REAL-TIME OBSTACLE DETECTION FOR USVs ON EMBEDDED GPU

A thesis submitted in partial fulfilment

of the requirements for the award of the degree of

Bachelor of Technology
In
Mechanical Engineering

Submitted by

Kovid Sharma
21ME01026

Under the Supervision of
Dr. Yogesh Bhumkar and Dr. Debi Prosad Dogra

School of Mechanical Sciences

Indian Institute of Technology, Bhubaneswar

DECLARATION

I certify that,

1. The work contained in this thesis is original and has been done by me under the guidance and supervision of Name of the supervisor.
2. The work has not been submitted to any other institute for any Degree or Diploma.
3. I have followed the guidelines stipulated by the institute in preparing the Thesis.
4. I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the institute.
5. Whenever I have used materials (data, theoretical analysis and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the References.

Kovid Sharma

Real-time Obstacle detection for USVs on Embedded GPUs

Abstract

Obstacle detection is crucial for unmanned surface vehicles (USVs). The USVs function in varied situations where obstacles can include floating debris, individuals, or coastline parts, posing a considerable challenge for ongoing detection from onboard cameras. A real-time USV-based obstacle detection system for Indian scenarios is required for the USV to perform a rescue operation in oceans and flood-affected regions. The research in this field is significantly lagging, especially in Indian scenarios, due to the unavailability of large annotated datasets. We plan to manufacture a USV to construct a vision dataset along with the additional data using sensors like LiDAR. We plan to create an obstacle detection system that can be performed in real-time using onboard embedded GPUs. A multi-modal approach can be used to perform segmentation and detection separately. An Embedded GPU is used to test multiple models and strategies for real-time inference.

Keywords: Unmanned Surface Vessels(USVs), Real-time Obstacle detection, maritime navigation, Embedded GPUs

Table of Contents

Contents	**Page No.**

List of Figures

Fig. No.: Caption **Page No.**

List of Tables

CHAPTER 1: INTRODUCTION

1.1 Theoretical background

Recent advancements in marine robotics have resulted in the development of a new kind of small unmanned surface vessels (USVs). These vessels can be navigated either through manual control or by being programmed to adhere to a specified trajectory. USVs can vary in dimensions according to the application, but typically, they measure less than 2 meters in length. USVs have the advantage of portability and relative ease of navigation in shallow waters and constricted marinas. However, the compact design of such USVs constrains the type and placement of sensors and computation payload, which is additionally limited by power consumption restrictions. The size restrictions also prevent the use of additional stabilisers.

The USV, operating on the water's surface, must independently perform multiple duties, necessitating the inclusion of situational awareness through environmental sensing to identify obstructions. The shoreline can serve as a primary obstruction, which may be managed to some degree by the utilisation of precise maps and satellite navigation. In most cases, and especially in the Indian scenario, these maps are not detailed and usually unavailable.

Fig.1 Difference between the existing dataset and Indian conditions.
Image Credits: https://danandholly.com/2021/01/indian-fishing-boats/

The research in USVs is still lagging due to the unavailability of a large publicly available annotated dataset. Research on USVs suffers from the lack of a substantial publicly accessible annotated dataset. Recently, numerous datasets have been developed for the USVs for the obstacle detection task. However, no publicly available dataset deals with the Indian waters. The model trained on the existing datasets is expected to perform poorly in Indian Scenarios due to a significant difference in the climate, coast conditions and camera specifications. Fig. 1 shows the difference between the existing foreign dataset and Indian conditions.

1.2 History of development

The obstacle detection problem was previously tackled in the domain of Unmanned Ground Vehicles(UGVs). Due to the more diverse conditions in the maritime environment, the majority of the models prepared for UGVs cannot be directly applied to the USVs. The obstacle detection in USVs majorly contains two tasks. First, to detect the shoreline and the sea-sky line. Second, to detect the obstacles floating and protruding from the water. The shoreline detection can be avoided by using maps. [1] employed an aerial image of the region sourced from Google Maps to create a map of static obstacles. However, this strategy is inadequate for managing a more challenging category of dynamic obstacles that do not appear on the map. Since USVs are built on a small form factor, the use of extremely power-hungry sensors like RADAR and SONAR cannot be used. Cameras are increasingly recognised as lightweight, low-power, and information-dense sensors, serving as a feasible alternative or complement to other sensor technologies [2, 3].

