FusedWSS: Water Surface Segmentation Fusing Machine Learning and Geometric Cues

Jon Torgeir Grini† Rudolf Mester‡ Trym Anthonsen Nygård† Nicholas Dalhaug†

Edmund Førland Brekke† Annette Stahl†

† Department of Engineering Cybernetics, NTNU Trondheim, Norway

‡ Department of Computer Science, NTNU Trondheim, Norway

***Abstract*—Navigating unmanned surface vehicles (USVs) in urban waterways presents unique challenges due to irregular waterlines, obstacles, and reflections in the water. Determining the collision-free navigable area is crucial to enable safe USV operation. This paper introduces Fused Water Surface Segmentation (FusedWSS), a novel approach to water surface segmentation that aims to enhance navigation capabilities for USVs in complex harbor environments using a stereo camera. The method locates the water plane by performing plane fitting with outlier rejection and plane validation on the reconstructed 3D point cloud. From the plane parameters, the virtual horizon line is inferred and used for point cloud and image cropping. The water surface mask and virtual horizon line are fused with a deep learning-based semantic segmentation method to produce accurate and reliable water masks for each image frame. Additional refinement of the water mask is performed using detected obstacle masks. Validation was carried out using data from the MilliAmpere 2 autonomous ferry prototype in Trondheim, Norway, and a publicly available maritime dataset, demonstrating the efficacy of the methods.**

***Index Terms*—Situational awareness, stereo camera, maritime autonomy, segmentation, detection, fusion**

I. Introduction

In recent years, significant advancements have been made in the field of situational awareness for USVs. These vessels are well-suited for operating in narrow waters such as canals, harbors, and coastal waters. To enable USVs to operate safely in these environments, a complete and accurate understanding of their surroundings is crucial. Water surface segmentation enables a rich scene understanding by determining the collision-free navigable area for surface vessels, ensuring situational awareness and safe decision-making. Different sensors are available for this task, but cameras and stereo cameras have proved to be effective, affordable, and potent alternatives for water segmentation [1], [2], [3].

Several techniques and elements should be considered building blocks for a successful water surface segmentation algorithm. One such technique is estimating the water surface as a plane in 3D. The plane can be estimated using distance points given by stereo vision (x, y, d), from reconstructed 3D points (x, y, z) [4], [5] or directly from disparity values. Given plane parameters, a reclassification of dense pixels into the classes water or non-water can be performed from a dense disparity or dense distance map [1], [6]. In order to provide the most accurate segmentation mask, the points used for 3D plane estimation should pass through a filter, reducing the amount of outliers. Filtering may include limiting the region of interest in an image by, for example, a horizon estimate, estimates of the water surface mask from previous frames, or appearance-based segmentation methods [5]. As the water surface appears as a horizontal surface, non-horizontal surfaces can also be excluded [7]. The estimated plane parameters for the water surface can provide us with the attitude information of the vessel [8]. By estimating the rotation and translation between the camera coordinate frame and the plane coordinate frame, the pitch, roll, and height of the camera above the water surface can be obtained. Yaw cannot be determined without further external reference, such as an inertial measurement unit (IMU), which alternatively can be used to obtain the attitude information of the vessel [9]. The waterline can be extracted using the pixel change in the water surface segmentation mask [10]. For providing a consistent and reliable water surface mask over time, temporal filtering is a technique that uses information from the last N time steps to provide the best possible water surface segmentation mask [11]. Recent advancements in water surface segmentation are

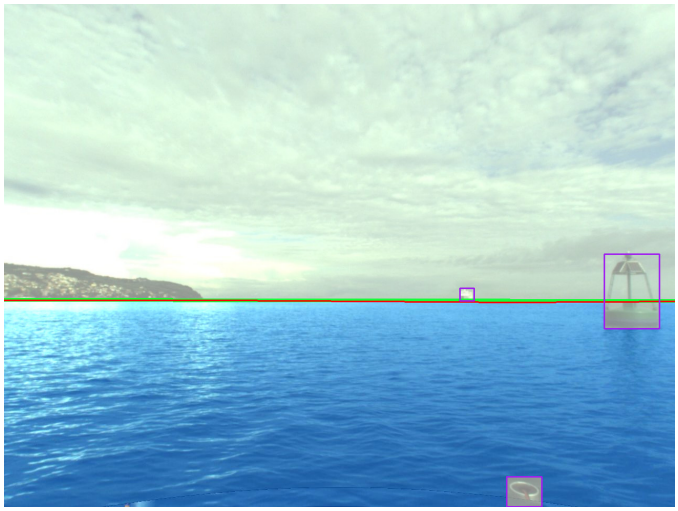


Fig. 1. Water segmentation in blue, and horizon line estimation in green with red ground truth annotation for the waterline. Bounding boxes for obstacles are in purple. Water segmentation is performed using a fusion of geometry cues obtained from a stereo camera and a learning-based approach.

This work was supported by The Research Council of Norway (project number 333917).

mostly focused on appearance-based approaches. Deep Neural Networks (DNNs), especially Convolutional Neural Networks (CNNs), are widely used for segmentation purposes with promising results [1], [3], [6], [10].

