

Deep Learning based Object Detection and Tracking for Maritime Situational Awareness

Rihab Lahouli^a, Geert De Cubber^b, Benoît Pairet^c, Charles Hamesse^d,
Timothée Fréville and Rob Haelterman^e

Royal Military Academy, Mathematics Department, Renaissanceaan 30, 1000 Brussels, Belgium

Keywords: Object Detection, Tracking, Situational Awareness, Maritime Dataset.

Abstract: Improving real-time situational awareness using deep-learning based video processing is of great interest in maritime and inland waterway environments. For instance, automating visual analysis for the classification and interpretation of the objects surrounding a vessel remains a critical challenge towards more autonomous navigational system. The complexity dramatically increases when we address waterway environments with a more dense traffic compared to open sea, and presenting navigation marks that need to be detected and correctly understood to take correct decisions. In this paper, we will therefore propose a new training dataset tailored to the navigation and mooring in waterway environments. The dataset contains 827 representative images gathered in various Belgian waterways. The images are captured on board a navigating barge and from a camera mounted on a drone. The dataset covers a range of realistic conditions of traffic and weather conditions. We investigate in the current study the training of the YOLOv5 model for the detection of seven different classes corresponding to vessels, obstacles and different navigation marks. The detector is combined with a pretrained Deep Sort Tracker. The YOLOv5 training results proved to reach an overall mean average precision of 0.891 for an intersection over union of 0.5.

1 INTRODUCTION


Deep learning algorithms and precisely CNNs (Convolutional neural networks) have shown to be a powerful tool for visual analysis and understanding tasks such as classification, object detection, and tracking (A. Khan and Qureshi, 2020)(G. Ciaparrone and Troiano, 2020). The great progress in these models promises to computationally generate an advanced intelligence for the detection of collision threats and recognition of navigation marks without significant need for human intervention (Z. Chen and Cheng, 2020)(B. Iancu and Liliu, 2021).


In this paper, we report our work in the scope of the SSAFE project (Shared Situational Awareness between Vessels), funded by the Flemish Agency for Innovation and Entrepreneurship. The goal of the project is to build a shared dynamic semantic map for


surveillance purposes. Several sensors are used in the project and we are in charge of the visual camera processing and useful information extraction to consolidate the real-time situational awareness system.


Our work consists of implementing a CNN-based solution able to generate safety alerts in order to avoid collisions; it can also support decision making in a maritime environment. The proposed solution intends to enhance monitoring and safety in waterway environments. In recent years, interesting open-source maritime datasets have been realized and used for the training of CNN models which has provided an important contribution in the field of image segmentation in addition to ships and obstacle detection in the maritime environment. We can cite the Singapore Maritime dataset (DK. Prasad and Rajabally, 2017) that was created around Singapore waters, the MaSTr1325 dataset with 1325 diverse images and the MODD2 dataset captured both in the gulf of Koper "Slovenia" (B. Bovcon and Kristan, 2019)(Borja and Kristan, 2020).


These datasets are mostly realized in an uncrowded open-sea which make using them for our application irrelevant. In fact, in our project we address

^a  <https://orcid.org/0000-0002-0883-8225>

^b  <https://orcid.org/0000-0001-7772-0258>

^c  <https://orcid.org/0000-0002-8731-033X>

^d  <https://orcid.org/0000-0002-2321-0620>

^e  <https://orcid.org/0000-0002-1610-2218>

waterways navigation zones with high risk of collision and with the presence of several maritime marks required for mooring assistance and priorities indication. To address this issue, we conducted measurement campaigns to collect our custom SSAVE dataset tailored to the training of the relevant classes that we need to detect and keep track of, aboard a vessel. Our main contribution is a new diverse training dataset addressed for waterways maritime environment captured by cameras mounted in a drone and a moving barge in realistic weather and traffic conditions. We investigate also the training of the YOLOv5 model for the detection of seven different classes (ship, barge, cutter, yellow mark, red mark, line mark and other obstacle) for situational awareness purposes. This paper is organized as follows: Sect.2 describes the collected SSAVE dataset and the classes that we considered for the semantic annotation; in Sect.3 distortion correction of a part of the dataset is presented. In Sect.4, the results of YOLOv5 detector training are presented and evaluated. And finally Sect.5 presents inference results of combining the detection with a Deep Sort pre-trained tracker; also it gives conclusions and discusses possible further research perspectives.

