Gorad, Ajinkya; Hassan, Syeda Sakira; Särkkä, Simo

Vessel Bearing Estimation Using Visible and Thermal Imaging

Published in:
Image Analysis - 23rd Scandinavian Conference, SCIA 2023, Proceedings

DOI:
10.1007/978-3-031-31438-4_25

Published: 01/01/2023

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Please cite the original version:
Gorad, A., Hassan, S. S., & Särkkä, S. (2023). Vessel Bearing Estimation Using Visible and Thermal Imaging. In R. Gade, M. Felsberg, & J.-K. Kämäräinen (Eds.), *Image Analysis - 23rd Scandinavian Conference, SCIA 2023, Proceedings* (pp. 373-381). (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Vol. 13886 LNCS). Springer. https://doi.org/10.1007/978-3-031-31438-4_25

Vessel bearing estimation using visible and thermal imaging

Ajinkya Gorad, Sakira Hassan, and Simo Särkkä

Aalto University, Dept. of Electrical Engineering and Automation
Otakaari 3, 02150 Espoo, Finland

Abstract. Maritime awareness and autonomous navigation can be enabled by state-of-the-art deep learning methods, by monitoring and tracking the position of maritime vessels. In our experiment, we acquire ship dataset from Megastar cruise ferry campaigns in Baltic Sea. We detect the nearby vessels using visible and infrared imaging sensors and object detectors You-Only-Look-Once v5 (YOLOv5), and Detectron2 RCNN network and use that information along with DeepSORT method to track the position of the vessel. We obtain the bearing of the vessels detected from both infrared and visible image sequences of ship and fuse them using a Kalman filter (KF) and Rauch-Tung-Striebel (RTS) smoother. We then compare the result to the bearing obtained from the automatic identification system (AIS) of the vessel, and also compare it among object detectors. We obtained a root-mean-square error of 0.17° in vessel bearing tracking as compared to AIS.

Keywords: Maritime awareness · Ship detection · Bearing estimation

1 Introduction

Maritime awareness has recently gained large interest due to its need in enabling autonomous harbour-to-harbour navigation [22]. Awareness can be obtained using various sensing like incoming sound direction, or sound reflections through sonar, or imaging in various electromagnetic spectra through commercially available cameras. From visible light to heat emissions, visual information can provide the vessel a large amount of data regarding its environment and other ships in addition to the vessel's automatic identification system (AIS), and radar.

Artificial Intelligence (AI) using deep learning on image information through cameras can provide detection of nearby obstacles and avoid them, such as other vessels, buoys, small boats, and so on. Moreover, small objects which are unable to transmit the AIS messages, or objects which are undetected by radar can be detected, and even detecting breach from those ships deliberately switching off AIS. Such use of cameras can offer improved safety of the vessel and could provide critical information in detecting an imminent collision with the nearby vessels or obstacles. Projects like Mayflower by International Business Machines (IBM) have already demonstrated working prototypes of autonomous ships [1].

Numerous developments have been accomplished in the past decade to aid autonomous navigation system through imaging sensors for effective target tracking [6,14,21]. For instance, feature-from-the-accelerated-segment-test (FAST) algorithm has been used to detect a small boat object in a Region of Interest (ROI) near horizon, and also its bearing and range information in real time [6]. The result is used to estimate the trajectory of the target ship by using an extended Kalman filter (EKF). In recent years, feature extraction in object detection task has been significantly improved by AI-based algorithms.

Imaging sensors and machine learning techniques such as convolutional neural networks (CNNs) have been used to detect and classify marine objects [22]. Region proposal CNN (R-CNN) networks were among the earliest object detectors with good performance, while its improved versions are Fast-RCNN, and Faster-RCNN [8, 9, 17]. Deep networks have also be utilized to perform a fused image detection of ships [7]. Among CNNs, You-Only-Look-Once (YOLO) [16] has been proven to be a very efficient and fast object detection method, and also are improved for maritime object detection tasks [10]. An object detection on an thermal infrared image taken from sky is presented in [12]. A YOLOv3 ship detection in both visible and near-infrared from dataset taken in northern Taiwan port is achieved in [4,5]. U-Net segmentation and YOLOv4/5 detections on thermal images of ships are presented in [18]. Fusion of AIS and visible image information is used to get better position estimate in [15].

