected in one location. Note that in [10] and [11] the training and testing results in terms of precession are only shown without presenting the achieved detection speed or indicating the used target device.

3 Method

3.1 Dataset

We create a diverse dataset of images showing humans in maritime environment. The images are collected from several internet resources. We make use of the dataset published by [9]. The dataset offers images extracted by videos captured by the means of UAV for Humans swimming in open water. We edited this dataset by eliminating images with high similarity. Also, a great effort is done to enhance the labeling by adjusting the existing bounding boxes to meet with the dimensions of the persons and by adding bounding boxes of unlabeled persons. In addition,

we add 2000 new images including showing persons in maritime environment with different positions and from different perspectives. The number of humans in the scene varies between the gathered images. In addition, the images show human bodies in numerous positions and different perspectives and scales, and have various backgrounds, lighting conditions and resolutions. The final dataset includes 6462 images with 16795 bounding boxes[1]. The images are split randomly by 70% as training dataset, 10% as validation dataset and 20% as testing dataset. Table 1 shows the distribution of images and objects in each dataset.

Table 1. Specifications of the created dataset

Dataset	Training	Validation	Testing	Total
Number of images	4463	666	1333	6462
Number of objects	11913	1677	3205	16795

3.2 Target Models

In this work we examine three different YOLOv4 networks: YOLOv4 Large, YOLOv4 Tiny and YOLOv4 Tiny-3l. The original YOLOv4 network consists of 162 layers and uses mish activation functions. YOLOv4 Tiny is the compressed version of YOLOv4. It uses the simplified network structure of CSPDarknet53-tiny. It compromise 38 layers with LeakyRelu activation functions and only two detector heads. YOLOv4 Tiny-3l architecture is similar to YOLOv4 Tiny, but with three detector heads. Table 2 presents the target networks specifications.

Table 2. Specifications of the targeted YOLOv4 models

Model	Number of layers	Activation function	Model weights' volume (MB)
YOLOv4	162	Mish	256.2
YOLOv4 Tiny	38	LeakyRelu	23.5
YOLOv4 Tiny-3l	45	LeakyRelu	24.5

3.3 Training

The training is conducted using the Darknet framework [27] using Quadro RTX 4000 from Nvidia. Transfer learning is adopted in order to maintain the generalization. We make use of the weights generated in previous training processes of networks with similar architecture specifications targeting COCO dataset. Note that the imported weights of the feature extraction layers are kept; whereas, the weights of the neck and the detector layers are eliminated. The networks' general

[1] https://www.kaggle.com/datasets/mostafarizk/maritimesar.

architectures have not been altered. Only the depth size of the three convolution layers allocated before the YOLO detector layers are adjusted. The number of filters in these three convolution layers are modified considering our case where only one class (Person) is targeted.

The number of images per batch is set to 64. The total number of iterations is set to 2000. The initial learning rate for training is set to 0.001 and it scales down two times by 0.1 at iteration 1600 and 1800. The input images are down sampled into 416×416 or 608×608. While training the models, data augmentation is activated. Mosaic data augmentation type is used where 4 images are merged into one. When activated, Cutmix data augmentation type is applied for the classifier only. The saturation of input images and their exposure (brightness) are randomly changed as well as the rotation.

The models are validated using the validation dataset. Mean average precisions (mAP) is calculated during training for each 4 epochs. Figure 1 illustrates the training performances. Note that the blue curves correspond to the training losses whereas the red curves corresponds to the computed mAP values. The mAP calculation starts after 1000 iterations and it adopts the AP50 metric defined in the MS COCO competition (same to the metric of precision in the Pascal VOC competition) and uses the following expressions to compute the Precision and Recall values:

$$P = \frac{TP}{(TP + FP)} \qquad R = \frac{TP}{(TP + FN)} \tag{1}$$

where P is the Precision, R is the Recall and TP, FP and FN stand for True Positive, False Positive and False Negative respectively. Table 3 shows the required time for training the targeted networks with different input resolutions using Nvidia Quadro RTX 4000.

3.4 Evaluation

The trained models are evaluated using the test dataset. Sample detection results from network testing are shown in Fig. 2. The figure shows that trained models are able to accurately detect and classify the presence of human bodies in different maritime environments. Table 3 shows the obtained mean average precision considering VOC07 and VOC12 performance metrics [28]. In addition, the table shows the obtained values of precession, recall, F1-score and average intersection over union (IOU) considering 0.5 IOU threshold.