2 MARITIME WATERWAY DATASET: SSAVE

Maritime image datasets are essential for the training of neural networks performing object detection and tracking. They should present an important variety of weather and lighting conditions; also, the images should be taken with different angles of view.

2.1 SSAVE Dataset

The SSAVE dataset was realized in collaboration with our industrial partners (Deme and Tresco) in the project and with the Belgian naval base at Zeebrugge. We collected thousands of images with high definition (1080x1920 pixels) in realistic conditions of traffic and in different weather conditions such as sunny, cloudy and rainy days. We used AXIS Q3515-LV Network IP cameras mounted on a navigating barge in addition to GoPro Hero8 cameras placed on a drone and also attached to a navigating barge. Recording images from a drone was essential to detect the navigating barge itself and also to have a better view of the line marks. We hand-picked 827 representative images out of the gathered footage to construct our dataset. The variety in terms of luminosity and distances separating the barges to the obstacles in addition to the presence of objects of interest with close

proportions have been considered in the selection of the images constructing the dataset. The dataset is composed of 78 images taken from a GoPro Hero 8 camera mounted on a drone, 175 images taken from a similar camera mounted on a navigating barge and 574 images taken from an AXIS Q3515-LV Network IP camera attached to a navigating barge. Indeed, the dataset is tailored to the scenarios addressed by the study corresponding to the navigation of a barge in a waterway and mooring into a fixed platform called a cutter. We annotated manually each image in the dataset using CVAT image annotator. Seven classes were considered in the annotation (ship, barge, cutter, yellow mark, red mark, line mark and other obstacle). Figure 1 presents an example of annotated images. For instance, we need to detect and track all ships surrounding the navigating barge in addition to all types of obstacles that can present collision threats. Navigation marks need to be detected and classified in order to respect priorities and interdiction areas while navigating and mooring, also the navigation marks can be subject to collisions. Finally the cutter should be detected and kept in track in order to assist the barge in mooring properly. We did not proceed to augmentation techniques to increase the dataset size.

2.2 Preprocessing of Distorted Images

Some of the collected images (about 100) presented radial distortion. To remedy to this error, we first calibrated the camera to identify its parameters using the Matlab camera calibrating tool and a checkerboard pattern. Then, we used the `UndistortImage` function of Matlab to rectify the lens distortion and therefore convert distorted images into undistorted ones. Figure 2 presents an example of an image before and after distortion correction.

3 OBJECT DETECTION AND TRACKING METHODS

This section presents a brief description of the deep learning models used for the detection and tracking tasks. Also, the evaluation metrics used for performance measurement are defined.

3.1 YOLOv5 Detection

Object detection is one of the fundamental computer vision problems that enables semantic video analysis and image understanding. Its concept consists in identifying precisely in an image the presence and the



Figure 1: Examples of annotated images.



Figure 2: Image with radial distortion at left and after correction at right.

location of objects defined in a pre-established list (PF. Felzenszwalb and Ramanan, 2009).

The “You only look once (YOLO)” detector has been proposed by (J. Redmon and Farhadi, 2016) as a novel approach for object detection that makes use of a unified framework to predict both classes confidence and bounding box coordinates at the same time. The basic idea of YOLO is to divide the input image into a grid and compute the bounding boxes and the probabilities for each cell grid. The predicted confidence scores are used to weigh these bounding

boxes. The algorithm performs only one forward propagation pass through the neural network to make predictions, so it “only looks once” at the input image. YOLOv5 is a recent update of the YOLO object detection family. It has been released in 2020 and it was developed with the Pytorch framework. The YOLOv5 detector is very fast, permitting to perform real-time object detection with high accuracy (Jocher, 2021). The YOLO object detection algorithms have been used for vehicle detection tasks (Fachrie, 2020) (M. Kasper-Eulaers and Sebulonsen, 2021) and they have proven to outperform other state-of-the-art CNN algorithms for object detection, such as Faster R-CNN, in sensitivity and processing time with a comparable precision (B. Benjdira and Ammar, 2019). These attributes led us to the selection of the YOLOv5 object detection algorithm for the training of the SSAVE dataset to detect the classes of interest in the waterway navigation environment. Decreasing the processing time gains a significant importance for real-time detection and traffic monitoring.

3.2 Deep Sort Tracker

Tracking objects in a video at real-time speeds is a powerful tool for monitoring applications. Deep Sort is a CNN architecture developed using the Pytorch framework. It tracks objects along video frames given their bounding boxes predicted first by a detector. It was used for pedestrian tracking (N. Wojke and Paulus, 2017). In our study, the detection generated by YOLOv5 are passed to the Deep Sort algorithm to keep track of the detected classes in the video frames.