Many researchers used monocular and stereo camera systems with technologies like pixel profile analysis, construction of disparity maps, distributed Gaussian components, Markov random fields, etc., to detect shorelines and obstacles. Most of these approaches assumed the shoreline to be straight and used horizon lines to detect shorelines. These approaches performed poorly near the coast.

Many researchers started using semantic segmentation and graphical models to detect obstacles, especially shorelines, which became a preferred methodology for this task due to its exceptional performance. Further, the researchers fused data from sensors and visual information from the camera to improve the performance. Recently, researchers have developed sophisticated models and have constructed large annotated datasets for further improvement in this field. However, the research on USVs in Indian scenarios is still lagging due to the lack of publicly available large annotated datasets.

1.3 Applications

USVs are commonly employed for perimeter monitoring, navigating along a predetermined route. They are also utilised for rescue operations and the identification of abnormal marine activities near ports[4]. Functioning on the surface of the water, the USV may be required to complete various tasks autonomously. An autonomous system requires onboard GPUs to perform excessive computational tasks in real-time. We plan to develop a USV with onboard computation capabilities specifically for rescue operations, but it can be employed on other applications as well.

CHAPTER 2: Literature Review and Formulation

2.1 Related Work

An obstacle detection problem has been previously tackled in the domain of unmanned ground vehicles (UGV), but the study field of unmanned surface vehicles is still in its early stages. Thus, there is substantial literature on the topic of obstacle detection and avoidance for UGVs.

Krotosky et al. [5] perform obstacle detection by using a stereo camera to compute a disparity map. Zhang et al. [6] and Cao et al. [7] also employ a similar methodology. Each of these approaches utilises distinct techniques to filter the disparity map and noise in the detected obstacles. Lu and Rasmussen[8] presented obstacle detection as a labelling challenge. They use a set of pre-trained classifiers to 3D point clouds and a Markov random field to consider the spatial smoothness of the labelling. Shim et al. [9] use LiDAR to detect general obstacles on the road with the assumption of the ground being a flat surface. Using a monocular camera and combining Histogram of oriented gradients(HOG) and Support Vector Machine algorithms, they identify pedestrians and vehicles by restricting the search region to bounding boxes of previously acquired detections from the LiDAR. Most methods designed for autonomous land vehicles depend on estimating the ground plane and are not easily applicable to the underwater environment of the USVs. Furthermore, obstacles in the marine environment have a more diverse appearance than on the road, and in addition, even small, submerged obstacles might present a significant threat to the USV. Moreover, obstacles in the maritime environment exhibit greater diversity than those on roadways, and even little submerged obstacles may pose a substantial risk to the USV.

A prevalent method for detecting obstacles in marine environments is the utilisation of RADAR, SONAR, or LIDAR. The compact dimensions and high power consumption of radar devices pose an extra challenge when used on small-sized USVs. Also, the onboard radar encounters difficulties in detecting small obstacles situated close to the boat[10]. LiDAR faces difficulty in bad weather conditions, while the other sensors are less accurate. Hence, cameras combined with computer vision algorithms might work as an effective alternative. Larson et al. [11] developed an obstacle detection system with a monocular camera. It depends on horizon estimation and image segmentation. Horizon was estimated using trigonometric calculations in the open sea and nautical charts adjacent to the coast. Edge detection approaches have been used to detect the horizon line[12]. The method described in [12] considers the sea edge as a straight line, which is often incorrect in the case of coastal waters. This method depends on a distinct border between the sea and sky to determine the horizon line. Nevertheless, in practical scenarios, this barrier is frequently obscured by severe weather conditions.