The most closely related study to ours is [5]. However, our approach extends beyond this by incorporating plane fitting with the ability to infer the horizon line from visual data alone. Additionally, we enhance water segmentation by integrating segmentation masks obtained from geometric analysis with those derived from appearance-based methods in the late stages. The preceding water mask and horizon line are used for filtering the point cloud, as well as limiting the image region processed by an appearance-based segmentation method. To the authors' knowledge, no existing method combines geometric and appearance-based segmentation with filtering using the horizon line and the previous water surface masks for water segmentation purposes.

Consequently, we present the following contributions: (i) A method for water surface segmentation based on a visual fusion of classical geometric analysis and learning-based approaches. (ii) Demonstrate the use of this water surface segmentation method in a harbor environment. (iii) Evaluation of the segmentation results and the horizon line estimation using the annotated MODD2 dataset [12].

II. Related Work

Wang et al. [4], Muhovič et al. [5] and Griesser et al. [8] uses RANdom SAmple Consensus (RANSAC)-based approaches [13] for plane estimation on 3D data from a stereo camera. [4] uses subsampling to reduce computational cost related to the plane fitting, while [5] and [8] uses semantic segmentation networks for pre-classification of the pixels into water and non-water before reconstruction, achieving more robust plane estimation. [5] addresses the problem that stereo vision fails in calm water due to homogeneous surfaces by utilizing an external IMU to constrain the plane orientation, while [8] proposes a method for estimating pitch, roll, and camera height from the plane parameters.

The paper by Kristan et al. [11] presents a supervised segmentation algorithm designed for monocular video streams. The algorithm uses a mixture model of three Gaussian components to represent the three central regions of an image: water, land, and sky, implying weak structural constraints on the segmentation process. Bovcon et al. [9] builds on this work by incorporating the pitch and roll from an onboard IMU to estimate the horizon line in the image, which is used to automatically adjust the priors and hyper-priors in the extended graphical model. In [14], Bovcon and Kristan propose a joint image segmentation approach based on [9] where they enhance the single-view model by adding a constraint that favors consistent class labeling to pixels in the left and right images that correspond to the same 3D regions. The method fits the graphical model to both images jointly, resulting in a more accurate class-label posterior map, which is then used to identify water edges and obstacles.

Zhan et al. [1] propose an adaptive segmentation method for classifying each pixel into the same three categories as [11]: land, water, and sky. The segmentation network is based on an encoder-decoder U-net architecture [15]. The classified regions are refined using a conditional random field, followed by analyzing the superpixel map. The network trains itself on the refined pseudo labels and a weight map generated by the prediction confidence making it adaptable to new environments

Steccanella et al. [10] presents a waterline and obstacle detection method. The image is segmented into water and non-water regions using a CNN inspired by U-Net [15], similar to [1]. The position of the waterline is retrieved using edge detection on the binary mask from the CNN.

Zhou et al. [2] presents a waterway segmentation network where accuracy is improved by learning the context dependency of features through a modified context prior, thereby reducing effects such as reflections and ripples which pose a challenge in waterline extraction methods such as [10]. Refinement of the waterline is achieved using an auto-encoder.

Both Erfani et al. [16] and Bovcon et al. [3] presents a deep neural networks for waterbody semantic segmentation using the encoder of ResNet-101 network as their backbone [17]. [16] then takes advantage of two different paths to segment aquatic bodies and non-aquatic bodies regions in parallel. Probability maps from each path are created before cross-path modulation blocks are applied to merge the probability maps, followed by concatenation and upsampling to produce segmentation masks. [3] gradually fuse inertial information from an IMU with visual information from the ResNet-encoder using a novel loss function designed to increase separation between water and obstacles. Building on this, [18] presents WaSR-T, which extracts temporal context from a sequence of recent frames to reduce ambiguity. It comes at a high computational cost, but according to the marine obstacle benchmark MODS [19], WaSR-T is the current state-of-the-art.

The paper by Bovcon et al. [19] addresses the issue that current maritime datasets do not adequately capture the complexity of real-world maritime scenes, which complicates cross-paper comparison. To overcome this issue, the paper proposes the Marine Obstacle Detection Benchmark (MODS), an obstacle detection benchmark addressing two essential maritime perception tasks: obstacle detection and obstacle segmentation. The paper presents a diverse maritime dataset containing about 81k stereo images synchronized with an onboard IMU and about 60k objects annotated.

III. The Water Segmentation Pipeline

The visual fusion method presented in this paper, Fused-WSS, combines geometric cues from stereo and appearance-based segmentation. The pipeline is visualized in figure 2. The method employs RANSAC-based Water Plane Segmentation (RWPS) and combines this with the convolutional neural network FastSAM in the segmentation step [20]. A refinement module ensures accurate segmentation.