Improvements have been made by modifying YOLO network. A YOLOv3 network optimized by Holistically-nested edge detection (HED) is used to detect and track moving marine targets in [24]. Reverse depthwise separable convolution (RDSC) was fused with the backbone of YOLOv4 in [13] and GhostbottleNet algorithm was merged with YOLOv5 in [23]. Recent developments in online target tracking, such as SORT [2], used YOLO for detecting the targets and estimate bounding box of the object in a video. Its successor DeepSORT tackled longer periods of occlusions in video sequences by using appearance feature descriptors for improved association [25].

In this work, we adopt YOLOv5 [11], and Detectron2 RCNN-FPN [26] object detectors with DeepSORT tracking method to investigate the use of visible and infrared imaging to detect various nearby vessels and obtain their relative bearing information. We report multi-sensor ship detection and tracking method (Section 2), and report the results (Section 3) obtained from both the detectors and tracking with DeepSORT with OSNet [3]. We fuse the bearings measurements obtained from visible and infrared camera detections with Kalman filter and Rauch-Tung-Striebel (RTS) smoother, and compare results among detectors, and to AIS. The method is applied to the data we obtained from Megastar cruise ferry campaign in Baltic Sea.

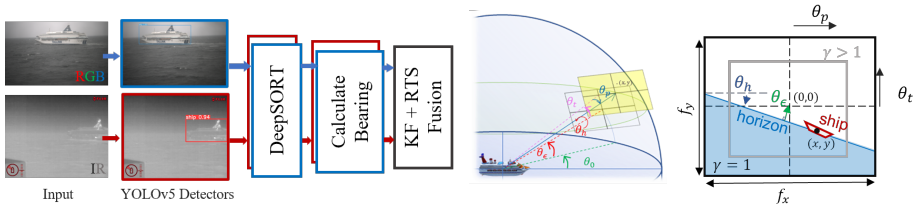
2 Materials and Methods

This section summarizes the data and methods used in our work.

2.1 Data collection, labeling, and training

The data collection experiment was conducted in Baltic Sea on Megastar cruise ferry which operates on Helsinki–Tallinn route and takes approximately 3 hours in one direction (∼80km). Two campaigns were conducted: first on 16th February 2021, and the second on 26th August, 2021. Route had sparse maritime traffic yet we were able to obtain image data from few nearby passing cruise ships. A FLIR Blackfly visible range camera with resolution of 1920×1200 px and a maritime grade FLIR M232 thermal infrared camera with resolution of 320×240 px were mounted on the command deck of the Megastar cruise. We collected data with both cameras and from the vessel's DGNSS SAAB R5 automatic identification system (AIS) system having self positioning error <2.5 m which provided ground-truth measurements for surrounding vessels. Dataset contained 2465 IR images, and 893 RGB images. In which, IR had 2482 ship labels, and 57 bird labels. And, RGB had 444 buoy, 384 boat, 795 ship, 6 helicopter, 9 bird, and 105 lighthouse labels. In this dataset, we were able to obtain synced RGB and IR image frames for 3 ships. Unfortunately for unknown reasons, AIS data was only found available for the Viking XPRS ship during our campaign, and was missing for other encounters such as Silja Europa.

We used a train–test partition of 70–30 from the shuffled dataset. Model YOLOv5-s was trained on visible, and infrared dataset separately for 500 epochs with 8 images per batch, and it took 12 hours on Tesla P100-PCIE-16GB GPU for visible,and 3 hours for infrared, which ended at 464 epochs with an early stop. We also trained a Detectron2 Base-RCNN-FPN detector on the same datasets with batch size of 16 for 5000 epochs with learning rate of 0.001, and it took 3 hours for visible, and 1.6 hours for infrared on the same device.