3.3 Performance Evaluation

The performance evaluation for an object detector is typically assessed by three metrics that we explain briefly in this sub-section (I. Goodfellow and Courville, 2016).

- Intersection over union (IoU): which measures the overlap degree between the predicted bounding boxes and the ground truth ones. It is a ratio between the area of overlap between the two bounding boxes divided by the area of union between both of them. For a specific IoU threshold we can measure the accuracy of the predictions by computing the precision recall metrics.
- Precision-Recall: which measures the trade-off between the precision, being the percentage that your predictions are correct and the recall which measures how well you find all the positive cases over your predictions as presented by the follow-

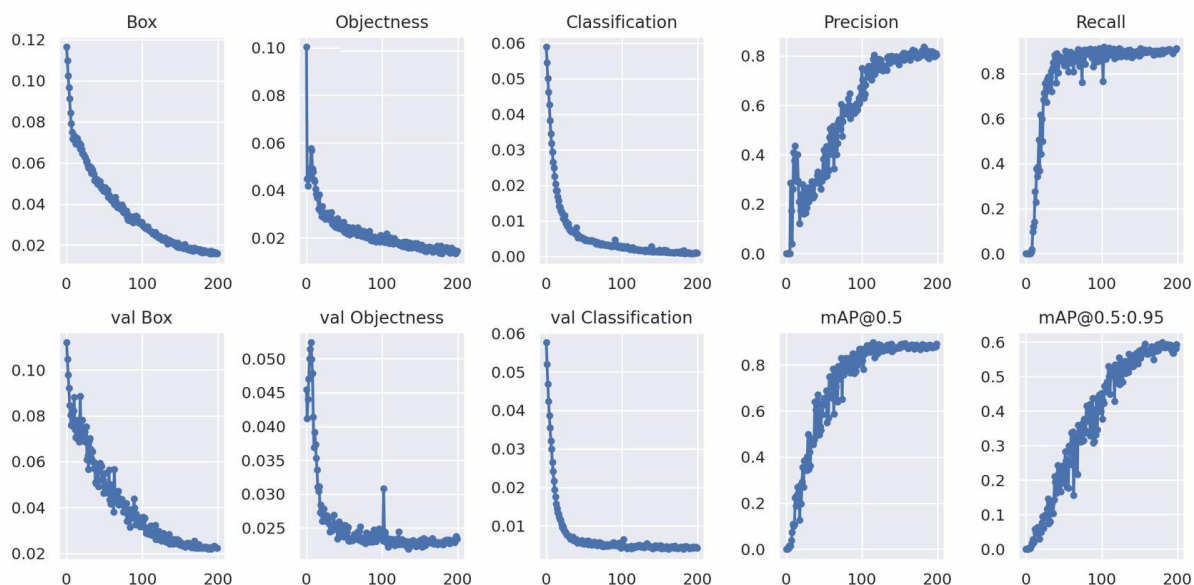


Figure 3: Plots of box loss, objectness loss, classification loss, precision, recall and mean average precision (mAP) over 200 training epochs for the training and validation set.

ing formulas.

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

Note that, TP is the True Positive, FP is the False Positive and FN is the False Negative all measured over the whole prediction of the CNN model. The relationship between precision and recall can be observed using a plot.

- Mean Average Precision (mAP): Given a fixed IOU threshold, the mean precision of each predicted class is calculated then averaged with all the classes precision to return the Mean Average Precision for the dataset.

4 YOLOV5 OBJECT DETECTION TRAINING USING THE SSAVE DATASET

In this section, we investigate the performance of the YOLOv5 detector using the SSAVE dataset.

4.1 Implementation details

All the implementations are performed under the deep learning framework Pytorch Ultralytics YOLOv5. We use the docker implementation on an NVIDIA DGX Station A100 with four fully interconnected NVIDIA

A100 Tensor Core GPUs and up to 128 GB of total GPU memory. We use the YOLOv5 model pretrained on the Common Objects in Context (COCO) dataset (TY. Lin and Hays, 2017) to initialize the network and extract features from the input images. Then we prepare our custom SSAVE dataset with 827 images by splitting it into training, validation and test data with percentage of 70%, 20% and 10% respectively. The computational cost of training the model for 200 epochs takes about 89 minutes.

Table 1: Detection precision per class.