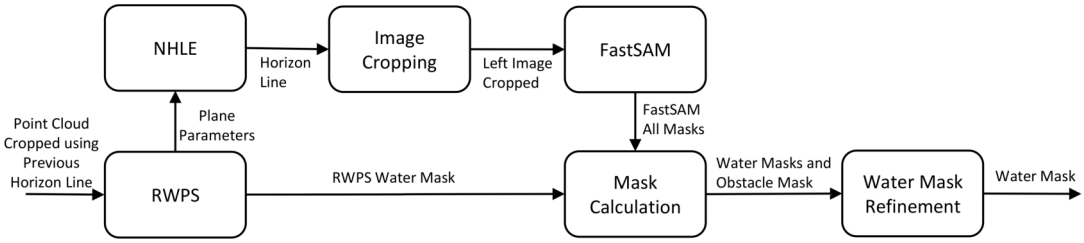


Fig. 2. The pipeline of FusedWSS. FusedWSS merges the methods of RWPS and FastSAM, resulting in an accurate and reliable water mask.

A. RANSAC-based Water Plane Segmentation (RWPS)

RWPS uses the point cloud, the set of 3D points, of the reconstructed scene from a stereo camera to find the most prominent plane in a pre-selected part of the image. RANSAC is applied in the plane fitting stage to reduce the impact of outliers.

Before fitting a plane to the point cloud, a point selection and cropping is performed by considering temporal clues. We use the pixels detected as part of the water surface by RWPS for the previous frame to guide the search for the water plane in the current frame. The previous water mask serves as a good predictor of the new mask. In addition to this point selection, the point cloud is cropped. Using the estimate of the horizon line (section III-D2), the points above the horizon line are discarded. These two steps accelerate the computation and serve as a good prediction for the water surface.

To retrieve the water segmentation mask from RWPS, the complete point cloud of 3D points is used along with the estimated plane parameters. The distance $d_\perp$ from each point (x_i, y_i, z_i) to the plane is calculated using

$$d_{\perp,i} = \frac{\mid ax_i + by_i + cz_i + d \mid}{\sqrt{a^2 + b^2 + c^2}}. \tag{1}$$

The points within a predefined distance threshold to the plane are backprojected into the image frame and included in the water segmentation mask RWPS outputs.

As the water surface may not always be the most prominent plane in the image, we use a validity check of the detected plane. The validity check ensures that the deviation in distance and rotation between the current and prior plane does not exceed some predefined threshold. The current plane is compared against the preceding and the nominal plane parameters. If the plane complies with either, it is considered valid. If the plane does not comply with either, the nominal plane parameters replace the current parameters to create the mask of the water surface. Therefore, correct initialization is essential.

B. Appearance-based Segmentation

The segmentation mask of the water surface obtained from RWPS completely depends on the quality of 3D information, which is not always available. 3D information from stereo cameras suffers in the presence of textureless surfaces and reflections that often appear in the maritime domain. To overcome the issues related to occasionally poor 3D information, we fused the water mask obtained from the geometric cues with an appearance-based segmentation mask. A suitable state-of-the-art method is FastSAM [20], which uses monocular images and is inspired by the well-proven segmentation method SAM [21]. SAM is built upon a Vision Transformer (ViT) architecture, trained on an extensive dataset, and generalizes very well into different environments. Although powerful, ViTs are computationally costly and not suited for real-time applications. FastSAM uses a CNN rather than ViT and organizes the segmentation process into two main stages. The first stage incorporates a CNN-based detector, based explicitly on YOLOv8-seg [22]. YOLOv8-seg uses the YOLACT real-time instance segmentation method [23]. After segmenting objects in the entire image, FastSAM uses prompts to provide specific masks. FastSAM supports point- and box-prompting. In the present paper, we use the option to provide segmentation masks for all objects in the scene.

In segmentation accuracy, FastSAM demonstrates a performance comparable to SAM, although it is trained on only 2% of the SA-1B dataset [21]. It also operates at speeds about 50 times faster than SAM [20].

C. Mask Selection and Refinement

The detection result of RWPS and the segmentation by FastSAM are two important parts of the water segmentation method FusedWSS presented in this paper. As seen in figure 2, a mask calculation between all the masks from FastSAM and the water mask from RWPS is performed before a refinement step for the resulting masks.

In the mask selection stage, an Intersection-over-Union (IoU) calculation is performed with the RWPS water mask for every FastSAM mask. Intersection-over-Union is calculated as

$$\text{IoU} = \frac{|A \cap B|}{|A| \cup |B|}, \tag{2}$$

where A and B denote the set of pixels corresponding to each mask. Masks with an IoU score above some threshold value are classified as water masks, while the masks below the threshold value are classified as obstacle masks. We merge the masks classified as water from FastSAM and RWPS into one using logical OR and do the same with the obstacle masks. Water mask refinement is performed by setting all pixels classified as both water and obstacle in the respecting masks to obstacle.

D. Normal-based Horizon Line Estimator

The Normal-based Horizon Line Estimator (NHLE) is a visual-based method for estimating the horizon line. In this paper, the horizon line is where the sky separates from water, even if it is hidden by land. A good estimate of the horizon line is especially useful for cropping of the point clouds and image. By discarding the information above the horizon line, we reduce computational cost and remove noise from, for example, depth estimates of the sky.

1) Estimating Roll and Pitch using Plane Parameters: We estimate the camera's roll and pitch using the fitted plane. The camera coordinate frame is defined with the x-axis pointing to the right, the y-axis pointing downwards, and the z-axis pointing forward. The roll is the rotation about the z-axis, and the pitch is the rotation about the x-axis.