(a) Algorithm (b) Camera positioning

Fig. 1: Algorithm flowgraph for ship bearing estimation (a). Camera angle definitions (b).

2.2 Vessel detection and tracking

In this section we describe the approach used for vessel detection and tracking. Method flow is presented in Fig. 1a. The image sequence consists of images

from both the test set. We use a combination of an object detectors YOLOv5 and Detectron2 based Base-RCNN-FPN along with DeepSORT tracking with OSNet features [3] with the our trained model to detect and track ships in image sequences. Fig. 2 consists of few images selected from an encounter with the Viking XPRS and Silja Europa cruise. A confidence threshold of 0.01 was set during detection. We then convert these coordinates into bearing information. These bearings are then fused to provide an bearing estimate of the target ship.

2.3 Vessel bearing calculation

Ship vessel bearing estimation through cameras is as follows. On both the visible and infrared views, YOLO ship detections are obtained and bearing is obtained from calibrated field-of-view (FoV) of camera. Camera angle parameters are defined in Fig. 1b. Let (x, y) be the camera coordinates in pixels of the center of the detected bounding box of a ship in an image. Camera placement is at an azimuth-elevation offset of $(\theta_0, \theta_\epsilon)$ w.r.t. the heading of ship. Camera has pan-zoom-tilt (PZT) as $(\theta_p, \gamma, \theta_t)$, and a small horizon angle θ_h is measured clockwise in the image. For each camera image, H, W be the height, and the width of the image in pixels, f_y be the vertical FoV, and f_x be the horizontal FoV. Then the calculated approximate bearing β, and elevation ψ of the observed ship w.r.t. heading of the on-boarded ship (1b), is:

$$\begin{bmatrix}\beta\\ \psi\end{bmatrix} = \frac{1}{\gamma}\begin{bmatrix}\cos\theta_h & \sin\theta_h\\ -\sin\theta_h & \cos\theta_h\end{bmatrix}\begin{bmatrix}\arctan(2\tan(\frac{f_x}{2})(\frac{x}{W}-0.5)) + \theta_p\\ \arctan(2\tan(\frac{f_y}{2})(\frac{y}{H}-0.5)) + \theta_t\end{bmatrix} + \begin{bmatrix}\theta_0\\ \theta_\epsilon\end{bmatrix}. \quad (1)$$

Camera FoV and offset is manually calibrated using sequence generated from a train images set and AIS calculated bearings. FoV calibration on the same image sequence resulted in the value of f_x as 19° for infrared and 20.0° for visible-range camera. Camera azimuth offsets θ_0 are -79.2°, and -61.8° while camera elevation offsets θ_ϵ are 2.7°, and 3.5°, for infrared and visible camera respectively obtained relative to ship heading. FLIR infrared camera pan information θ_p was extracted from its image data, and visible camera had no pan control. Horizon angle θ_h is found to be $-1.55°$ in visible image, and infrared had no noticeable tilt, which also can be concluded from images in Fig. 2.

AIS bearing β_0 relative to Megastar heading is calculated from the AIS log messages. At a given time instant, for the Megastar the heading is θ_h, and its latitude and longitude are $(\varphi_\mathrm{R}, \lambda_\mathrm{R})$, while the nearby target vessel has its value of $(\varphi_\mathrm{T}, \lambda_\mathrm{T})$. AIS based bearing relative to megastar heading is then calculated using the following formula:

$$\beta_0 = \theta_h - \arctan\left(\frac{\sin(\lambda_\mathrm{T}-\lambda_\mathrm{R})\cos\varphi_\mathrm{T}}{\cos\varphi_\mathrm{R}\sin\varphi_\mathrm{T} - \sin\varphi_\mathrm{R}\cos\varphi_\mathrm{T}\cos(\lambda_\mathrm{T}-\lambda_\mathrm{R})}\right). \quad (2)$$

