Obstacle Detection for USVs by Joint Stereo-View Semantic Segmentation*

Borja Bovcon[1] and Matej Kristan[1]

***Abstract*— We propose a stereo-based obstacle detection approach for unmanned surface vehicles. Obstacle detection is cast as a scene semantic segmentation problem in which pixels are assigned a probability of belonging to water or non-water regions. We extend a single-view model to a stereo system by adding a constraint which prefers consistent class labels assignment to pixels in the left and right camera images corresponding to the same parts of a 3D scene. Our approach jointly fits a semantic model to both images, leading to an improved class-label posterior map from which obstacles and water edge are extracted. In overall F-measure, our approach outperforms the current state-of-the-art monocular approach by 0.495, a monocular CNN by 0.798 and their stereo extensions by 0.059 and 0.515, respectively on the task of obstacle detection while running real-time on a single CPU.**

I. Introduction

Small-sized unmanned surface vehicles (USVs) are portable, affordable and capable of navigating in shallow waters and narrow marinas. This makes them particularly attractive for coastal environmental patrol and remote inspection of difficult-to-reach man-made structures such as dams of power plants and water reservoirs. These tasks require a high level of autonomy which primarily depends on timely detection and avoidance of nearby obstacles and floating debris.

Standard sensors for obstacle detection like radar [1], lidar [16] and sonar [8] are usually expensive and inappropriate due to limited payload capacity and a very restricted power supply of the small-sized USVs. This is why lightweight and information-rich sensors, such as cameras, are gaining prominence. Nevertheless, aquatic environments pose significant challenges for computer-vision-based obstacle detection. Standard object detectors [14] cannot be used due to huge variety of potential obstacle appearance. Pretrained texture models for water segmentation [6] cannot be used to detect water due to significant variations depending on weather conditions and camera gains. Standard stereo vision-based methods cannot be readily applied due to rapidly changing water surface, reflections, and absence of texture on calm water. In addition, obstacles that do not sufficiently protrude through the water surface cannot be reliably detected by 3D range sensor systems.

Recently, Kristan *et al.* [12] proposed a semantic model, which is fitted to each image and simultaneously classifies pixels into water and non-water. Edge of the water specifies the range of water, while the regions of non-water pixels within the water region are considered potential obstacles. Despite overall good segmentation results, the approach fails in presence of visual ambiguities (e.g., haze on the horizon). Bovcon *et al.* [4], [3] extended the approach [12] by including inertial measurement unit (IMU) measurements to calculate the location of the horizon in the image to constrain the segmentation. To address a large number of false detections caused by sun glitter and sea foam, they apply their segmentation independently to the left and right image of the stereo system and verify the potential detections by epipolar constraints.

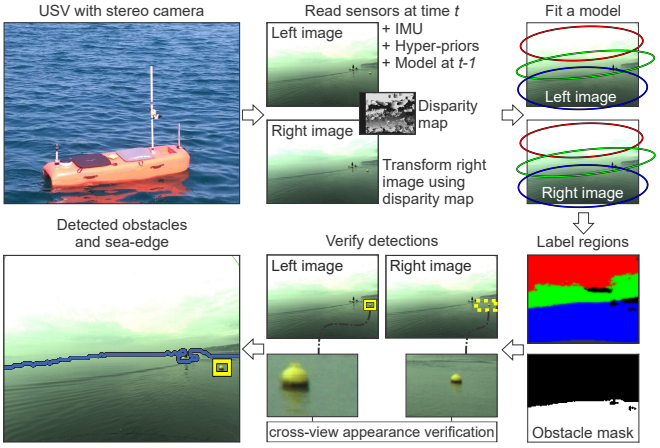

Fig. 1 *A semantic model is jointly fitted to stereo images, yielding a single pixel-wise posterior from which the list of potential obstacles are extracted and verified across the views.*

In this paper we improve the semantic segmentation obstacle detection approach [4] by enforcing consistency of the segmentation across the stereo views. A monocular graphical model [4] is extended to jointly consider pixel labels in stereo images and applies a disparity map as a weak correspondence constraint. An efficient EM algorithm for fitting the model simultaneously to the two views is derived. Since our approach considers a rectified stereo system, the epipolar lines in images are aligned which simplifies cross-view obstacle verification. Figure 1 outlines our approach. Our method outperforms the current state-of-the-art monocular USV obstacle detection ([4]), a monocular approach based on state-of-the-art convolutional neural network (CNN) [15], as well as their stereo-view extensions via [3], while running real-time in Matlab on a single CPU.

*This work was supported in part by the Slovenian research agency (ARRS) programmes P2-0214 and P2-0095, and the Slovenian research agency (ARRS) research project J2-8175.