Roll and pitch angles are extracted using the normal vector. The roll angle θ and pitch angle ϕ are given by

$$\theta = \arctan \frac{n_x}{n_y} \tag{3}$$

$$\phi = \arctan \frac{n_z}{n_y} \tag{4}$$

2) Estimating the Horizon Line using Plane Parameters: The horizon line can be directly computed from the plane equation together with camera parameters estimated during calibration. Any plane that is parallel to the estimated water plane can be written on the form

$$ax + by + cz + e = 0 \tag{5}$$

where e is an arbitrary number. The value of e that lets the plane pass through the perspective center is $e = 0$. The plane that passes through the perspective center intersects the image plane when $z = f$, where f is the focal length. Consequently, we have the horizon line in pixel space with the origin at the principal point given as

$$ax_p + by_p + cf = 0 \tag{6}$$

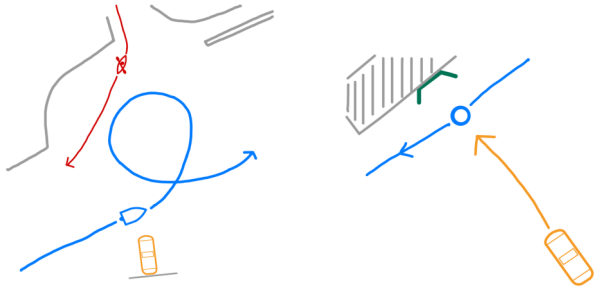
where (x_p, y_p) are coordinates on the horizon line in pixel units. We can transform the pixel coordinates into image coordinates (u, v) using

$$\begin{aligned} u &= x_p + c_x \\ v &= y_p + c_y \end{aligned} \tag{7}$$

where (c_x, c_y) denotes the principal point. Inserting (7) into (6) yields

$$\begin{aligned} au + bv + k &= 0 \\ k &= cf - ac_x - bc_y \end{aligned} \tag{8}$$

which is the equation for the horizon line in the image coordinate system.



(a) (b)

Fig. 3. Hand-drawn visualization of episodes. (a) Episode 1: The ferry (in orange) is at rest while observing a boat (in blue) and a kayak (in red) maneuvering in front of it. (b) Episode 2: An inflatable ring (in blue) crosses the scene from right to left while the ferry (in orange) moves towards the dock.

IV. Experimental Platform and Data

A. The MilliAmpere 2 Dataset

The MilliAmpere 2 dataset, recorded at Ravnkloa, Trondheim, Norway, using the MilliAmpere 2 autonomous ferry prototype [24], comprises 13 episodes depicting varied, realistic harbor navigation scenarios. It features onboard video recordings from two stereo camera setups alongside LIDAR, IMU, GNSS, and INS data.

For our experiments, we utilized the camera information from the ZED 2 camera placed on the aft port of the ferry. The stereo camera has a baseline of 120 mm and is equipped with an IMU, a barometer, and a magnetometer. The coordinate frame is oriented such that the x-axis points right, y-axis points downwards, and z-axis points forward [25]. The video data was captured at a resolution of 1920×1080 pixels and 15 fps. figure 3 visualizes the two test episodes we use from the MilliAmpere 2 dataset named episode 1 and episode 2.

B. Multi-modal Marine Obstacle Detection Dataset 2

The Multi-modal Marine Obstacle Detection Dataset 2 (MODD2) dataset consists of 28 video sequences, yielding a total of 11675 stereo frames, each with a resolution of 1278x958 pixels. It is a part of the MODS benchmark [19] and presents a range of challenging conditions, including diverse weather scenarios, sudden motion changes, and issues with sun glitter and reflections. It contains time-synchronized data from onboard sensors: GNSS, IMU, and compass [12]. Annotations in the dataset delineate the waterline with polygons, while obstacles, whether floating or at the sea's edge, are marked with bounding boxes. Evaluation of the dataset of MODD2 gives us a quantitative analysis of a publicly available dataset. Episode MODD2_1 is well-suited for evaluating the horizon line estimator, as it shows the open sea. The annotated water line will answer to the horizon line in the open sea. As an evaluation metric for the estimated horizon line, Root Mean Squared Deviation (RMSD) will be used for calculating the vertical deviation of the estimated horizon line and the annotated water line. That is the vertical distance for a point (x_i, y_i)

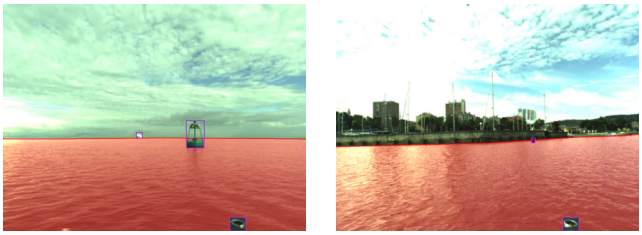


Fig. 4. Examples of ground truth water surface mask and annotated water line in red for the MODD2 dataset [12]. Bounding boxes for obstacles and parts of the USV in purple are excluded from the water surface mask.

on the polygon-annotated water line to the corresponding point $(x_i, \hat{y}_i)$ on the horizon line estimate

$$RMSD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{9}$$

where n is the number of points on the annotated water line.