2.4 Vessel bearing fusion

Fused estimate of bearings is obtained by combining of the infrared and visible bearing measurement. It is accomplished using a Kalman filter followed by the

(a) Viking XPRS (b) Silja Europa

Fig. 2: Ship detection (YOLOv5) shown for Viking XPRS (a) and Europa (b) in visible (top) and in infrared (bottom). Shown blue bounding boxes are hand labels, whereas, the detected and tracked bounding boxes are shown in black with its confidence values.

Rauch-Tung-Striebel (RTS) smoother [19, 20]. We manually choose the tracked bearings of the same ship after the DeepSORT, and filter through KF using discretized Wiener velocity model with time step differences obtained from the measurements. The model has the form:

$$\mathbf{x}_i = \mathbf{A}_i \mathbf{x}_{i-1} + \mathbf{q}_{i-1}, \quad \mathbf{y}_i = \mathbf{H}\mathbf{x}_i + \mathbf{r}_i, \tag{3}$$

$$\mathbf{A}_i = \begin{bmatrix} 1 & \Delta T_i \\ 0 & 1 \end{bmatrix}, \quad \mathbf{Q}_i = \begin{bmatrix} \frac{\Delta T_i^3}{3} & \frac{\Delta T_i^2}{2} \\ \frac{\Delta T_i^2}{2} & \Delta T_i \end{bmatrix}, \tag{4}$$

where the state contains bearing and its derivative $\mathbf{x} = [\beta, \dot{\beta}]^\intercal$, state transition matrix with Wiener velocity model at time step i as $\mathbf{A}_i$ and process noise $\mathbf{q}_i \sim N(0, 0.01^2\mathbf{Q}_i)$ with its covariance matrix $\mathbf{Q}_i$. Measurement matrix $\mathbf{H} = [1\ 0]$ with measurement noise $\mathbf{r}_i \sim N(0, 0.7^2)$ for visible, and $\mathbf{r}_i \sim N(0, 1^2)$ for infrared. At every time step i, an measurement sample $[s_i,\ t_i,\ \beta_i,\ c_i]$ is obtained. Where the source s_i, is 1 for visible, and 2 for infrared detection. Timestamp t_i is in seconds, bearing measurement β_i is calculated according to (1), and detection confidence is c_i. ΔT_i is the time step. At every i'th measurement sample, matrices $\mathbf{A}_i$, and $\mathbf{Q}_i$ are updated, and then prediction step of the Kalman filter is executed, and based on the source of bearing information from the visible or infrared, the update step is executed with suitable variance of the measurement. KF result is then smoothed with an RTS smoother. Finally, the RMS error is reported between smoother output and AIS based bearing β_0.

3 Results

Detection results of surface vessels in Baltic sea are summarized in this section. Result of YOLOv5+DeepSORT tracked detections and corresponding hand-labels on an image sequence of Viking XPRS ferry are shown in Fig. 2a and

