Active Mapping of Underwater Litter Using Camera-Sonar Fusion

David Reţe
Department of Automation
Technical University of Cluj-Napoca
Romania
dvdrete@gmail.com

Patrick Boros
Department of Automation
Technical University of Cluj-Napoca
Romania
patrick.boros@gmail.com

Lucian Buşoniu
Department of Automation
Technical University of Cluj-Napoca
Romania
Corresponding Member of the Romanian Academy
Bucharest, Romania
lucian.busoniu@aut.utcluj.ro

***Abstract*—Marine litter is a growing threat to the underwater ecosystem, driving demand for autonomous survey methods that can locate debris efficiently over large areas. Existing survey methods typically follow predefined paths or operate with a single sensing modality, typically a camera (with image quality suffering in poor-visibility conditions) or sonar (usually noisy and low-resolution). We present an active mapping framework in which a forward-looking sonar and a camera both feed into a shared Bayesian occupancy map, and an optimization problem is solved at each step to decide on the next best view. Candidate viewpoints are scored by a two-term utility that balances exploration of uncertain regions via voxel entropy against exploitation of likely objects. Each sensor is characterized by range- and bearing-dependent detection and false-alarm probability tables determined from data. We evaluate the approach in a realistic underwater simulator, demonstrating that active mapping finds objects faster than a lawnmower coverage pattern, and that the dual-sensor approach works better than using either of the individual sensors.**

***Index Terms*—Active sensing, next best view, underwater robotics, occupancy grid mapping, marine litter detection.**

I. Introduction

Underwater robots face significant sensing limitations. Cameras suffer from poor optical visibility due to turbidity and inconsistent lighting. Sonars do not need light and are less affected by turbidity, but are often noisier and have poorer resolution than cameras; higher-frequency sonars improve these aspects somewhat at the expense of a narrower field of view, shorter range, and often prohibitive financial costs. These sensing constraints are especially problematic for marine-litter surveys, as debris is often small and/or visually ambiguous. Furthermore, following preplanned paths to search for litter can be a waste of time because it often leads to re-observing already confirmed areas while failing to resolve regions that require additional observations. This motivates active litter mapping, where an underwater robot adaptively chooses its next move so as to concentrate sensing effort on the most informative areas of the seafloor. These challenges are all encountered e.g. in the SeaClear and SeaClear2.0 projects [7], which focus on a multi-robot system for detection, mapping, and collection of marine litter.

This work was been financially supported from SeaClear2.0, a project that received funding from the European Climate, Infrastructure and Environment Executive Agency (CINEA) under grant agreement No 101093822.

Motivated by the limitations of individual sensors and the need for active mapping, we propose in this paper a method that maintains a litter occupancy map built using Bayesian updates that fuse camera and sonar detections, and selects the next-best-view waypoint that maximizes mapping progress. The maximization objective balances uncertainty reduction (exploration) against confirmation of likely litter (exploitation). In the realistic MARUS underwater simulator, we experimentally show that an uncrewed underwater vehicle (UUV) using our active mapping method finds litter faster than when it uses a uniform-coverage lawnmower pattern; and that fusing the two sensors works better than using either of them individually.

In related work, information-driven planning for robotic mapping and exploration has been extensively studied for terrestrial, aerial, and marine robots, typically using information objectives like entropy reduction. E.g., reference [2] formulates active 3D mapping as maximization of mutual information over a candidate set of trajectories that combines global shortest-path plans with local motion primitives, followed by gradient-based trajectory optimization that refines the selected trajectory in continuous control space. At a broader level, the survey [1] categorizes information-driven planners along two axes: the map representation (Gaussian processes for continuous spatial fields vs. occupancy grids for structural mapping) and the planning horizon (myopic single-step vs. non-myopic multi-step). Our work falls within the occupancy-grid, myopic-planner branch. Differently from most classical methods, which seek to reduce overall map uncertainty – via e.g. mutual information, map entropy, or unseen voxel count – our two-term objective explicitly separates exploration (entropy of uncertain seafloor voxels) from exploitation (count of voxels that belong to likely but unconfirmed objects). This better reflects our litter-search task where confirming litter matters more than obtaining a uniformly low uncertainty across the entire map.

Methods that balance exploitation and exploration include e.g. [13], which proposes an informative path planner for an aerial drone that combines Gaussian-process maps with an

evolutionary trajectory optimizer. The planner reduces overall map uncertainty while using confidence-based level sets to prioritize regions of interest. This line of work is extended in [16] with a deep reinforcement learning approach, achieving much faster replanning than optimization. Also related is the field of multitarget search and tracking [3], [4], [21], where a very different type of target representation is used, based on intensity functions. Most such approaches focus on tracking targets without exploration, but others do include exploration terms [22].

Considering now specifically marine robotics, [20] addresses tracking of freely drifting surface targets with an ASV using a single stereo camera, by combining a dynamic occupancy grid with a spatiotemporal prediction neural network that estimates future target positions under wind-driven drift, feeding a two-term utility that balances entropy reduction with a redetection reward. In our problem, litter is static on the seafloor. Active viewpoint planning with single acoustic sensors for classifying already detected targets has been studied in [17], [18].