Classes	Precision
ship	0.877
barge	0.995
cutter	0.940
yellow_mark	0.986
red_mark	0.887
line_mark	0.842
other obstacle	0.713
all classes	0.891mAP@0.5

4.2 Experimental Analysis

Figure 3 shows the evolution of different performance metrics over the 200 training epochs for both the training and validation sets. There are three losses that the model should minimize to achieve a detection, location and classification convergence. The box loss represents how well the predicted bounding box covers an object. Objectness loss refers to the prob-

ability that an object exists in a proposed region of interest. The classification loss represents how well the algorithm can predict the correct class of a given object. We notify a rapid decline of the box, objectness and classification losses until around 150 epochs. The model also showed a rapid improvement in terms of precision, recall and mean average precision. It achieved a 0.891 mAP at 0.5 IOU as shown by Figure 4. We also see a high precision between 0.842 and 0.995 for all the classes except the “other obstacle” class where the precision reaches only 0.713 as resumed in Table 1. This is explained by the fact that this class includes objects with different forms going from poles to small obstacles that can not be included in the other categories but they need to be detected in order to notify the navigating vessel about a collision threat. Figure 9 presents the qualitative performance of prediction of the new trained YOLOv5 model applied to a test batch for detecting the seven classes (ship, barge, cutter, yellow mark, red mark, line mark and other obstacle).

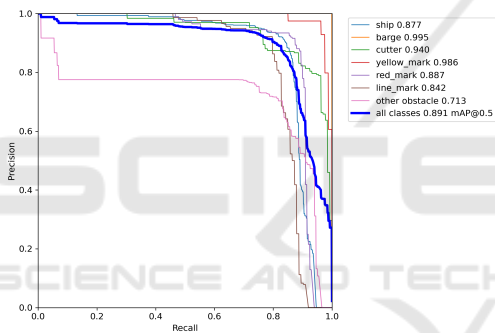


Figure 4: Precision recall curve.

4.3 Discussion

Figure 5 shows the proportion distribution of the seven annotated classes into the training dataset. “Ship” category is the most presented with more than 770 bounding boxes. On the other hand, the barge category is the least represented. This is due to the fact that most images have been captured using the camera attached to the barge. Thus the barge was present only in images captured by the drone. therefore the classes are unbalanced in the dataset and this problem can be considered in future work to further enhance the detection of under-represented classes. The low precision of the “other obstacle” class led to some false negative detection as shown by Figure 6 where some obstacles present in the image were not detected although detection with a confidence score of 0.55 has been done for a neighbour obstacle. Also the vegetation background in the waterways led to some false positive detection as presented by Figure 7 where the

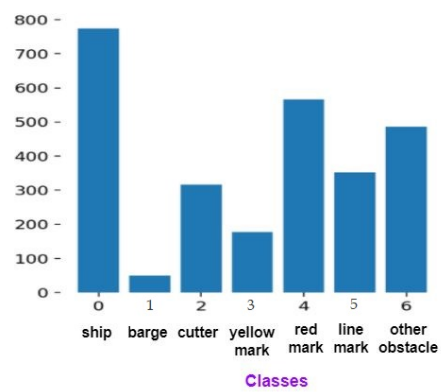


Figure 5: Proportion of each annotated class in the training dataset.



Figure 6: False negative detection for “other obstacle” class.

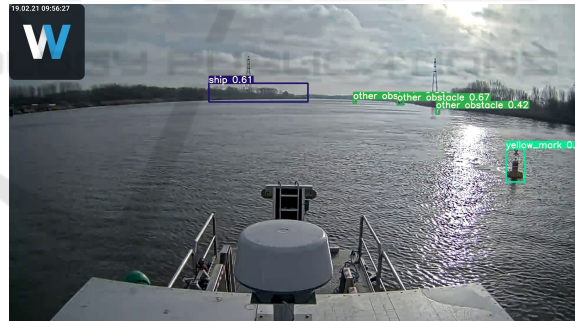


Figure 7: False positive detection for “ship” class.

detector has assumed by mistake a ship in the vegetation. In the other hand, Figure 8 presents example results of correct detection for all the objects present in the images. Data augmentation techniques are also commonly used to increase the data size and variance by generating geometric perturbations, Gaussian blur and noise (B. Zoph and Ghiasi, 2020). We suggest to investigate the capability of augmentation strategy to enhance the detection performances.



Figure 8: Correct detection results.