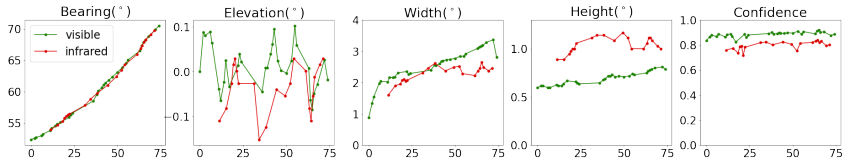


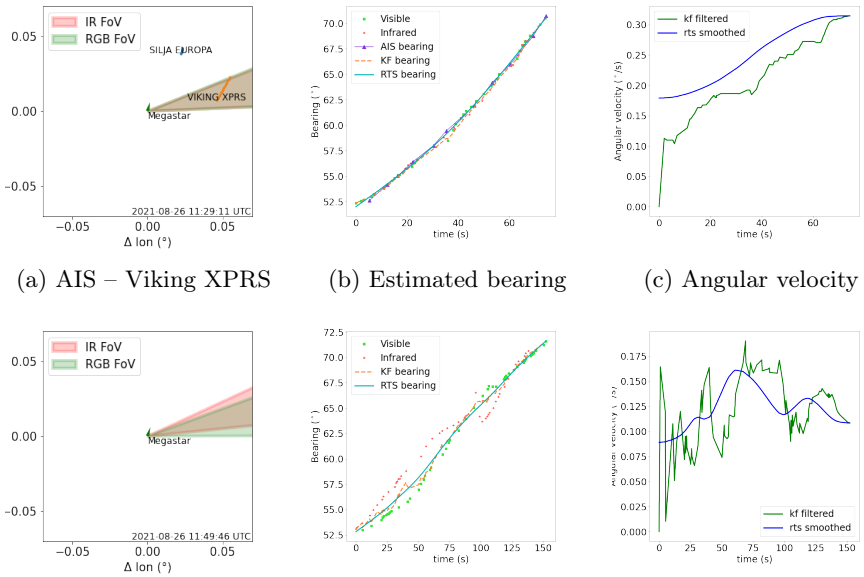
Fig. 3: Calculated bearing β and elevation ψ from (1), and bounding box width and height, and object detection confidence obtained through YOLOv5 on Viking XPRS ferry sequence, with x-axis as time in seconds.

	Base-RCNN-FPN		**YOLOv5-s**	
Imaging spectra	Visible	Infrared	Visible	Infrared
Object detection	137.2	72.2	27.7	10.1
DeepSORT tracking	33.2	33.2	22.9	22.1
Net	170.4	105.3	50.6	32.2
Net (+ KF + RTS)	274.8 (3.64 FPS)		83.0 (12.05 FPS)	
Bearing RMSE	0.42°	0.29°	0.43°	0.28°
Bearing RMSE (+ KF + RTS)	0.19°		0.17°	

Table 1: Processing time per image frame (in ms), performance in FPS, and bearing RMSE to AIS (in °) for object detection using YOLOv5-s, and a RCNN-FPN, and their DeepSORT tracking times on Tesla P100-PCIE-16GB GPU device. KF + RTS fusion took 220 μs per measurement update.

on Silja Europa ferry are shown in Fig. 2b. YOLO detector may also output multiple low-confidence false bounding box detections, which are filtered by DeepSORT to provide a single ship bounding-box trajectory and erroneous bounding-box detections are removed. Whereas, RCNN detector may provide multiple detections with strong confidence for the same object. Fig. 3 shows the YOLOv5+DeepSORT obtained bounding box azimuth or bearing, elevation, width, height, and detection confidence value for the Viking XPRS ferry. We observe that bearing from both visible and infrared is almost matched. Bounding box elevation captures the elevation of the ship relative to horizon in view. Change of object size is apparent in the YOLOv5 detector. Object detection confidence stays high during the visibility of ships in the cameras.

Fig. 4b shows the Viking XPRS bearing obtained from tracked bounding boxes. We can observe in Fig. 4b that the fusion of visible and infrared angle measurements gives an smooth estimated bearing. Using camera information allows more frequent bearing measurements than AIS. We can also obtain observed apparent angular velocity of the ship and is shown in Fig. 4c. RMSE of between KF+RTS fused bearing measurements and AIS based bearings for YOLOv5 was 0.17°, while it was 0.19° for Detectron2 Base-RCNN-FPN network. For comparison, we computed the errors with YOLOv5 + DeepSORT only which resulted in errors 0.43° for visible, and 0.28° for infrared which are both inferior to the KF + RTS fused results, and same for RCNN network based bearing output (See Table 1). Hence KF+RTS fusion has benefited the bearing accuracy. Although,