To obtain a quantitative analysis of the water surface segmentation mask, an annotated mask is created using everything below the annotated waterline, with bounding boxes for obstacles and parts of the USV itself removed from the mask. The IoU-score between the annotated water surface mask and the estimated water surface mask is then calculated for each frame in an episode, providing the mean IoU-score (mIoU) for a whole episode. Two examples of the ground truth water surface mask and ground truth waterline are depicted in figure 4

V. Analyzing RWPS and FastSAM for Detection and Segmentation

From figure 6, we see instances of water surface segmentation masks produced by RWPS for the maneuver and docking episode. The method of RWPS fits a flat plane to the water surface. Still, we can see from figure 6(a)(b) that wakes and ripples produced by the boat are included as part of the segmentation mask for the water surface. The outlier rejection threshold specified for the robust estimation in RWPS is set to $0.1\,\mathrm{m}$ such that smaller wakes and ripples are included in the water mask. The outlier rejection threshold also introduces Non-Water-to-Water (NW2W) labeling errors. This is part of the reason why the bottom parts of the boat in figure 6(a) are included in the water mask. The balance in choosing an outlier rejection threshold that includes wakes and ripples in the water surface mask and excludes all parts of surface objects and debris is a challenge with RWPS.

Figure 6(d) displays the scenario where the ferry moves close to the dock, and other planar surfaces become more significant than the water surface in the image. We see that RWPS is still able to make the correct detection of the water plane due to the validity check and use of prior information of the water plane as explained in section III-A.

As seen in figure 6, the segmentation masks contain both alternating and stationary gaps, which introduce Water-to-Non-Water labeling errors. RWPS depends upon reliable 3D information, which is not always available due to reflections or homogeneous regions in the water surface. Alternating gaps may only be present for a frame or two, and the effect of these could be reduced by looking for time consistency in the segmentation mask. Other gaps are stationary due to homogeneous areas on the water surface where stereo-matching is difficult and is not as easily solvable using the time aspect.

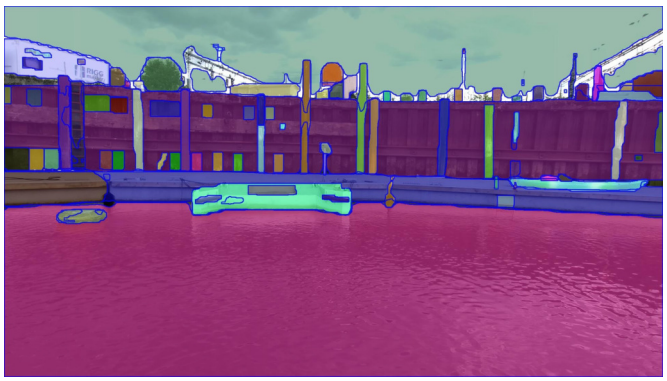


Fig. 5. All segmentation masks produced by FastSAM for one frame in the docking episode.

The accuracy of depth estimates from stereo cameras decreases with longer distances, resulting in more unreliable 3D data for greater distances. The data points also become more sparse the further away they are, which further degrades the quality of the 3D point cloud in the distance. Thus, with increasing distance, more outliers are rejected by RANSAC, increasing the number of gaps and W2NW labeling errors. Looking at the Birds Eye View (BEV) in figure 11(a), we observe gaps and reduced data quality with increasing distance, as well as sparser data.

Depth information in a stereo camera setup can only be retrieved in parts of the scene that both cameras can see. As the cameras do not have the same view of the scene, there will be regions in the image where no depth information is possible to gather. These regions occur to the left in the left camera image and to the right in the right camera image. We

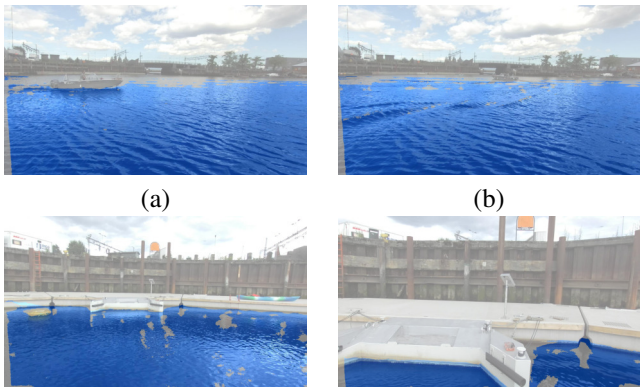
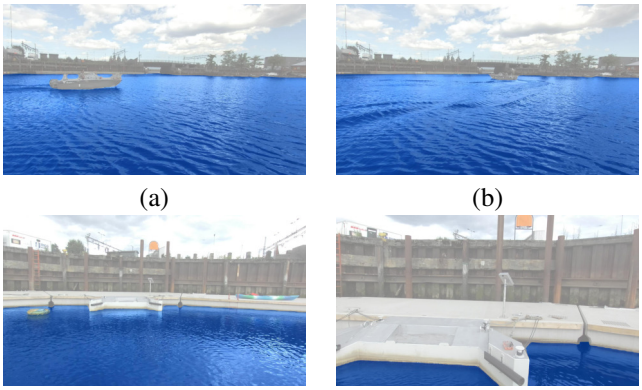


Fig. 6. RWPS results in the (a-b) maneuver episode (c-d) docking episode.