Wang et al. [13, 14] used pixel profile analysis and RANSAC regressor to estimate the sea-sky line. The method proposed in [13] relies on the fact that the brightness of the sky is generally much different from that of the sea area. Along with the sea-sky line, they used saliency detection, Harris corner extraction and tracking for object detection by a monocular camera. For a stereo camera system, they used saliency detection in the left camera to detect obstacles. Then, in the next step, they use Epipolar Constraint and template matching to search for correspondences of detected obstacles in the right camera. To further improve the position of the correspondences, disparity filtering is introduced[14]. Cane et al. [15] used saliency maps with wake and glint suppression for obstacle detection and tracking. They attained exceptional results on the open-seas dataset [16]. Many of these approaches approximate the edge of the water by the horizon line, which has a significant drawback. In coastal scenarios, the water's edge no longer aligns with the horizon and cannot be represented as a straight line. This necessitates a more generalised segmentation approach.

Kristan et al. [17] introduced a graphical model for monocular obstacle detection by structurally constrained semantic segmentation of the observed maritime environment. This algorithm treats all objects in the water region as obstacles and generates a water segmentation mask. This model partitions an image into three distinct and approximately parallel semantic regions: sky, ground and water. To generate the semantic structure, vertically distributed Gaussian components are fitted, and their result is regularised with a Markov random field. The model effectively detects obstacles that protrude from the surface, as well as floating obstacles. It does not presume a straight water edge and operates in real-time. However, it remains ineffective in the presence of visual discrepancies such as haze. Along with the model, they also constructed a large annotated dataset called the Marine obstacle detection dataset(MODD). It contains 4454 fully annotated images with a resolution of 640 x 480 pixels.

[18] proposed a new segmentation-based detector based on [17]. They included pitch and roll measurements from the IMU to project the horizon onto the input image. They derived a practical IMU-to-camera calibration and the horizon projection equations. They improved the segmentation-based obstacle detection by introducing stereo verification. They constructed a new challenging dataset called Marine Obstacle Detection Dataset 2(MODD2). It contains multiple video sequences with time-synchronized data streams from IMU and GPS. The video sequences total 11675 frames at a resolution of 1278 × 958 pixels.

[19] introduced a new dataset called MaSTr1325, which contains 1325 diverse images which are per-pixel semantically labelled. They also proposed a dataset augmentation protocol to address slight differences in the images. The dataset surpasses prior efforts in this domain in size, scene complexity, and domain realism.

[20] showed that the existing methods poorly estimate the water edge in the presence of visual ambiguities, poor detection of small obstacles and high false-positive rate on water reflections and wakes. To address these issues, they proposed a new encoder-decoder architecture. They improved the accuracy by fusing inertial information from IMU with the visual features from the encoder. In addition, they introduced a novel loss function to increase the separation between water and obstacle features early on in the network.

[21] propose a large unified dataset to estimate depth maps and detect obstacles as entities benchmark -MODS, designed for evaluation of classical object detection methods as well as segmentation-based obstacle detection methods. The datasets introduced in [17, 18, 19, 21] and their annotations are publicly available.

[22] introduced a vision-based framework for target tracking in extreme conditions with highly reduced visibility in a coastal environment. The framework consists of a generative adversarial network(GAN) model for image de-hazing and de-noising of the image. They used a YOLOv5-based object detection system to detect objects from enhanced images. Using object detection for obstacle detection is a rare approach.

The current deep learning-based object detectors are divided into two broad categories: one-stage and two-stage detectors. Typically, the two-stage[23-25] approach attains a high level of accuracy but lacks efficiency, making it challenging to execute on a UAV platform with restricted computational resources. They employ a region-proposal network to ascertain whether the previous anchors correspond to an object or background. The previous anchors consist of multiple explicitly defined potential bounding boxes. Then, they employ two head networks to categorise potential anchors and estimate the offset between the anchors and ground truth boxes. One-stage detectors[26, 27] eliminate the region proposal network completely. The categories and offsets of the past anchors are predicted directly using two detectors. A novel category of detectors, known as anchor-free detectors[28, 29], has been recently introduced. Their approach involves simplifying the bounding box prediction to the key point and size estimation. It presents an improved method for detecting objects with various scales.