[1]Borja Bovcon and Matej Kristan are with Faculty of Computer and Information Science, University of Ljubljana, Slovenia {borja.bovcon, matej.kristan}@fri.uni-lj.si

II. Related work

Obstacle detection for unmanned surface vehicles is still a relatively young research area. Most of the approaches that were developed for autonomous ground vehicles [5], [17] cannot be readily applied to the aquatic environment since they rely on estimation of the ground plane. Socek *et al.* [19] use a static camera and combination of background subtraction and motion cues to detect obstacles. Similarly Guo *et al.* [7] use an omni-directional camera and foreground extraction to detect moving obstacles in the water. However, such approaches fail to consider the visual properties of water and dynamic nature of weather and the marine environment, resulting in incorrect foreground extraction due to spurious differences between consecutive frames.

Wang *et al.* [21] proposed to combine saliency detection and motion estimation to detect and refine obstacles below the estimated water edge. Using a stereo camera system, epipolar constraints, and template matching, they search for correspondences of detected obstacles in the second view and triangulate their 3-D position. However, their assumption of a sharp boundary between water and sky when estimating the water edge is in practice often violated. In [20], Wang *et al.* propose to use a stereo camera system to perform 3-D reconstruction of the scene. They fit a plane to the 3-D reconstructed points corresponding to the sea, which enables them to detect obstacles above the water surface. This approach is capable of detecting only obstacles that significantly protrude through the water. Another problem arises on a calm sea, where water lacks a texture, thus leading to degraded 3-D reconstruction of the scene from corresponding stereo images and consequently to inaccurate water surface estimation.

The issues of using pre-calibrated stereo system and its lack of robustness to mechanical stress caused by open water, are addressed in the paper by Shin *et al.* [18]. They propose an automatic camera system calibration on open water with the help of detected horizon. The information about horizon is further used for fitting a plane on the water surface and extracting 3-D position of obstacles above it. Similar to [20], this approach does not account for smaller, partially-submerged obstacles and debris, which may damage small USVs. In addition, their assumption of sharp boundary between water and sky is in practice often violated.

The idea of combining semantic segmentation and stereo matching has been explored by Ladacky *et al.* [13] and Jafari *et al.* [10]. In [13] they propose using a conditional random field (CRF) for separate dense stereo reconstruction and object class segmentation. They present a principled energy minimization framework to unify the two problems and achieve state-of-the-art results on a street-view dataset. The authors of [10] propose using separate convolutional neural network methods to independently compute semantic segmentation and depth prediction on image. The obtained results are refined using their joint refinement network (JRN).

Recently, Kristan *et al.* [12] proposed a graphical model for monocular obstacle detection via semantic segmentation (SSM) of the observed marine scene. The algorithm generates a water-segmentation mask, and treats all blobs inside the water region as obstacles. The model assumes that an image of marine environment can be partitioned into three distinct and approximately parallel semantic regions: sky at the top, ground or haze in the middle, and water at the bottom. This semantic structure is enforced by fitting vertically-distributed Gaussian components and regularizing the result with a Markov random field. This approach successfully detects both obstacles protruding through the surface and the floating ones, it does not assume a straight water edge, and runs in real-time. Nevertheless, the SSM still fails in the presence of visual ambiguities. For example, when the boat faces open water and the horizon is obscured by haze, the SSM approach drastically over- or under-estimates the extent of the water region.

Bovcon *et al.* [4] address the shortcomings of the SSM [12] by introducing measurements from the on-board IMU into the segmentation model. The IMU measurements are used to project the horizon into camera view and automatically adjust the priors and hyper-priors of the segmentation model. The modified algorithm can correctly estimate the horizon even when obscured, and avoids over-estimating the extent of the water region by considering the prior information given by the estimated horizon. In their recent work [3] they address the problem of numerous false positive detections using a stereo system. They independently segment the left and right image, extract a list of potential obstacles and apply epipolar geometry and template matching for verification. In contrast, we propose a formulation for joint image segmentation, where corresponding pixels in the left and right image are assigned to the same semantic region, which consequently improves obstacle detection through enforced segmentation consistency.

III. Obstacle detection by stereo-view semantic segmentation

We briefly overview the monocular IMU-assisted semantic segmentation (ISSM) [4] in Section III-A and present our novel joint stereo segmentation formulation in Section III-B.

A. Single-view IMU-assisted semantic segmentation (ISSM)

Following the notation from [12], [4], [3], the input image is represented by an array of M pixel values $\boldsymbol{Y} = \{\boldsymbol{y}_i\}_{i=1:M}$, where $\boldsymbol{y}_i = (u_i, v_i, r_i, g_i, b_i)^T$ denotes a feature vector comprising the pixel's position (u_i, v_i) and color values (r_i, g_i, b_i).