(a) AIS – Viking XPRS (b) Estimated bearing (c) Angular velocity

(d) AIS missing – Europa (e) Estimated bearing (f) Angular velocity

Fig. 4: Results of our algorithm with YOLOv5 object detection. (a) Available AIS information, (b) its fused ship bearing, and (c) angular velocity on Viking XPRS. (d) AIS information is missing, (e) its obtained bearing, and (f) its angular velocity for Silja Europa.

we take AIS as the ground truth, the maximum bearing error from the given AIS when localization error is 2.5 m and at range of 3 km is 0.1°. Hence, from our obtained error it can be said that the calibrated vision based bearing can provide accuracy close to AIS. In the case of missing AIS, for example, in Silja Europa ferry (Fig. 4d), we obtain a mismatch of few degrees between bearings obtained through visible and infrared, this however may be due to closeness of the ferry to the camera. Also a sudden shift in bearing values in infrared in Fig. 4e is due to sudden pan motion of the camera. Regardless, a fusion value is obtained representing the AIS equivalent bearing of the ship (Fig. 4e) and estimate of its angular velocity (Fig. 4f). Although, due to lack of AIS measurements, its error cannot be computed. However, this example in our experiment is exactly one of the scenarios when vision system can take over the AIS measurements for positioning due to failure of missing AIS information.

Proposed system can run in realtime, and its time performance in milliseconds for processing on Tesla P100-PCIE-16GB GPU is summarized in Table 1. RGB image size when input to the YOLOv5 is 1920×1920×3, and IR image size is 320×320×3. During bearing fusion, Kalman filter took 160 μs ms per image frame, and RTS smoother took 60 μs, and so KF + RTS fusion took

220 μs. YOLOv5 object detector performed at 12.05 frames per second (FPS), and Detectron2 RCNN implementation at 3.64 FPS for bearing estimation using DeepSORT tracking and KF + RTS fusion.

4 Conclusion

In this paper, we have presented a method for obtaining maritime vessel ship bearing by fusing information from both visible and thermal camera images. For this purpose, we have collected data from two campaigns in Baltic sea from Tallink Megastar cruise ferry. We presented an approach which utilizes object detectors with DeepSORT tracking followed by fusion of vessel bearings using Kalman filter and RTS smoother, where YOLOv5 detector outperforms Detectron2's Base-RCNN-FPN in bearing estimation. Our proposed system with YOLOv5 detector obtained RMS error of 0.17° deg for vessel bearing tracking through fusion, and is capable of real-time performance on the GPU at 12 FPS. Thus, we show the scenario of multi-sensor fusion as an application using existing methods is possible in maritime scenario and has potential use in near future.

5 Acknowledgement

We thank European Space Agency (ESA) for funding and Tallink Megastar crew for enabling the measurements onboard the vessel. We also thank Henrik Ramm-Schmidt from Fleetrange, Toni Hammarberg, and Martta-Kaisa Olkkonen from Finnish Geospatial Research Institute (FGI) for aiding in collecting the data from the campaign.

References

1. Anderson, M.: Bon voyage for the autonomous ship Mayflower. IEEE Spectrum **57**(1), 36–39 (2020)
2. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. In: Proc. IEEE ICIP. pp. 3464–3468 (2016)
3. Broström, M.: Real-time multi-camera multi-object tracker using YOLOv5 and DeepSORT with OSNet. `https://github.com/mikel-brostrom/Yolov5_DeepSort_OSNet` (Accessed 28th April, 2022)
4. Chang, L., Chen, Y.T., Hung, M.H., Wang, J.H., Chang, Y.L.: YOLOv3 based ship detection in visible and infrared images. In: IEEE International Geoscience and Remote Sensing Symposium. pp. 3549–3552 (2021)
5. Chang, L., Chen, Y.T., Wang, J.H., Chang, Y.L.: Modified YOLOv3 for ship detection with visible and infrared images. Electronics **11**(5), 739 (2022)
6. Cho, Y., Park, J., Kang, M., Kim, J.: Autonomous detection and tracking of a surface ship using onboard monocular vision. In: 12th International Conference on Ubiquitous Robots and Ambient Intelligence. pp. 26–31 (2015)
7. Farahnakian, F., Heikkonen, J.: Deep learning based multi-modal fusion architectures for maritime vessel detection. Remote Sensing **12**(16), 2509 (2020)

8. Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE international conference on computer vision. pp. 1440–1448 (2015)
9. Girshick, R., Donahue, J., Darrell, T., Malik, J.: 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 (2014)
10. Huang, H., Sun, D., Wang, R., Zhu, C., Liu, B.: Ship target detection based on improved YOLO network. Mathematical Problems in Engineering **2020** (2020)
11. Jocher, G., Stoken, A., et al.: Ultralytics/YOLOv5: v3.1. `https://doi.org/10.5281/zenodo.4154370` (Accessed 5th March, 2022)
12. Li, L., Jiang, L., Zhang, J., Wang, S., Chen, F.: A complete YOLO-based ship detection method for thermal infrared remote sensing images under complex backgrounds. Remote Sensing **14**(7), 1534 (2022)
13. Liu, T., Pang, B., Zhang, L., Yang, W., Sun, X.: Sea surface object detection algorithm based on YOLO v4 fused with reverse depthwise separable convolution (RDSC) for USV. Journal of Marine Science and Engineering **9**(7), 753 (2021)
14. Loomans, M.J., de With, P.H., Wijnhoven, R.G.: Robust automatic ship tracking in harbours using active cameras. In: Proc. IEEE ICIP. pp. 4117–4121 (2013)
15. Lu, Y., Ma, H., Smart, E., Vuksanovic, B., Chiverton, J., Prabhu, S.R., Glaister, M., Dunston, E., Hancock, C.: Fusion of camera-based vessel detection and AIS for maritime surveillance. In: 26th International Conference on Automation and Computing. pp. 1–6 (2021)
16. 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. pp. 779–788 (2016)
17. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
18. Samar, A.R., Mumtaz, A.: Enhancing performance of IR ship detection with baseline AI models over a new benchmark. In: IEEE Asia-Pacific Conference on Computer Science and Data Engineering. pp. 1–6 (2021)
19. Särkkä, S.: Bayesian Filtering and Smoothing. Cambridge University Press (2013)
20. Särkkä, S., Solin, A.: Applied Stochastic Differential Equations. Cambridge University Press (2019)
21. Teutsch, M., Krüger, W., Beyerer, J.: Fusion of region and point-feature detections for measurement reconstruction in multi-target Kalman tracking. In: 14th International Conference on Information Fusion. pp. 1–8 (2011)
22. Thombre, S., Zhao, Z., Ramm-Schmidt, H., Vallet García, J.M., Malkamäki, T., Nikolskiy, S., Hammarberg, T., Nuortie, H., H. Bhuiyan, M.Z., Särkkä, S., Lehtola, V.V.: Sensors and AI techniques for situational awareness in autonomous ships: A review. IEEE Transactions on Intelligent Transportation Systems **23**(1), 64–83 (2022)
23. Ting, L., Baijun, Z., Yongsheng, Z., Shun, Y.: Ship detection algorithm based on improved YOLO v5. In: International Conference on Automation, Control and Robotics Engineering. pp. 483–487 (2021)
24. Wang, Z., Zhou, Y.: A HED-optimized automatic detection and tracking algorithm for marine moving targets based on YOLO v3. In: Journal of Physics: Conference Series. vol. 1449. IOP Publishing (2020)
25. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: Proc. IEEE ICIP. pp. 3645–3649 (2017)
26. Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., Girshick, R.: Detectron2. `https://github.com/facebookresearch/detectron2` (Accessed 13th November, 2022)