2.2 Literature Gap

Most of the modern approaches used for obstacle detection are segmentation-based, which makes it tough to implement them on Embedded GPUs due to power and size restrictions. In addition, the accuracy of the existing approaches will significantly vary when employed in Indian coastal scenarios. Therefore, there exists a significant gap in using deep learning-based object detectors for obstacle detection tasks. Furthermore, there is no publicly available USV-based obstacle detection dataset for Indian scenarios, which calls for the generation of an annotated dataset for USV-based obstacle detection.

2.3 Objectives

With this research, we plan to complete three objectives. Our first objective is to manufacture a USV with a camera and LiDAR with IMU sensor calibration. Our second objective is to explore and experiment with existing obstacle detection systems in Indian scenarios. Conducting experiments with existing approaches is crucial to assess their effectiveness in Indian situations. Our final objective is to create an annotated dataset for Indian scenarios and to develop a robust real-time obstacle detection system that can work on an embedded GPU. The use of an onboard embedded GPU for detection is crucial as it improves the operational range and enables significantly faster performance.

2.4 Methodology

The current approach finalised for obstacle detection is to use a multi-modal system, a segmentation model for detecting the water edge and an object detection model to detect obstacles. Models will be selected based on their real-time inference capabilities. While the USV is under development, we plan to use the MODS[21] dataset to explore and experiment with custom models. We plan to experiment with multiple modern segmentation and object detection models as well as with the existing models.

The USV under-development is inspired by the rescue water drone developed by Saif Seas, the demo of which can be seen in the video[4]. The USV is being developed for two main reasons. First, to develop annotated datasets for Indian coastal regions to boost research in this domain. Second, to create an obstacle detection system specifically for rescuing people drowning in oceans, rivers and people affected by floods. The CAD model of USV is shown in Fig. 2.

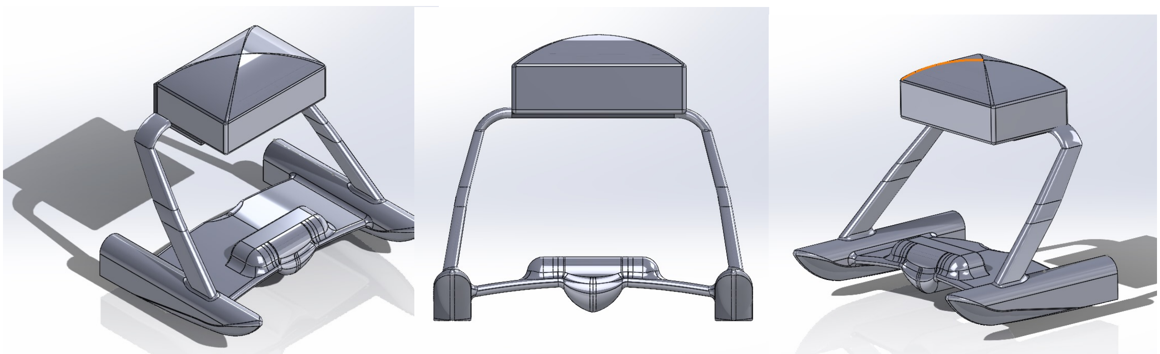

Fig.2 - CAD model of the USV

The USV is expected to be equipped with sensors like a camera along with a gimble system, LIDAR, GPS, etc. and an Embedded GPU. The embedded GPU we expect to use is Nvidia Jetson Orin AGX[30]. We plan to fuse visual information with the outputs of other sensors using IMU-sensor synchronisation and camera calibration. After the development of the USV, we plan to test it on multiple locations and create an annotated dataset for obstacle detection.

To begin with the experimentation, we plan to use U-Net[31] as a segmentation model for the water edge detection and YOLO series[26] and FCOS[28] as object detection models. In future, we plan to create sophisticated models to achieve good accuracy in both tasks in real-time on an Embedded GPU. In the future, we intend to develop advanced models to attain high accuracy in both tasks in real-time on an embedded GPU.