The ISSM [4] segmentation model decomposes the image into three, approximately vertically-aligned regions (corresponding to water, land/haze and sky), and obstacles located in the water region. This is formalized by modeling the pixel values by a four-component mixture model

$$p(\boldsymbol{y}_i \mid \boldsymbol{\Theta}, \boldsymbol{h}_t, \boldsymbol{\varphi}_0, \boldsymbol{\pi}) = \sum_{k=1}^{3} \phi(\boldsymbol{y}_i \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\tilde{\boldsymbol{\pi}}_{ik} + \mathcal{U}(\boldsymbol{y}_i)\tilde{\boldsymbol{\pi}}_{i4}, \tag{1}$$

where $\phi(\cdot \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$ are Gaussian distributions corresponding to three semantic regions with means and covariances $\boldsymbol{\Theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1:3}$, $\mathcal{U}(\cdot)$ is a uniform distribution that

models the outliers and $\boldsymbol{\varphi}_0 = \left\{\boldsymbol{\mu}_{\mu_k}(\boldsymbol{h}_t), \boldsymbol{\Sigma}_{\mu_k}(\boldsymbol{h}_t)\right\}_{k=1:3}$ are hyper-priors on the Gaussian means, enforcing an approximately vertically aligned regions. The variables $\tilde{\boldsymbol{\pi}}_{ik}$ are pixel-wise class priors estimated from the current horizon line parameters $\boldsymbol{h}_t$, where subscript t denotes time. The horizon-dependent hyperpriors on Gaussian means in (1) are defined by a conjugate prior

$$p(\boldsymbol{\Theta} \mid \boldsymbol{h}_t, \boldsymbol{\varphi}_0) = \Pi_{k=1}^{3} \phi(\boldsymbol{\mu}_k \mid \boldsymbol{\mu}_{\mu_k}(\boldsymbol{h}_t), \boldsymbol{\Sigma}_{\mu_k}(\boldsymbol{h}_t)). \quad (2)$$

Smooth segmentation is encouraged by treating the priors $\boldsymbol{\pi}$ and posteriors $\boldsymbol{P} = \{\boldsymbol{p}_i\}_{i=1:M}$ over pixel class labels $\boldsymbol{x}_i$ as random variables, which form a Markov random field in [4]. The resulting graphical model is illustrated in Figure 2 and the corresponding joint probability density function, computed from (1) and (2), is written as

$$p(\boldsymbol{P}, \boldsymbol{Y}, \boldsymbol{\Theta}, \boldsymbol{\pi} \mid \boldsymbol{\varphi}_0, \boldsymbol{h}_t) \propto \exp\left(\sum_{i=1}^{M} \log p(\boldsymbol{y}_i, \boldsymbol{\Theta} \mid \boldsymbol{\varphi}_0, \boldsymbol{h}_t, \boldsymbol{\pi}) - \tfrac{1}{2}\left(E(\boldsymbol{\pi}_i, \boldsymbol{\pi}_{N_i}) + E(\boldsymbol{p}_i, \boldsymbol{p}_{N_i})\right)\right), \quad (3)$$

where $\boldsymbol{\pi}_{N_i}$ (and $\boldsymbol{p}_{N_i}$) are the averages of prior (and posterior) pixel-class distributions neighboring i-th pixel, respectively. The pair-wise potentials are defined by $E(\boldsymbol{\pi}_i, \boldsymbol{\pi}_{N_i}) = D(\boldsymbol{\pi}_i | \boldsymbol{\pi}_{N_i}) + H(\boldsymbol{\pi}_i)$, where $D(\cdot \mid \cdot)$ is the Kullback-Leibler divergence and $H(\cdot)$ is the entropy. Equation (3) is maximized by an EM-like algorithm [3].

1) Obstacle detection: The result of model fitting is an a posteriori probability distribution across the four semantic components for each pixel, denoted as $\hat{\boldsymbol{q}}_{ik}$. Pixels with maximal a posteriori probability for the water label (indexed as $k = 3$) are labeled as water and used to construct the water-region mask $\boldsymbol{B}_t = \{B_i\}_{i=1:M}$,

$$B_i = \begin{cases} 1, & \arg\max_k \hat{\boldsymbol{q}}_{ik} == 3 \\ 0, & \text{otherwise} \end{cases}, \quad (4)$$

where the largest connected component represents water and its upper edge corresponds to the edge of the water. The list of potential obstacles is obtained by extracting the blobs of non-water pixels.

B. Joint stereo view segmentation for obstacle detection

We improve on [4] by tightly coupling the segmentation process of the stereo views during the model fitting. Pixels in the left and right image corresponding to the same 3D structure should ideally be assigned to the same semantic class. Since the images from the left and right camera (i.e., $\boldsymbol{Y}^L$ and $\boldsymbol{Y}^R$) are rectified, the coupling is efficiently achieved by transforming $\boldsymbol{Y}^R$ into the coordinate frame of the left image, which leads to a single, left-image-aligned, posterior, as described below.

The right image $\boldsymbol{Y}^R$ is aligned by a mapping function $\boldsymbol{Y}^{R'} = f(\boldsymbol{Y}^R, \boldsymbol{d})$, which shifts pixels by their corresponding disparity values $\boldsymbol{d} = \{d_i\}_{i=1:M}$. The mapping quality depends on the estimated disparity quality from the two views, which is likely to mismatch the pixels in poorly textured (homogeneous) regions. Nevertheless, this is acceptable, since pixels of homogeneous regions correspond to the same semantic structure (in terms of our four classes) with a high probability, therefore the matching is correct at the level of the semantic label.