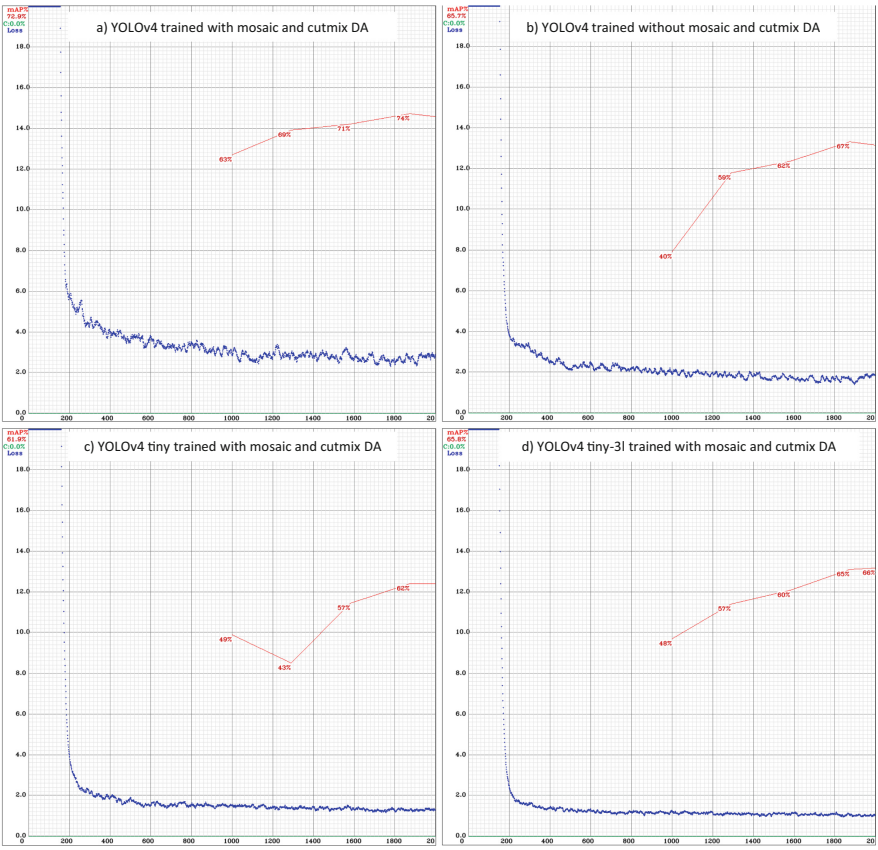

Fig. 1. Sample training performances

Furthermore, the inference speed of trained models is evaluated using several captured videos targeting embedded platforms. Table 4 shows the obtained speed of the trained models in frames per second (FPS) when applied to the captured videos on Jetson Nano and Jetson Xavier NX development kits while operating on different power modes. Both used kits are small powerful computers that allow running neural networks for applications like image classification, object detection, segmentation, etc. Jetson Nano provides 472 GFLOPS of FP16 computing performance with 5W and 10W of power consumption. Whereas, Jetson Xavier NX provides up to 21 TeraOPS of compute performance in configurable

Table 3. Evaluation results of the trained YOLOv4 models

Target model	Image resolution	Training time (h)	Data augmentation	mAP VOC07	mAP VOC12	Precession	Recall	F1 score	avg IOU
YOLOv4 Large	416 × 416	03:22	–	60.46	58.78	61.61	70.80	65.88	62.14
			mosaic	64.27	65.66	62.57	75.48	68.42	63.8
			mosaic+cutmix	65.63	69.04	61.95	78.03	69.07	64.83
	608 × 608	06:00	–	55.16	59.15	66.15	69.98	68.01	63.13
			mosaic	64.64	64.91	66.74	73.45	69.93	64.92
			mosaic+cutmix	65.82	69.37	63.96	78.28	70.4	65.39
YOLOv4 Tiny	416 × 416	00:24	–	57.00	56.53	49.03	73.39	58.79	61.34
			mosaic	56.34	56.10	48.25	73.95	58.40	61.75
			mosaic+cutmix	56.90	56.90	47.40	73.85	57.74	61.85
	608 × 608	00:35	–	59.29	60.07	53.98	74.91	62.75	62.08
			mosaic	60.91	63.10	52.83	77.00	62.66	62.58
			mosaic+cutmix	60.59	62.47	53.04	76.57	62.67	62.66
YOLOv4 Tiny-3l	416 × 416	00:25	–	54.89	53.31	53.78	72.32	61.69	60.97
			mosaic	55.28	54.17	53.16	73.04	61.53	61.57
			mosaic+cutmix	55.80	55.41	53.09	73.95	61.81	61.88
	608 × 608	00:47	–	59.12	57.46	59.27	70.92	64.57	61.81
			mosaic	59.54	59.43	56.63	73.26	63.88	62.35
			mosaic+cutmix	60.08	59.92	57.60	73.17	64.46	62.21