CHAPTER 3: EXPERIMENTATION DETAILS AND MODEL DESCRIPTION

3.1 Dataset Modification

We utilised the MODS dataset[21] and modified it for a binary segmentation task for water edge detection. The annotations for the dataset were available as a JSON file, which consisted of an image path, sensor data, and water edge coordinates. The binary Mask of an image was created using OpenCV by first creating a blank image and then drawing a water edge using the cv2.line method with a thickness of 5 pixels. An example of an image and mask is shown in Fig. 3. The image was further resized to 640 × 640 pixels for training. The segmentation dataset contains 8175 images.

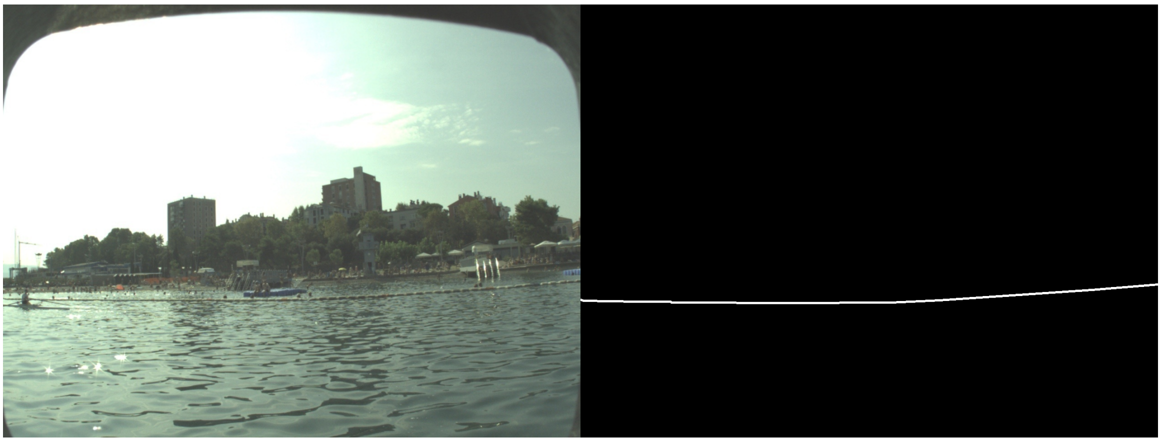

Fig. 3 - Example of an image and its respective mask

3.2 Model Description

The model proposed for the segmentation task is U-Net[31]. U-Net is a completely Convolutional Neural Network Architecture developed specifically for biomedical image segmentation. However, it has shown exceptional performance on other segmentation tasks as well. As shown in Fig. 4, the model extracts deep features from an image in the encoder part. The Decoder part

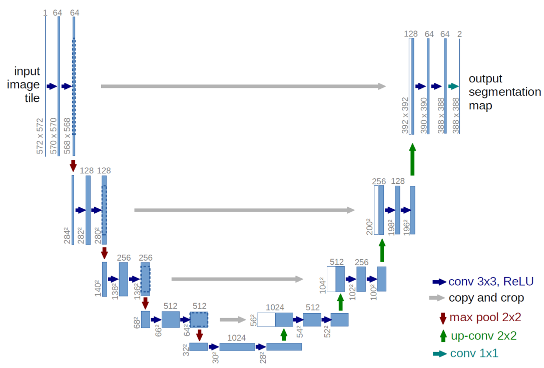

Fig. 4 - U-Net Architecture

uses transpose convolutions and residual connections to output the mask. We modified the existing architecture by increasing the input image resolution and then adding a BatchNorm[32] layer to get better results. The model is created by using the PyTorch library and Python programming language. The final model contains 118,577,922 trainable parameters.

3.3 Experimentation Details

The model was trained parallelly on two NVIDIA T4 GPUS provided by Kaggle, each of them containing 16 GB of GPU memory. The loss function used is Binary cross-entropy with logits loss. This loss combines a Sigmoid layer and the BCELoss in one single class. This version is more numerically stable than using a plain Sigmoid followed by a BCELoss by combining the operations into one layer[33]. The optimiser used for training is the Adam optimiser [34, 35]. Data augmentation like Rotation, HorizontalFlip, and Vertical Flip were applied using the albumentations library[36]. The training was done for 25 Epochs with different learning rates, as shown in Table 1. The complete training took nearly 4 hours.