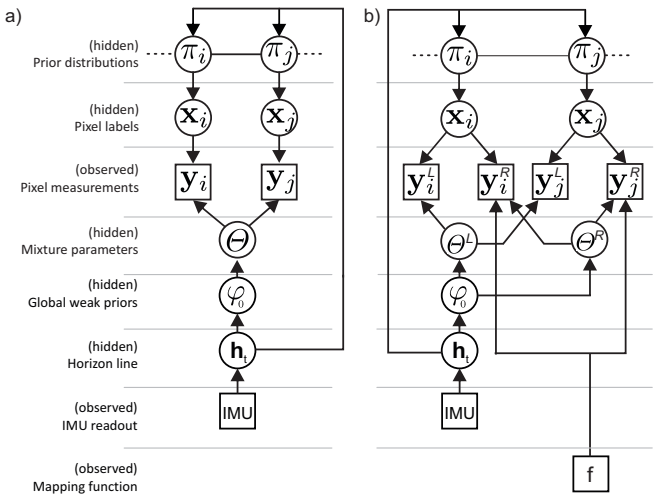

Fig. 2 *The MRF graphical model from single-view ISSM [4] (left) and our proposed modification of MRF graphical model for stereo-based semantic segmentation (right).*

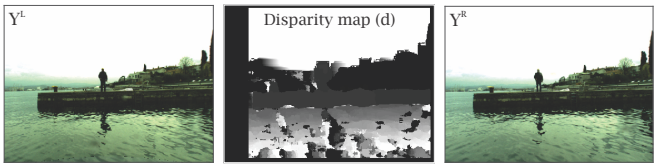

Fig. 3 *Left image $\boldsymbol{Y}^L$, right image $\boldsymbol{Y}^R$ and their corresponding disparity map $\boldsymbol{d}$.*

Assuming pixels in the left and right image are conditionally independent given the model parameters, the generative model (1) is rewritten into

$$p(\boldsymbol{y_i} \mid \boldsymbol{\Theta}, \boldsymbol{h}_t, \boldsymbol{\varphi}_0, \boldsymbol{\pi}) = p(\boldsymbol{y}_i^L \mid \boldsymbol{\Theta}^L, \boldsymbol{h}_t, \boldsymbol{\varphi}_0, \boldsymbol{\pi}) p(\boldsymbol{y}_i^{R'} \mid \boldsymbol{\Theta}^R, \boldsymbol{h}_t, \boldsymbol{\varphi}_0, \boldsymbol{\pi}) . \quad (5)$$

In our case $\boldsymbol{\Theta} = \left\{\boldsymbol{\Theta}^L, \boldsymbol{\Theta}^R\right\}$, where $\boldsymbol{\Theta}^L$ and $\boldsymbol{\Theta}^R$ denote the means and covariances of Gaussian kernels for the left and right image, respectively. This leads to a new graphical model (Figure 2) and the joint probability density function

$$p(\boldsymbol{P}, \boldsymbol{Y}, \boldsymbol{\Theta}, \boldsymbol{\pi} \mid \boldsymbol{\varphi}_0, \boldsymbol{h}_t) = p(\boldsymbol{P}, \boldsymbol{Y}^L, \boldsymbol{\Theta}^L, \boldsymbol{\pi} \mid \boldsymbol{\varphi}_0, \boldsymbol{h}_t) p(\boldsymbol{P}, \boldsymbol{Y}^{R'}, \boldsymbol{\Theta}^{R'}, \boldsymbol{\pi} \mid \boldsymbol{\varphi}_0, \boldsymbol{h}_t) , \quad (6)$$

with each of the right-hand-side terms defined as in (3). Following procedure of [12], the new posterior (6) is maximized by introducing auxiliary variables for priors and posteriors denoted as $\hat{\boldsymbol{s}}_i$ and $\hat{\boldsymbol{q}}_i$, respectively. This leads to an EM-like algorithm, where the E-step is applied to (5) and defined as:

$$\begin{aligned} \hat{\boldsymbol{s}}_{\cdot k} &= (\xi_{s\cdot} \circ \tilde{\boldsymbol{\pi}}_{\cdot k} \circ (\tilde{\boldsymbol{\pi}}_{\cdot k} \star \lambda)) \star \lambda_1, \\ \hat{\boldsymbol{q}}_{\cdot k} &= (\xi_{q\cdot} \circ \tilde{\boldsymbol{p}}_{\cdot k} \circ (\tilde{\boldsymbol{p}}_{\cdot k} \star \lambda)) \star \lambda_1, \\ \boldsymbol{\pi}_{\cdot k}^{\text{opt}} &= (\hat{\boldsymbol{s}}_{\cdot k} + \hat{\boldsymbol{q}}_{\cdot k}) p(\boldsymbol{x}_{\cdot} = k \mid \boldsymbol{h}_t)/4, \end{aligned} \quad (7)$$

where $\tilde{\boldsymbol{p}}_{\cdot k}$ is computed from (5), while $\circ$ and $\star$ denote the Hadamarad product and convolution, respectively. λ is a small discrete Gaussian kernel with its central element set to zero and its elements summing to one, $\xi_{s\cdot}$ and $\xi_{q\cdot}$ are the normalization constants, and $\lambda_1 = \lambda + 1$.