(a) (b) (c) (d)

Fig. 7. FastSAM results in the (a-b) maneuver episode (c-d) docking episode.

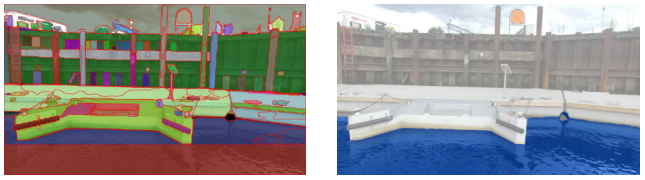
can see the region for the left camera in the images of figure 6. The non-overlapping regions in the images introduce W2NW labeling errors and pose a weakness of methods that rely on depth from a stereo camera.

Figure 7 displays the results of the segmentation of the water surface using FastSAM. From the figures, we can see that FastSAM creates precise segmentation masks both in the maneuver and docking episode. As FastSAM does not rely on the 3D information in the same way as RWPS, the gaps caused by poor 3D information in figure 6 are eradicated from the water mask in figure 7. As seen in figure 7, the left part of the images does not suffer from W2NW labeling errors due to lack of depth information as in figure 6. This enables FastSAM to obtain the complete water mask from the image.

Looking at figure 5, we can analyze the masks for the objects detected by FastSAM for a single image. For the most part, FastSAM creates precise segmentation masks for all objects in the image. However, it detects multiple masks for the inflatable ring, some of which the ring appears as a separate object and some of which it is detected as part of the water body. This motivates the use of obstacle masks to refine the water surface mask. The segmentation result from FastSAM sometimes includes different objects in several of the output masks, which means that obstacles in the water might be included as part of the water surface. Therefore, being able to analyze obstacle masks to identify this misdetection is an important step for accurate water surface segmentation results. FastSAM produces segmentation masks for all objects it finds in the images, and is not able to detect which mask belongs to the water surface. This means that FastSAM alone is not sufficient for water surface detection and segmentation. To select the water masks from FastSAM in figure 7, point prompting is applied using the bottom-centered pixel.

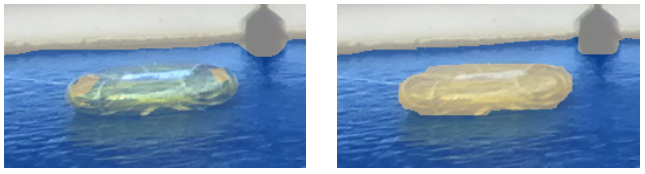
VI. Results

Figure 12 presents the results by applying FusedWSS to the maneuver and docking episode of the MilliAmpere 2 dataset. The method has comparable performance with FastSAM for these frames. We see the same tight mask around the boat, as well as the exclusion of the kayaker in the water surface



(a) (b)

Fig. 8. Example where FastSAM has divided the water surface into different masks, and FusedWSS is able to recover the complete water mask through the mask comparison. (a) All segmentation masks produced by FastSAM (b) Segmentation result using FusedWSS

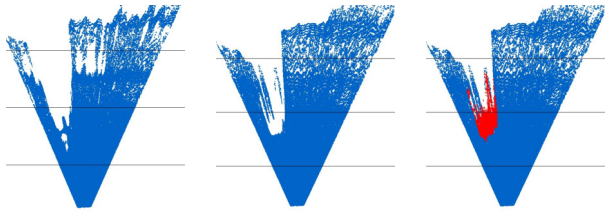


(a) (b)

Fig. 9. Segmentation comparison of the inflatable ring for a single frame present in the docking episode of the MilliAmpere 2 dataset. Segmentation using (a) FastSAM (b) FusedWSS

(a) (b)

Fig. 10. Segmentation comparison of failed segmentation using FastSAM for a single frame present in the docking episode of the MilliAmpere 2 dataset. Segmentation using (a) FastSAM (b) FusedWSS



(a) (b) (c)

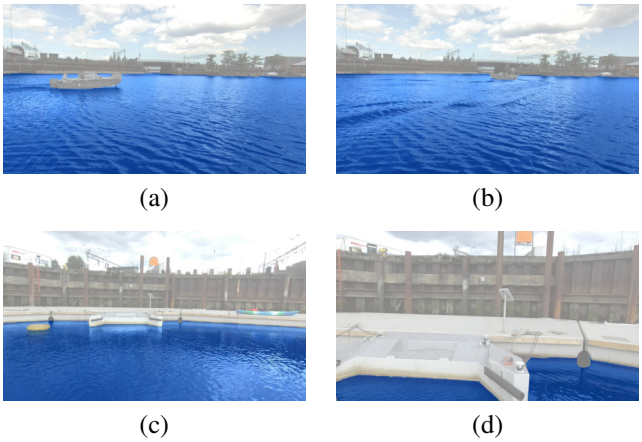
Fig. 11. The figure shows BEVs in the maneuver episode. The blue points depict water, and the red points in (c) depict obstacles. (a) BEV using RWPS (b) BEV using FastSAM (c) BEV using FusedWSS.