Table-1: Learning Rate Details

Epoch No.	Learning Rate
1-10	0.001
11-15	0.0001
16-20	0.00001
21-25	0.000001

CHAPTER 4: RESULTS AND DISCUSSIONS

4.1 Observations

The following observations were made:

1. The model achieved 99.5% accuracy and 0.68 as the final dice score.
2. The final dice score increased by lowering the learning rate in the last epochs.
3. The input image, predicted mask and ground truth mask are shown in Fig. 5.
4. From Fig. 5 we can say that our model performed significantly well on the dataset

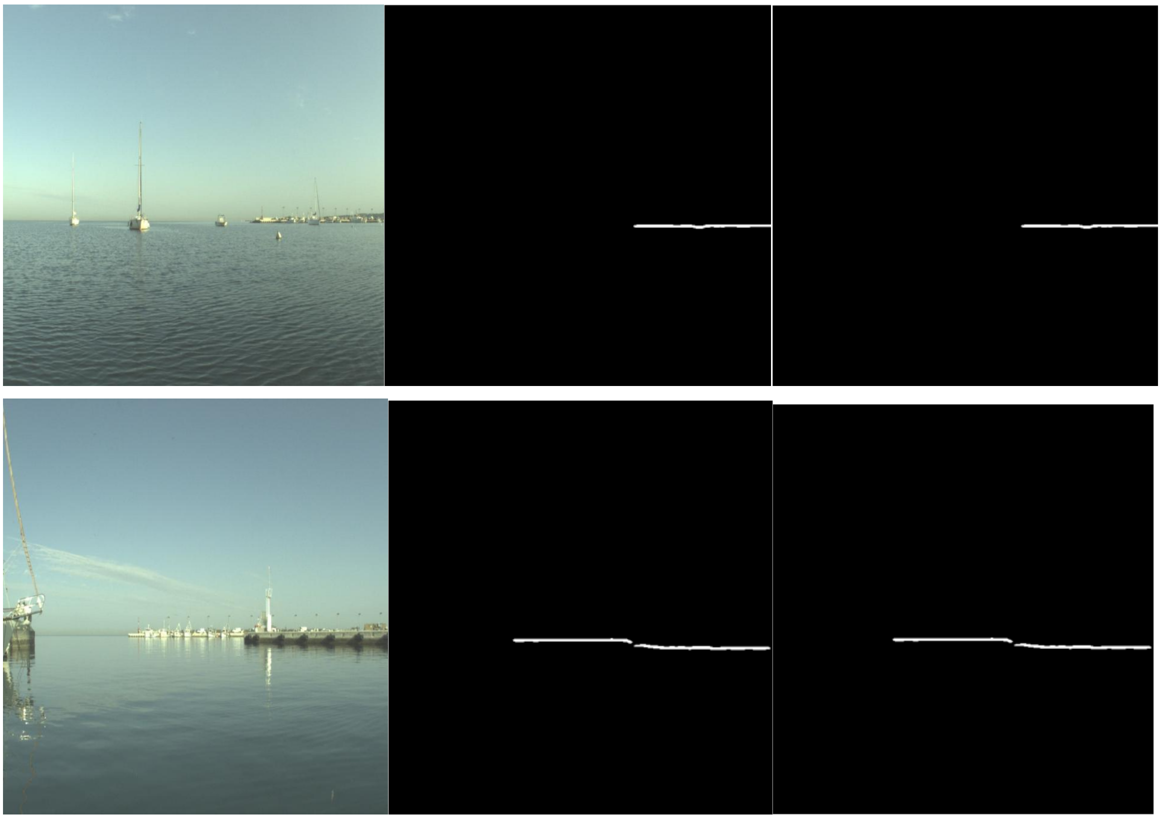

Fig. 5: Input image, ground truth mask and their predicted mask

4.2 Discussions

Our model achieved 0.68 as the dice score, which is a decent score, but it still needs improvement as a dice score greater than 0.8 is considered a successful segmentation. Accuracy is a misleading metric for this task, as the true values in the binary mask are significantly less than the false values. Therefore, it is important to consider only the dice score as a metric. The scores can be further improved by pre-training on a large dataset.