Using horizon-dependent hyper-priors, the M-step recomputes the means and variances of the Gaussians from (5) for each camera separately, but using the same pixel-wise posterior $\hat{\boldsymbol{q}}_i$. The means and variances for the left camera are thus recomputed as

$$\boldsymbol{\mu}_k^{L\,\mathrm{opt}} = \boldsymbol{\beta}_k^{-1}\left(\boldsymbol{\Lambda}_k^L\left(\sum_{i=1}^M \hat{\boldsymbol{q}}_{ik}(\boldsymbol{y}_i^L)^T\right)\left(\boldsymbol{\Sigma}_k^L\right)^{-1} - \boldsymbol{\mu}_{\mu_k}{}^T(\boldsymbol{h}_t)\boldsymbol{\Sigma}_{\mu_k}{}^{-1}(\boldsymbol{h}_t)\right)^T,$$
$$\boldsymbol{\Sigma}_k^{L\,\mathrm{opt}} = \boldsymbol{\beta}_k^{-1}\sum_{i=1}^M \hat{\boldsymbol{q}}_{ik}(\boldsymbol{y}_i^L - \boldsymbol{\mu}_k^L)(\boldsymbol{y}_i^L - \boldsymbol{\mu}_k^L)^T, \tag{8}$$

where $\boldsymbol{\beta}_k = \sum_{i=1}^M \hat{\boldsymbol{q}}_{ik}$ and $\boldsymbol{\Lambda}_k^L = ((\boldsymbol{\Sigma}_k^L)^{-1} + \boldsymbol{\Sigma}_{\mu k})^{-1}$. The means and variances for the right camera are recomputed similarly. The steps of our approach are summarized in Algorithm 1. The EM typically converges within a few iterations and results in the joint posterior $\hat{\boldsymbol{q}}_{\cdot k}$ over the semantic labels for each pixel. Obstacles and water edge are extracted from the posterior by the procedure described in Section III-A.1.

Algorithm 1 Fitting modified MRF graphical model to images

Require:
Pixel features $\{\boldsymbol{Y}^L, \boldsymbol{Y}^R\}$, disparity $\boldsymbol{d}$, horizon estimate $\boldsymbol{h}_t$, hyper-priors $\boldsymbol{\varphi}_0$, initial values for $\{\boldsymbol{\Theta}^L, \boldsymbol{\Theta}^R\}$ and $\boldsymbol{\pi}$.

Ensure:
Estimated parameters $\boldsymbol{\pi}^{\mathrm{opt}}$, $\boldsymbol{\Theta}^{L\,\mathrm{opt}}$, $\boldsymbol{\Theta}^{R\,\mathrm{opt}}$ and regularized posterior $\{\hat{\boldsymbol{q}}_{\cdot k}\}_{k=1:4}$.

1: Shift-align the right image: $\boldsymbol{Y}^{R'} = f(\boldsymbol{Y}^R, \boldsymbol{d})$.
2: Calculate the left and right image pixel posteriors $\boldsymbol{p}_{\cdot k}^L$ and $\boldsymbol{p}_{\cdot k}^R$ from $\boldsymbol{\pi}$, $\boldsymbol{\Theta}^L$ and $\boldsymbol{\Theta}^R$ following (1).
3: Compute the joint posterior (5): $\tilde{\boldsymbol{p}}_{\cdot k} = \boldsymbol{p}_{\cdot k}^L \circ \boldsymbol{p}_{\cdot k}^R$.
4: Compute the new priors $\boldsymbol{\pi}_{\cdot k}^{\mathrm{opt}}$ and posteriors $\hat{\boldsymbol{q}}_{\cdot k}$ by (7).
5: Reestimate the parameters ($\boldsymbol{\Theta}^L$,$\boldsymbol{\Theta}^R$) using (8).
6: Iterate steps 2 to 5 until convergence.

Cross-view appearance verification: Our semantic segmentation generates a list of potential obstacles, which might contain false positives caused by sun glitter, sea foam, lens flare, etc. Furthermore, the automatic camera exposure may also only partially segment smaller obstacles in one of the cameras, leading to poor localization.

These issues are addressed by enforcing cross-view appearance consistency of the potential obstacles by template matching. A template is extracted at a potential obstacle in the left image. Since the images are rectified, the search region is extracted in the right image at the same location and enlarged horizontally to account for the possible disparity. A normalized cross-correlation (NCC) is computed between the template and the search region and detection is accepted only if the maximum NCC exceeds a high threshold θ_{ncc}.

IV. Experimental evaluation

The dataset, the USV acquisition system and evaluation protocol are described in Section IV-A, implementation details are given in Section IV-B and detailed experimental analysis is reported in Section IV-C.