TABLE I
RMSD for NHLE in episode MODD2_1. "wo init" means without initialization and "wo fd" means without failed detections.

Episode	Avg. RMSD	Avg. RMSD wo init	Avg. RMSD wo init&fd.
MODD2_1	3.77	3.76	**3.65**

TABLE II
MIOU IN EPISODES FROM THE MODD2 DATASET. "WO INIT" MEANS WITHOUT INITIALIZATION AND "WO FS" MEANS WITHOUT FAILED SEGMENTATIONS.

Episode	FusedWSS	FusedWSS wo init	FusedWSS wo init&fs.	mIoU RWPS
MODD2_1	0.977	0.976	**0.986**	0.634
MODD2_2	0.935	0.936	**0.992**	0.434
MODD2_3	0.974	0.980	**0.987**	0.528



(a) (b) (c) (d)

Fig. 12. FusedWSS results in the (a-b) maneuver episode (c-d) docking episode.

mask. We can analyze the BEVs in figure 11(b)(c) to separate the mechanisms of FastSAM and FusedWSS. FusedWSS uses the mask from RWPS to classify the segmentation masks from FastSAM into water and obstacles (explained in section III-C), while the purely FastSAM-based approach relies on simple point-prompting. This enables FusedWSS to detect masks for obstacles visualized in red in figure 11(c), which is used to refine the water mask. An example where the refinement using obstacles improves segmentation is shown in figure 9. Figure 9 shows a zoomed-in view of the inflatable ring present in the docking episode. Using the detected obstacle masks, FusedWSS is able to exclude the ring from the water mask, whereas FastSAM alone includes the inflatable ring as water.

Figure 8(a) displays the instance where the segmentation result of FastSAM divides the water surface into several different masks. Choosing one mask from this selection would not be sufficient to represent the complete water surface. FusedWSS combines the different masks produced by FastSAM for the water surface into one complete mask for the water surface. FusedWSS is able to do so due to the fusion with RWPS. The fusion with RWPS is also convenient when occasionally FastSAM fails to find parts of, or the entire mask for the water surface as in figure 10. FusedWSS is then still able to produce a segmentation result using the RWPS water mask.

By applying FusedWSS to the MODD2 dataset, we are able to obtain a quantitative analysis of the method. To evaluate the horizon line estimator, we use episode MODD2_1, where the scene mostly consists of the open sea. In this way, the annotated water line approximately equals the actual horizon line. The USV experiences large variations in pitch and roll throughout the episode, as visualized in figure 13. In the two frames, the horizon line visualized in green closely follows the ground truth for the horizon line. The resulting average RMSD values are displayed in Table I. We see that the average RMSD value slightly improves as we remove the numbers from the initialization phase and misdetections. Until we have initialized the plane, the horizon line has the default value of the middle height in the image, causing a larger deviation. Misdetections can happen when stereo matching conditions are difficult, such as strong reflections and homogeneous areas. Usually, this sporadically happens for a very limited number of frames. In these cases, the deviations from the annotated water line can be large. However, an average RMSD of ≈ 3.8 pixels with initialization and misdetections proves an impressive horizon line estimator.

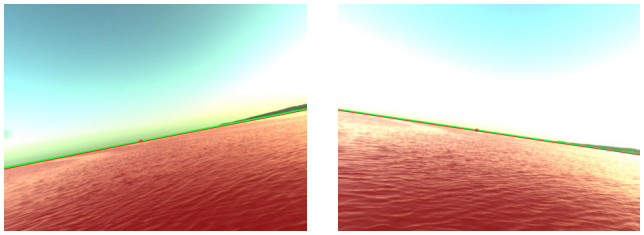


Fig. 13. Evaluation of horizon line estimator in open sea on the MODD2 dataset. The ground truth water surface mask and annotated water line are in red, and the estimated horizon line is shown in green. The USV experiences large variations and pitch and roll throughout the episode.

For the water segmentation task, we used three different episodes from the MODD2 dataset to obtain a quantitative analysis. The mIoU-scores from Table II show that FusedWSS achieves impressive accuracy for the water segmentation task. There is a small difference in the mIoU-score when including or excluding the results during the initialization. However, we see a significant increase in performance when neglecting the frames with failed segmentations. Failed segmentation happens either if RWPS is not able to find a valid plane, or FastSAM doesn't recognize the water surface and does not create a mask for it. In the case that FastSAM does not provide a segmentation mask for the water surface, the RWPS water mask serves alone as the segmentation mask for the water surface, as seen in Table II.

VII. CONCLUSION

In this paper, a water segmentation pipeline for unmanned surface vehicles was proposed. A stereo camera was used to reconstruct the 3D scene, and RANSAC-based plane fitting was applied to estimate the water surface plane, resulting in RWPS. NHLE inferred the horizon line from the plane parameters. RWPS and NHLE were fused together with an appearance-based segmentation method to provide an accurate water surface segmentation method, FusedWSS. RWPS and the appearance-based method were analyzed and compared using two different episodes from the MilliAmpere 2 dataset,

while FusedWSS was additionally evaluated on the MODD2 dataset. FusedWSS achieved up to 97.7% in mIoU-score on the MODD2 episodes, suggesting it can contribute to improved situational awareness and safe-decision making.