CHAPTER 5: CONCLUSION AND FUTURE PROSPECTS

The segmentation performed using U-Net was successful, and segmentation models can be used for water-edge detection. Further experimentation on U-Net can be done by pre-training on large segmentation datasets like ImageNet [37]. Performance can be improved by using sensor data from IMU. Modern segmentation models like SegFormer[38] can be used to detect the shoreline. We can use sophisticated loss functions, as mentioned in [20] and experiment with different models to get real-time inference speed. Segmentation features learned can be used to detect obstacles by creating a detection head and training the model for detection tasks. A completely separate model can also be prepared for object detection. A new dataset for Indian scenarios can be created to cater to Indian coastal requirements.

REFERENCES

1. Heidarsson, Hordur & Sukhatme, Gaurav. (2011). Obstacle detection from overhead imagery using self-supervised learning for Autonomous Surface Vehicles. IEEE International Conference on Intelligent Robots and Systems. 3160-3165. 10.1109/ IROS.2011.6048233.

2. J. Woo and N. Kim, "Vision based obstacle detection and collision risk estimation of an unmanned surface vehicle," 13th Int. Conf. on Ubiquitous Robots and Ambient Intelligence, 2016.

3. D. K. Prasad, D. Rajan, L. Rachmawati, E. Rajabaly, and C. Quek, "Video processing from electro-optical sensors for object detection and tracking in maritime environment: A survey," IEEE Trans. Intell. Transp. Syst., 2017.

4. https://youtu.be/QsFtE8aKcto?si=9qKblsL4e8v9712j

5. Krotosky, Stephen & Trivedi, Mohan. (2008). On Color-, Infrared-, and Multimodal-Stereo Approaches to Pedestrian Detection. Intelligent Transportation Systems, IEEE Transactions on. 8. 619 - 629. 10.1109/TITS.2007.908722.

6. Zhang, Z., Xu, H., Chao, Z., Li, X., & Wang, C. (2015). A Novel Vehicle Reversing Speed Control Based on Obstacle Detection and Sparse Representation. IEEE Transactions on Intelligent Transportation Systems, 16(3), 1321-1334.

7. Cao, Teng & Xiang, Zhiyu & Liu, Ji-Lin. (2015). Perception in Disparity: An Efficient Navigation Framework for Autonomous Vehicles With Stereo Cameras. IEEE Transactions on Intelligent Transportation Systems. 16. 1-14. 10.1109 TITS.2015.2430896.

8. Y. Lu, C. Rasmussen, "Simplified markov random fields for efficient semantic labeling of 3D point clouds," in 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 2690-2697.

9. Shim, I., et al. "An Autonomous Driving System for Unknown Environments Using a Unified Map," in IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 4, pp. 1999-2013, 2015.

10. Almeida, C. & Franco, Tiago & Ferreira, Hugo & Martins, Alfredo & Santos, Ricardo & Almeida, J.M. & Carvalho, João & Silva, E.. (2009). Radar based collision detection developments on USV ROAZ II. 1-6.10.1109/OCEANSE.2009.5278238.

11. Larson, Jacoby & Bruch, Michael & Halterman, Ryan & Rogers, John & Webster, Robert. (2007). Advances in Autonomous Obstacle Avoidance for Unmanned Surface Vehicles.

12. Gal, O. (2011). Automatic Obstacle Detection for USV's Navigation Using Vision Sensors. In A. Schlaefer & O. Blaurock (Eds.), Robotic Sailing (pp. 127–140). Springer Berlin Heidelberg.

13. Wang, H., Wei, Z., Wang, S., Ow, C. S., Ho, K. T., Feng, B., & Lubing, Z. (2011). Real-time obstacle detection for Unmanned Surface Vehicle. 2011 Defense Science Research Conference and Expo (DSR), 1–4. https://doi.org/10.1109/DSR.2011.6026880.