A. The dataset, acquisition system and evaluation protocol

The methods are analyzed on the Multi-modal obstacle detection dataset Modd2 [3], consisting of 11675 images captured from a small USV in the coastal waters of the Adriatic sea. The USV (Figure 1) uses a steerable thrust propeller with a small turn radius, and reaches a maximum velocity of $2.5\,\mathrm{m\,s^{-1}}$. It is equipped with an on-board computer, a compass, GPS unit, IMU unit and a pixel-synchronized stereo system *Vrmagic VRmMFC*, which consists of two *Vrmagic VRmS-14/C-COB* CCD sensors, *Thorlabs MVL4WA* lens with $3.5\,\mathrm{mm}$ focal length, maximum aperture of $\mathrm{f}/1.4$, and a 132.1° FOV. The stereo system is mounted $0.7\,\mathrm{m}$ above the water surface with baseline at $0.2\,\mathrm{m}$, resulting in depth estimation range up to $185\,\mathrm{m}$. The cameras capture rectified images at 10 frames per second.

The Modd2 contains obstacles and water edge manually annotated for each frame. The evaluation protocol of [3] measures the accuracy of the water-edge estimation by mean-squared error and standard deviation over all sequences, while the accuracy of obstacle detection is measured by the number of true positives (TP), false positives (FP), false negatives (FN) and by the overall F-measure, i.e., a harmonic mean of precision and recall. The water edge estimation is measured only in the left image to keep evaluation consistent between monocular and stereo systems.

B. Implementation and parameters

Our approach from Section III-B (denoted as IeSSM) was compared against four alternative state-of-the-art methods for obstacle detection and image segmentation in USVs: (i) a single-view ISSM [4], (ii) the most recent stereo-view ISSM [3] (sISSM), (iii) a convolutional neural network U-Net [15] applied only to the left camera and (iv) U-Net applied to both cameras and extended by our cross-view appearance verification method (sUNET).

For initialization of IeSSM we use the same initial parameters as in [3], while the hyper-priors were learned on a different USV dataset from [12]. The disparity map was computed at the full image size (1278×958) by block matching [11], with block size set to 51 pixels, disparity range set to the interval $[0, 128]$, contrast threshold set to one, and with uniqueness and texture thresholds both set to zero. Disparities that could not be reliably estimated were set to zero.

The scale factors for NCC verification were determined empirically on a separate training dataset [12]. The template region is expanded by a factor of 1.2 in all directions, while its corresponding search region is expanded by a factor of 1.2 in height and by a factor of 4 in width. The NCC acceptance threshold was set to a high, conservative value $\theta_{\mathrm{ncc}} = 0.95$.

The standard U-Net [15] architecture was trained on 223 images, sampled from dataset from [12]. Since U-Net has several size reduction layers, we resized the input images

TABLE I *We report water-edge estimation error (μ_{edg}) and its standard deviation, the number of true positive (TP), false positive (FP), false negative (FN) detections and the F-measure.*

	μ_{edg}	TP	FP	FN	F-measure
ISSM [4]	**0.056** (0.066)	538	1641	144	0.376
sISSM [3]	**0.056** (0.066)	504	**55**	178	0.812
UNET [15]	0.108 (0.096)	232	5461	447	0.073
sUNET [15]	0.108 (0.096)	222	346	457	0.356
IeSSM	0.058 (0.068)	**579**	68	**103**	**0.871**

according to [15] to 572×572 pixels to maximize performance accuracy.

The U-Net [15] was implemented in Python, while all other methods were implemented in Matlab R2016a. The experiments were run on a desktop computer with Intel Core i7-7700 3.6 GHz CPU and nVidia GTX970 GPU.

C. Results

The results are summarized in Table I. Our IeSSM achieves the highest number of true positive detections, along with the highest F-measure. It accurately detects 84.9% of all obstacles from the dataset. In F-measure, it significantly outperforms monocular approaches ISSM and UNET by 0.495 and 0.798, respectively, as well as their corresponding stereo applications sISSM and sUNET by 0.059 and 0.515, respectively. On the task of water edge estimation it achieves drastically better results than the CNN approach (UNET and sUNET), but slightly worse than ISSM and sISSM, which is due to higher sensitivity to the sun glitter.

Figure 4 shows a qualitative comparison. The first row of images shows problematic false positive detections caused by sea foam. The ISSM is unable to discard such detections, however methods using cross-view appearance verification (sISSM, sUNET and IeSSM) are capable of discarding most of false detections. In the second and third row of images there is a distant buoy which is undetected by ISSM, sISSM and sUNET, but IeSSM successfully and accurately detects it in both cases. We observe that the CNN approach (sUNET) drastically over-estimates the water-edge, while the estimation is much better for ISSM, sISSM and IeSSM.

1) Ablation study: The importance of the following parts of IeSSM was further studied: (i) image alignment by disparity map and (ii) cross-view appearance verification. The results are summarized in Table II.

We first test alternative pixel alignment functions $f(\cdot, \boldsymbol{d})$ from Section III-B. To disable the alignment, we set all values in the disparity map to zero (denoted as IeSSM_{ZD}). Table II shows a decrease in estimated water-edge accuracy (by 0.001) as well as in F-measure (by 0.025). This means that proper alignment plays an important role in obstacle detection. Using semi-global block matching [9] (IeSSM_{SG}) did not improve the results from Table I, while computing disparity on scaled images with block matching [11] (IeSSM_{RB}) led to sharp and pixelated disparity map resulting in poor water-edge estimation and obstacle detection.