REFERENCES

[1] W. Zhan, C. Xiao, Y. Wen, C. Zhou, H. Yuan, S. Xiu, X. Zou, C. Xie, and Q. Li, "Adaptive semantic segmentation for unmanned surface vehicle navigation," *Electronics*, vol. 9, no. 2, p. 213, 2020.

[2] R. Zhou, Y. Gao, P. Wu, X. Zhao, W. Dou, C. Sun, Y. Zhong, and Y. Wang, "Collision-free waterway segmentation for inland unmanned surface vehicles," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–16, 2022.

[3] B. Bovcon and M. Kristan, "WaSR — A Water Segmentation and Refinement Maritime Obstacle Detection Network," *IEEE Transactions on Cybernetics*, vol. 52, no. 12, pp. 12661–12674, 2021.

[4] H. Wang and Z. Wei, "Stereovision based obstacle detection system for unmanned surface vehicle," in *2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 917–921, IEEE, 2013.

[5] J. Muhovič, B. Bovcon, M. Kristan, J. Perš, *et al.*, "Obstacle tracking for unmanned surface vessels using 3-D point cloud," *IEEE Journal of Oceanic Engineering*, vol. 45, no. 3, pp. 786–798, 2019.

[6] W. Zhan, C. Xiao, Y. Wen, C. Zhou, H. Yuan, S. Xiu, Y. Zhang, X. Zou, X. Liu, and Q. Li, "Autonomous visual perception for unmanned surface vehicle navigation in an unknown environment," *Sensors*, vol. 19, no. 10, p. 2216, 2019.

[7] H. Badino, U. Franke, and D. Pfeiffer, "The Stixel World - A Compact Medium Level Representation of the 3D-World," in *Pattern Recognition: 31st DAGM Symposium, Jena, Germany, September 9-11, 2009. Proceedings 31*, pp. 51–60, Springer, 2009.

[8] D. Griesser, G. Umlauf, and M. O. Franz, "Visual pitch and roll estimation for inland water vessels," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1961–1967, IEEE, 2023.

[9] B. Bovcon, J. Perš, M. Kristan, *et al.*, "Improving vision-based obstacle detection on USV using inertial sensor," in *Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis*, pp. 1–6, IEEE, 2017.

[10] L. Steccanella, D. D. Bloisi, A. Castellini, and A. Farinelli, "Waterline and obstacle detection in images from low-cost autonomous boats for environmental monitoring," *Robotics and Autonomous Systems*, vol. 124, p. 103346, 2020.

[11] M. Kristan, V. S. Kenk, S. Kovačič, and J. Perš, "Fast image-based obstacle detection from unmanned surface vehicles," *IEEE transactions on cybernetics*, vol. 46, no. 3, pp. 641–654, 2015.

[12] B. Bovcon, J. Perš, M. Kristan, *et al.*, "Stereo obstacle detection for unmanned surface vehicles by imu-assisted semantic segmentation," *Robotics and Autonomous Systems*, vol. 104, pp. 1–13, 2018.

[13] K. G. Derpanis, "Overview of the RANSAC algorithm," *Image Rochester NY*, vol. 4, no. 1, pp. 2–3, 2010.

[14] B. Bovcon and M. Kristan, "Obstacle detection for USVs by joint stereo-view semantic segmentation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5807–5812, IEEE, 2018.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234–241, Springer, 2015.

[16] S. M. H. Erfani, Z. Wu, X. Wu, S. Wang, and E. Goharian, "ATLANTIS: A Benchmark for Semantic Segmentation of Waterbody Images," *Environmental Modelling & Software*, vol. 149, p. 105333, 2022.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[18] L. Žust and M. Kristan, "Temporal context for robust maritime obstacle detection," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6340–6346, IEEE, 2022.

[19] B. Bovcon, J. Muhovič, D. Vranac, D. Mozetič, J. Perš, and M. Kristan, "MODS—A USV-Oriented Object Detection and Obstacle Segmentation Benchmark," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 13403–13418, 2021.

[20] X. Zhao, W. Ding, Y. An, Y. Du, T. Yu, M. Li, M. Tang, and J. Wang, "Fast segment anything," 2023.

[21] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," *arXiv:2304.02643*, 2023.

[22] G. Jocher, A. Chaurasia, and J. Qiu, "YOLO by Ultralytics," Jan. 2023.

[23] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "Yolact: Real-time instance segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9157–9166, 2019.

[24] E. F. Brekke, E. Eide, B.-O. H. Eriksen, E. F. Wilthil, M. Breivik, E. Skjellaug, Ø. K. Helgesen, A. M. Lekkas, A. B. Martinsen, E. H. Thyri, *et al.*, "milliAmpere: An autonomous ferry prototype," in *Journal of Physics: Conference Series*, vol. 2311, p. 012029, IOP Publishing, 2022.

[25] Stereolabs, "Coordinate frames - stereolabs," 2023.