14. Wang, Han & Wei, Zhuo & Ow, Chek & Ho, Kah & Feng, Benjamin & Huang, Junjie. (2012). Improvement in real-time obstacle detection system for USV. 1317-1322. 10.1109/ICARCV.2012.6485336.

15. Cane, T., & Ferryman, J. (2016). Saliency-based detection for maritime object tracking. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 1257–1264. https://centaur.reading.ac.uk/68994/

16. Andersson, M., Johansson, R., Stenborg, K.-G., Forsgren, R., Cane, T., Taberski, G., Patino, L., & Ferryman, J. (2016). The IPATCH system for maritime surveillance and piracy threat classification. 2016 European Intelligence and Security Informatics Conference (EISIC), 200–200. https://doi.org/10.1109/EISIC.2016.054

17. Kristan, M., Sulic, V., Kovacic, S., & Pers, J. (2015). Fast image-based obstacle detection from unmanned surface vehicles. https://arxiv.org/abs/1503.01918

18. Bovcon, B., Mandeljc, R., Perš, J., & Kristan, M. (2018). Stereo obstacle detection for unmanned surface vehicles by IMU-assisted semantic segmentation. Robotics and Autonomous Systems, 104, 1–13. https://doi.org/10.1016/j.robot.2018.02.017

19. Bovcon, B., Muhovič, J., Perš, J., & Kristan, M. (2019). The MaSTr1325 dataset for training deep USV obstacle detection models. 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 3431–3438. https://doi.org/10.1109/IROS40897.2019.8967909

20. Bovcon, B., & Kristan, M. (2020). A water-obstacle separation and refinement network for unmanned surface vehicles. https://arxiv.org/abs/2001.01921

21. Bovcon, B., Muhovic, J., Vranac, D., Mozetic, D., Pers, J., & Kristan, M. (2022). MODS—A USV-Oriented Object Detection and Obstacle Segmentation Benchmark. IEEE Transactions on Intelligent Transportation Systems, 23(8), 13403–13418. https://doi.org/10.1109/tits.2021.3124192

22. Ahmed, M., Bakht, A. B., Hassan, T., Akram, W., Humais, A., Seneviratne, L., He, S., Lin, D., & Hussain, I. (2023). Vision-Based Autonomous Navigation for Unmanned Surface Vessel in Extreme Marine Conditions. https://arxiv.org/abs/2308.04283

23. R. Girshick, “Fast r-cnn,” in 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440–1448.

24. S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137– 1149, 2017.

25. T.-Y.Lin, P.Dolla ́r, R.Girshick, K.He, B.Hariharan, and S.Belongie, “Feature pyramid networks for object detection,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 936–944.

26. Joseph Redmon, Santosh Divvala, Ross Girshick, & Ali Farhadi. (2016). You Only Look Once: Unified, Real-Time Object Detection.

27. Liu, Wei & Anguelov, Dragomir & Erhan, Dumitru & Szegedy, Christian & Reed, Scott & Fu, Cheng-Yang & Berg, Alexander. (2016). SSD: Single Shot MultiBox Detector. 9905. 21-37. 10.1007/978-3-319-46448-0_2.

28. Zhi Tian, undefined., et al, "FCOS: Fully Convolutional One-Stage Object Detection," 2019.

29. Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, & Jian Sun. (2021). YOLOX: Exceeding YOLO Series in 2021.

30. https://www.nvidia.com/en-in/autonomous-machines/embedded-systems/jetson-orin/

31. Olaf Ronneberger, Philipp Fischer, Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation." (2015).

32. Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. https://arxiv.org/abs/1502.03167

33. https://pytorch.org/docs/stable/generated/torch.nn.BCEWithLogitsLoss.html

34. Kingma, D. P., & Ba, J. (2017). Adam: A Method for Stochastic Optimization. https://arxiv.org/abs/1412.6980

35. https://pytorch.org/docs/stable/generated/torch.optim.Adam.html

36. https://github.com/albumentations-team/albumentations

37. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. https://arxiv.org/abs/1409.0575

38. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., & Luo, P. (2021). SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. https://arxiv.org/abs/2105.15203