Table 4. Average detection performance in FPS

Trained model	Input frame resolution	Jetson Nano		Jetson Xavier NX					
		Mode0	Mode1	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
		10W	5W	15W 2CORE	15W 4CORE	15W 6CORE	10W 2CORE	10W 4CORE	10W Desktop
YOLOv4 Large	416 × 416	2.0	1.5	10.1	10.7	10.8	8.8	9.4	6.5
	608 × 608	1.0	0.7	5.6	5.7	5.7	5.0	4.9	3.7
YOLOv4 Tiny	416 × 416	19.2	12.5	58.4	80.6	72.0	54.3	67.4	53.0
	608 × 608	9.6	6.5	35.3	44.3	45.6	31.3	39.4	30.3
YOLOv4 Tiny-3l	416 × 416	16.8	10.9	50.1	60.0	69.9	43.6	58.6	48.0
	608 × 608	8.5	5.7	32.6	39.7	40.3	35.1	35.7	26.8

10W or 15W power budgets by capping the GPU and CPU frequencies and the number of online CPU cores at a pre-defined level. Figure 3 shows samples of the obtained detection results in captured video sequences. The obtained results show that applying DA enhances the detection performance (mAP, precision, recall, F1-score and average IOU). The use of cutmix DA increases the enhancement ratio in most of the cases. The use of higher image resolution enhances the mAP performance but at the cost of reduced inference speed and longer training time.

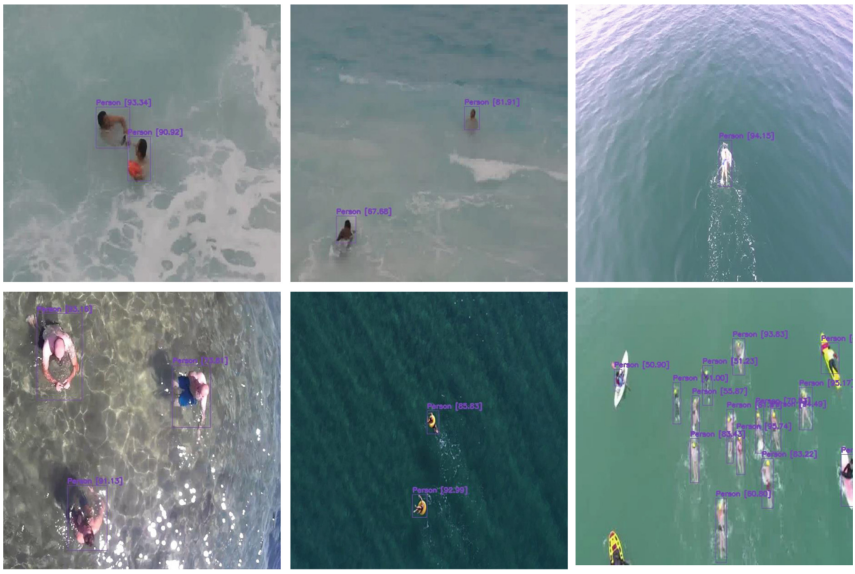

Fig. 2. Sample detection results in testing dataset images

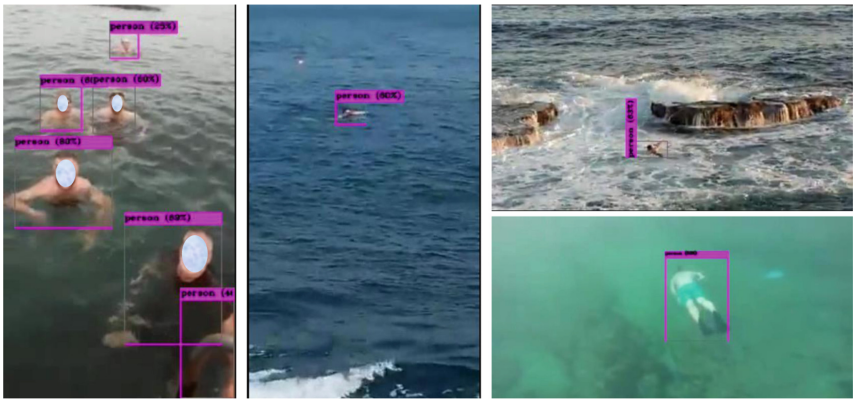

Fig. 3. Samples of the obtained detection results in video sequences

4 Conclusion

In this paper, the use of YOLOv4 in detection of humans in maritime environments is investigated. Available YOLOv4 architectures are trained on a custom dataset. The trained models are evaluated in terms of mAP, precession, recall and average IOU. Also, the performances of the models are examined on embedded platforms using our own videos showing humans in open water. The obtained results show that YOLOv4 can achieve real-time detection of humans in maritime environment with acceptable accuracy and precession. For example,

YOLOv4 Tiny achieves an inference speed of 45.6 FPS with mAP of 63.10 when running on Jetson Xavier NX considering 608×608 resolution. Future work will include applying optimization techniques such as quantization and pruning to increase the inference speed and study their impact on the detection performance.