TABLE II *Ablation study of IeSSM, where subscripts denote different switches. ZD, SG and RB are different pixel-alignment techniques, while NV denotes disabled cross-view appearance verification method.*

	μ_{edg}	TP	FP	FN	F-measure
IeSSM_{ZD}	0.059 (0.070)	574	73	108	0.864
IeSSM_{SG}	**0.057** (0.068)	571	97	111	0.846
IeSSM_{RB}	0.066 (0.074)	569	**34**	113	**0.886**
IeSSM_{NV}	0.058 (0.068)	**619**	3241	**63**	0.273

TABLE III *Speed analysis. The times required for segmentation, verification and the potential overall framerate are denoted as t_{seg}, t_{ver} and ω, respectively.*

	t_{seg} [ms]	t_{ver} [ms]	ω [fps]
ISSM [4]	**33.83**	**0**	**29.56**
sISSM [3]	67.65	7.67	13.28
UNET [15]	43	**0**	23.26
sUNET [15]	86	9.50	10.47
IeSSM	71.19	6.01	12.95

Next the cross-view appearance verification was disabled (IeSSM_{NV}). Comparing the results from Table I and Table II, we observe a significant increase in false positive detections (by 3173) and a slight increase in true positive detections (by 40). The F-measure score is drastically decreased (by 0.598) due to the increased number of false positive detections.

With exception of IeSSM_{NV}, each of the described partial implementations outperforms the state-of-the-art sISSM [3] on the task of obstacle detection.

2) Computational performance analysis: The potential processing speed of compared methods is presented in Table III. The monocular ISSM [4] operates with a single view and does not use any verification method, thus both its segmentation and verification process are the fastest. We witness an approximately $50\,\%$ speed reduction in the segmentation process when performing stereo segmentation with an additional slow-down due to cross-view appearance verification. The speed difference between sISSM [3] and IeSSM is minimal. Since on-board cameras are limited to 10 frames-per-second, all compared methods are considered real-time.

V. Conclusion

We presented a novel approach to stereo segmentation-based obstacle detection for unmanned surface vehicles. The proposed approach extends a state-of-the-art monocular MRF segmentation model from [4] to jointly consider segmentation of stereo images within a single framework. The approach applies disparity to align the left and right image, jointly fits the segmentation model to both images and estimates a single posterior over the semantic labels at

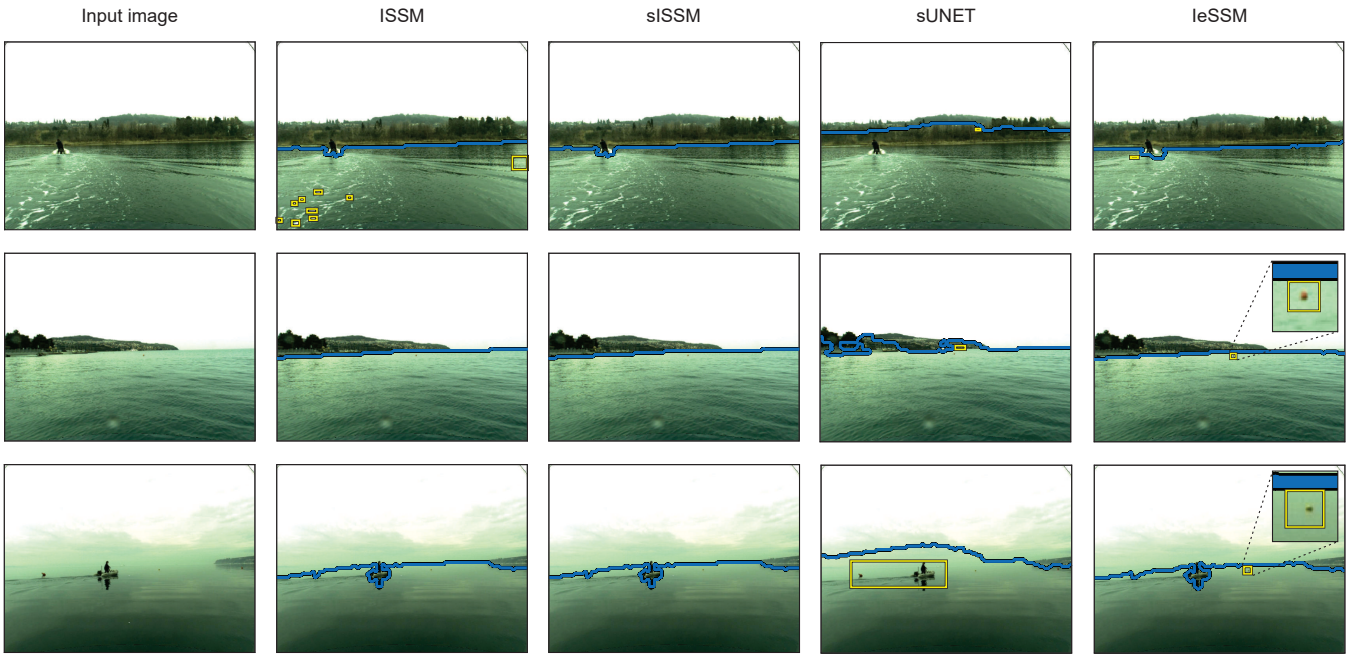

Fig. 4 *Qualitative comparison of observed methods on the task of water-edge estimation and obstacle detection. The estimated water edge is depicted by blue curve, while the obstacles in water region are depicted by yellow rectangles.*

each pixel. An efficient EM-like algorithm is derived for fast segmentation.