Our key novelty with respect to the work above is that we close the loop between dual, camera-sonar sensor fusion and an active search algorithm for underwater target discovery. The closest related work that uses camera-sonar fusion is in underwater SLAM [14], where the robot's pose is not known, so the algorithm focuses heavily on using the map to reduce pose uncertainty, rather than to search for specific targets.

Next, Section II outlines occupancy grids and neural networks for computer vision, while Section III details our method. Experimental results are given in Section IV, and Section V concludes the paper.

II. Background

A. Occupancy-grid mapping

We represent the seafloor map as a 3D occupancy grid, consisting of voxels that we index by an integer i for simplicity of notation. Each voxel is associated with a Bernoulli distribution with probability $b_i \in [0, 1]$ of being occupied. The real map is $m_i \in \{0, 1\}$, where $m_i = 1$ means that voxel i is occupied. Bayesian updates of the voxel grid are performed based on uncertain sensor measurements z_i about m_i for all voxels i in the field of view of the sensor, starting from some prior initial values, which may be taken 0.5 if we do not have any information in advance:

$$b_i' = \frac{P(z_i \mid m_i = 1) \cdot b_i}{P(z_i \mid m_i = 0)(1 - b_i) + P(z_i \mid m_i = 1) \cdot b_i} \tag{1}$$

where the probabilities $P(z_i \mid m_i)$ describe sensing uncertainty.

It will be useful to compute the entropy h_i of each voxel i:

$$h_i = -\left(b_i \cdot \log(b_i) + (1 - b_i) \cdot \log(1 - b_i)\right) \tag{2}$$

which is a measure of the uncertainty with which the state of this voxel is known.

The specific implementation that we use is the octree-based Octomap [6], which provides an efficient multi-resolution representation of the 3D environment.

B. Deep learning for computer vision

Deep learning methods are widely used for computer vision tasks such as image classification, object detection and segmentation. Convolutional neural networks learn hierarchical feature representations from raw sensor data, enabling extraction of semantic and spatial information [9]. Image classification assigns a single semantic label to an image, providing scene understanding but not spatial localization [8]. Object detection extends this by identifying and localizing multiple objects within the scene allowing for variable object counts [15]. Segmentation methods further extend spatial understanding by operating at the pixel level [12]. While semantic segmentation assigns class labels to pixels, instance segmentation separates individual object instances, which is useful in robotic perception scenarios. In this paper, we use the Mask R-CNN model for instance segmentation, leveraging its two-stage detection architecture, which first generates region proposals for candidate objects and then performs class-specific mask prediction [5].

III. Active dual-sensor underwater mapping

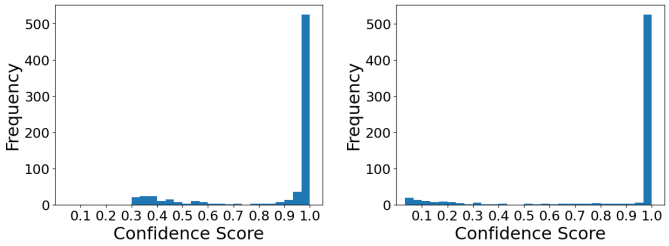
All experiments reported in this paper are performed in the MARUS underwater simulation environment [11], a Unity-based simulator developed for marine robotics. The simulator provides realistic sensor image rendering and vehicle dynamics; specifically, we use the MiniTortuga UUV [19], which is equipped with a forward-looking camera and sonar. The simulated scene consists of a static seafloor environment containing multiple objects (cubes and spheres) representing seafloor litter, see Figure 3 below for some views of the scene.

Next, we describe the components of our active search framework: instance segmentation in Section III-A, sensor models in Section III-B, map updates in Section III-C, and the active search planner Section III-D.

A. Instance segmentation from sonar and camera images

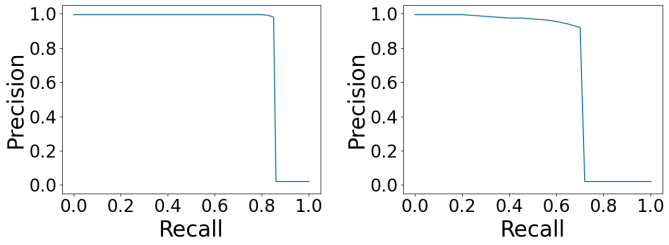
To perform instance segmentation (including classification) of litter objects, we choose Mask R-CNN with a ResNet-50 backbone and Feature Pyramid Network [5], initialized from COCO-pretrained weights [10]. This model is selected as it is well-established and stable. Separate models are trained for the sonar and RGB modalities, with no architectural modifications.

To train the two models, we generated synthetic datasets for both sonar and camera sensors in the MARUS simulator, and used standard training/validation/test data splits. For the sonar dataset, a custom Sonar3D sensor was configured to render forward-looking sonar images. Each dataset contains frames with different object configurations, including scenes with no objects, with a single object, and with multiple objects from both the cube and sphere classes. This variability is essential for characterizing missed detections and false results, and ensures that the network produces meaningful outputs across a broad range of scenes. After constructing the images, we determined camera or sonar hit points belonging to each object instance, and computed their convex hull to generate polygonal object outlines representing ground-truth segmentation masks.