References

1. International Organization for Migration Missing Migrants Project website. https://missingmigrants.iom.int. Accessed 1 May 2022
2. E. M. S. Agency: Annual overview of marine casualties and incidents 2021, EMSA, Annual Report, December 2021
3. Prasad, D.K., et al.: Challenges in video based object detection in maritime scenario using computer vision. arXiv preprint arXiv:1608.01079 (2016)
4. Castellano, G., Castiello, C., Mencar, C., Vessio, G.: Preliminary evaluation of TinyYOLO on a new dataset for search-and-rescue with drones. In: International Conference on Soft Computing Machine Intelligence (ISCMI), pp. 163–166 (2020)
5. Liu, C., Szirányi, T.: Real-time human detection and gesture recognition for on-board UAV rescue. Sensors **21**(6), 2180 (2021)
6. Rizk, M., Slim, F., Charara, J.: Toward AI-assisted UAV for human detection in search and rescue missions. In: 2021 International Conference on Decision Aid Sciences and Application (DASA), Sakheer, Bahrain, December 2021, pp. 781–786 (2021)
7. Sambolek, S., Ivasic-Kos, M.: Automatic person detection in search and rescue operations using deep CNN detectors. IEEE Access **9**, 37 905–37 922 (2021)
8. Rosero, R.L., Grilo, C., Silva, C.: Deep learning with real-time inference for human detection in search and rescue. In: Abraham, A., Piuri, V., Gandhi, N., Siarry, P., Kaklauskas, A., Madureira, A. (eds.) Intelligent Systems Design and Applications, pp. 247–257. Springer, Cham (2021)
9. Lygouras, E., et al.: Unsupervised human detection with an embedded vision system on a fully autonomous UAV for search and rescue operations. Sensors **19**(16), 3542 (2019)
10. Feraru, V.A., Andersen, R.E., Boukas, E.: Towards an autonomous UAV-based system to assist search and rescue operations in man overboard incidents. In: IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 57–64, UAE, Abu Dhabi, November 2020
11. Qingqing, L., et al.: Towards active vision with UAVs in marine search and rescue: analyzing human detection at variable altitudes. In: IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 65–70, UAE, Abu Dhabi, November 2020
12. Qiao, D., Liu, G., Lv, T., Li, W., Zhang, J.: Marine vision-based situational awareness using discriminative deep learning: a survey. J. Marine Sci. Eng. **9**(4), 397 (2021)
13. Bochkovskiy, A., Wang, C., Liao, H.M.: YOLOv4: optimal speed and accuracy of object detection. CoRR, vol. abs/2004.10934 (2020). https://arxiv.org/abs/2004.10934
14. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of the IEEE Computer society Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 886–893 (2005)

15. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision **60**, 91–110 (2004)
16. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 580–587 (2014)
17. Girshick, R.: Fast R-CNN. In: IEEE International Conference on Computer Vision (ICCV) 2015, pp. 1440–1448 (2015)
18. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 28. Curran Associates Inc. (2015)
19. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You Only Look Once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788 (2016)
20. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017
21. Redmon, J.: YOLOv3: an incremental improvement (2018)
22. COCO - common objects in context web site. https://cocodataset.org/. Accessed 20 June 2020
23. Tan, M., Pang, R., Le, Q.V.: EfficientDet: scalable and efficient object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10 781–10 790 (2020)
24. Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2980–2988 (2017)
25. Hraybi, S., Rizk, M.: Examining YOLO for real-time face-mask detection. In: Smart Cities Symposium (SCS 2021), vol. 2021. Institution of Engineering and Technology, pp. 571–575 (2021)
26. Katsamenis, I., Protopapadakis, E., Voulodimos, A., Dres, D., Drakoulis, D.: Man overboard event detection from RGB and thermal imagery: possibilities and limitations. In: Proceedings of the 13th ACM International Conference on Pervasive Technologies Related to Assistive Environments, ser. PETRA 2020. New York, NY, USA. Association for Computing Machinery (2020)
27. Redmon, J.: Darknet: Open source neural networks in C. https://pjreddie.com/darknet/. Accessed 14 Apr 2022
28. Padilla, R., et al.: A comparative analysis of object detection metrics with a companion open-source toolkit. Electronics **10**(3), 279 (2021)