Experiments on a challenging dataset show that our approach (IeSSM) outperforms the most recent state-of-the-art stereo-view obstacle detector [3] by 0.059 in F-measure and can successfully detect approximately 11% more obstacles. The IeSSM also outperforms a state-of-the-art CNN for semantic image segmentation ([15]), adapted for stereo obstacle detection, on the task of water-edge estimation as well as obstacle detection. Results show that our approach runs in real-time on a single CPU.

Our future work will explore additional sensors, such as GPS and compass, as well as feature learning to further improve the segmentation process. The dynamic auto-exposure of cameras in the stereo system may affect the segmentation quality in special cases, which we plan to address with temporal consistency methods such as [2].

References

[1] C. Almeida, T. Franco, H. Ferreira, A. Martins, R. Santos, J. M. Almeida, J. Carvalho, and E. Silva. Radar based collision detection developments on usv roaz ii. In *OCEANS - EU*, pages 1–6, May 2009.

[2] N. Bonneel, J. Tompkin, K. Sunkavalli, D. Sun, S. Paris, and H. Pfister. Blind video temporal consistency. *ACM TOG*, 34(6), 2015.

[3] B. Bovcon, R. Mandeljc, J. Pers, and M. Kristan. Stereo obstacle detection for unmanned surface vehicles by imu-assisted semantic segmentation. *RAS*, 104:1–13, 2018.

[4] B. Bovcon, R. Mandeljc, J. Perš, and M. Kristan. Improving vision-based obstacle detection on usv using inertial sensor. In *ISPA*, pages 1–6, Sept 2017.

[5] A. Broggi, C. Caraffi, R. I. Fedriga, and P. Grisleri. Obstacle detection with stereo vision for off-road vehicle navigation. In *CVPR*, pages 65–65, June 2005.

[6] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE TPAMI*, 40(4):834–848, 2018.

[7] Y. Guo, M. Romero, S. H. Ieng, F. Plumet, R. Benosman, and B. Gas. Reactive path planning for autonomous sailboat using an omni-directional camera for obstacle detection. In *ICM*, pages 445–450, 2011.

[8] H. K. Heidarsson and G. S. Sukhatme. Obstacle detection and avoidance for an autonomous surface vehicle using a profiling sonar. In *ICRA*, pages 731–736, May 2011.

[9] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *PAMI*, 30(2):328–341, Feb 2008.

[10] O. H. Jafari, O. Groth, A. Kirillov, M. Y. Yang, and C. Rother. Analyzing modular cnn architectures for joint depth prediction and semantic segmentation. In *ICRA*, pages 4620–4627. IEEE, 2017.

[11] K. Konolige. Small vision systems: Hardware and implementation. In *Robotics research*, pages 203–212. Springer, 1998.

[12] M. Kristan, V. S. Kenk, S. Kovačič, and J. Perš. Fast image-based obstacle detection from unmanned surface vehicles. *IEEE TCYB*, 46(3):641–654, 2016.

[13] L. Ladickỳ, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. Clocksin, and P. H. Torr. Joint optimization for object class segmentation and dense stereo reconstruction. *IJCV*, pages 1–12, 2012.

[14] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[15] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241. Springer, 2015.

[16] A. R. J. Ruiz and F. S. Granja. A short-range ship navigation system based on ladar imaging and target tracking for improved safety and efficiency. *ITS*, 10(1):186–197, March 2009.

[17] I. Shim, J. Choi, S. Shin, T.-H. Oh, U. Lee, B. Ahn, D.-G. Choi, D. H. Shim, and I.-S. Kweon. An autonomous driving system for unknown environments using a unified map. *ITS*, 16(4):1999–2013, 2015.

[18] B.-S. Shin, X. Mou, W. Mou, and H. Wang. Vision-based navigation of an unmanned surface vehicle with object detection and tracking abilities. *MVA*, pages 1–18, 2017.

[19] D. Socek, D. Culibrk, O. Marques, H. Kalva, and B. Furht. A hybrid color-based foreground object detection method for automated marine surveillance. *LNCS*, 3708:340, 2005.

[20] H. Wang and Z. Wei. Stereovision based obstacle detection system for unmanned surface vehicle. In *ROBIO*, pages 917–921, 2013.

[21] H. Wang, Z. Wei, C. S. Ow, K. T. Ho, B. Feng, and J. Huang. Improvement in real-time obstacle detection system for usv. In *ICARCV*, pages 1317–1322, 2012.