5 CONCLUSIONS

In this paper, we introduced a new dataset tailored to the training of object detection CNN models for situational awareness purposes in the maritime waterway environments. Up to 827 representative images with high variety have been selected manually from video recording in real scenarios of navigation and mooring. Ip cameras attached to a navigating barge in addition to a camera attached to a drone have been used to ensure a variety in the angles of view of the objects to detect. Seven different classes have been annotated manually which are; ship, barge, cutter, yellow mark, red mark, line mark and other obstacle. The main concern was about the implementation of a CNN-based object detection algorithm able to alert the vessels about the presence of collision threats in the neighborhood in addition to the detection and the correct interpretation of the navigation marks present in the waterway in order to help the vessel taking the right

decisions. We chose to train the YOLOv5 object detection model due to its better performances in terms of processing time with similar precision compared to other state-of-the-art object detection models. We have trained the YOLOv5 using our dataset and we obtained successful results achieving a mean average precision of 0.891 at an intersection over union of 0.5. The precision of predictions by classes was also high reaching 0.995 and not less than 0.713. Qualitative results corresponding to prediction results on test images have been presented. The detection generated by YOLOv5 has been passed to the pretrained Deep Sort algorithm and we tested the results on real videos of a navigating barge and the results were satisfying. For future work, we suggested to investigate into increasing the presence of the least representative classes in the dataset in order to deal with unbalanced classes. Also we suggested to test some common augmentation techniques on the SSAVE dataset and analyse if we can achieve better results.

ACKNOWLEDGEMENTS

The research presented in this paper has been funded by the Flemish Agency for Innovation and Entrepreneurship (VLAIO), project SSAVE.

REFERENCES

- A. Khan, A. Sohail, U. Z. and Qureshi, A. (2020). A survey of the recent architectures of deep convolutional neural networks review. *Artificial Intelligence Review*, 53:5455–5516.
- B. Benjdira, T. Khursheed, A. K. and Ammar, A. (2019). Car detection using unmanned aerial vehicles: Comparison between faster r-cnn and yolov3. In *2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS)*. IEEE.
- B. Bovcon, J. Muhović, J. P. and Kristan, M. (2019). The mastr1325 dataset for training deep usv obstacle detection models. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE.
- B. Iancu, V. Soloviev, L. Z. and Lilius, J. (2021). Aboships-an inshore and offshore maritime vessel detection dataset with precise annotations. *Remote Sensing*, 13(5):988.
- B. Zoph, E. C. and Ghiasi, G. (2020). Learning data augmentation strategies for object detection. In *ECCV*.
- Borja, B. and Kristan, M. (2020). A water-obstacle separation and refinement network for unmanned surface vehicles. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Computer Science.
- DK. Prasad, D. Rajan, L. R. and Rajabally, E. (2017). Video processing from electro-optical sensors for object de-



Figure 9: Images from the test dataset showing the performance for predicting the seven classes; ship, barge, cutter, yellow mark, red mark, line mark and other obstacle.

tection and tracking in a maritime environment: a survey. *IEEE Transactions on Intelligent Transportation Systems*.

Fachrie, M. (2020). A simple vehicle counting system using deep learning with yolov3 model. *Journal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 4:462–468.

G. Ciaparrone, F.L.Sánchez, S. T. and Troiano, L. (2020). Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 381:61–88.

I. Goodfellow, Y. B. and Courville, A. (2016). *Deep Learning*. MIT Press. <http://www.deeplearningbook.org>.

J. Redmon, S. Divvala, R. G. and Farhadi, A. (2016). You only look once: Unified, real-time object detection.

Computer Vision and Pattern Recognition.

- Jocher, G. (2021). Yolov5 in pytorch. <https://github.com/ultralytics/yolov5>.
- M. Kasper-Eulaers, N. Hahn, S. B. and Sebulonsen, T. (2021). Short communication: Detecting heavy goods vehicles in rest areas in winter conditions using yolov5. *Algorithms*, 14:114.
- N. Wojke, A. B. and Paulus, D. (2017). Simple online and realtime tracking with a deep association metric. In *Computer Vision and Pattern Recognition*. Computer Science.
- PF. Felzenszwalb, RB. Girshick, D. M. and Ramanan, D. (2009). Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645.
- TY. Lin, M. Maire, S. B. and Hays, J. (2017). Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer.
- Z. Chen, D. Chen, Y. Z. and Cheng, X. (2020). Deep learning for autonomous ship-oriented small ship detection. *Safety Science*, 130:104812.

APPENDIX

The SSAVE dataset has been made publicly available at:

<https://mecatron.rma.ac.be/index.php/publications/datasets/>