(a) RGB camera (b) Sonar

Fig. 1: Confidence score distributions for the camera and sonar networks evaluated on test data.



(a) RGB camera (b) Sonar

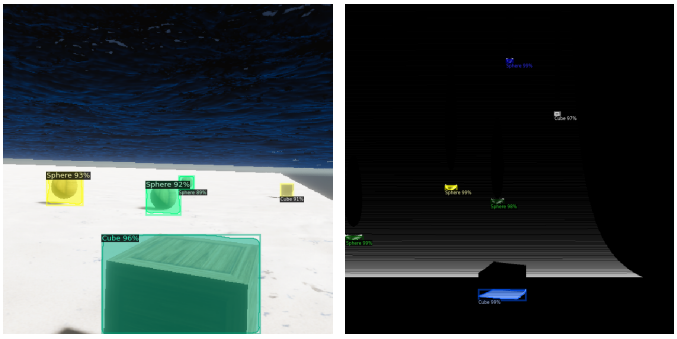
Fig. 2: Precision-recall curves for the camera and sonar networks evaluated on test data.

The resulting images were stored together with their corresponding ground-truth annotations, including the object class (cube or sphere) and the polygon vertex coordinates.

Results from test data shows that confidence is highly concentrated near one, as illustrated by the confidence score distributions in Figure 1. To understand Figure 2, note first that precision measures the ratio of predicted detections that are correct: $\frac{TP}{TP+FP}$, where TP and FP respectively count true and false positives. Recall measures the ratio of correctly detected objects to the total number of ground-truth objects: $\frac{TP}{TP+FN}$, where FN is the number of false negatives. The precision-recall curve in Figure 2 shows the trade-off between precision and recall as the detection confidence threshold varies. The behavior is near-ideal: for most of the recall range, the precision value remains close to the maximum, while getting true positive detections. This behavior shows a strong score difference between true objects and background clutter, with false positives introduced only when the confidence threshold is driven very low to achieve very large recall.

Figure 3 illustrates image and sonar detections. Note that bounding boxes are shown for identified classes, but behind the boxes, more precise segmentation masks are visible.

This very good performance is expected given our simulated problem, but is unrealistic for real underwater sensing, where environmental effects and sensor imperfections introduce variability. To bridge the gap between this idealized behavior and realistic sensing uncertainty, we used a temperature scaling parameter applied at inference time to degrade the confidence. The segmentation model generates for each pixel logits ℓ_c that represent relative confidence for each class c, and the scalar



(a) RGB camera (b) Sonar

Fig. 3: Instance segmentation outputs for the camera and sonar networks evaluated on test data.

temperature parameter T is applied to the softmax operation to regulate the sharpness of the resulting confidence distribution across classes c:

$$P_c = \frac{\exp(\ell_c/T)}{\sum_{c'} \exp(\ell_{c'}/T)} \tag{3}$$

Increasing the temperature parameter value makes the confidence scores more uniform and reflects a more realistic, higher uncertainty.

B. Sensor models

Let q denote the UUV pose. The pose of each sensor $s \in \{\text{camera}, \text{sonar}\}$ is obtained by a fixed rigid-body transform $q_s = T_s\, q$, where T_s encodes the known mounting offset. To determine measurements z_i for each voxel i, a ray is cast from the sensor origin through the center of the voxel. If that ray intersects the image plane inside a segmentation mask for an object, then $z_i = 1$; otherwise, i.e. if the intersection is outside all the segmentation masks, then $z_i = 0$.

We assume that the UUV pose q and therefore the sensor poses q_s are sufficiently well known to disregard their uncertainty, and model sensing uncertainty as follows:

$$\begin{aligned} P_s(z_i = 1 \mid m_i = 1) &= \hat{t}_s(q_s, c_i) \\ P_s(z_i = 1 \mid m_i = 0) &= \hat{f}_s(q_s, c_i) \end{aligned} \tag{4}$$

where of course $P_s(z_i = 0 \mid m_i) = 1 - P_s(z_i = 1 \mid m_i)$. Here, $\hat{t}_s$ and $\hat{f}_s$ are (approximate) probabilities of a true positive and of a false positive (false alarm), respectively, represented as functions of the sensor pose q_s and the center c_i of voxel i. Since our UUV stays horizontal and maintains the same altitude above the seafloor, in practice we will need only a relative pose represented by a scalar range $r(q_s, c_i)$ and bearing $\theta(q_s, c_i)$.

We generated two types of simulated datasets in MARUS to learn the geometry-dependent sensor models for both the sonar and camera. We have target-present runs, used to estimate the true-positive model $\hat{t}_s$; and empty-scene experiments, in which the target object is removed and any detection will be labeled as a false alarm, to estimate the false-positive model $\hat{f}_s$ that captures background clutter and spurious detections.

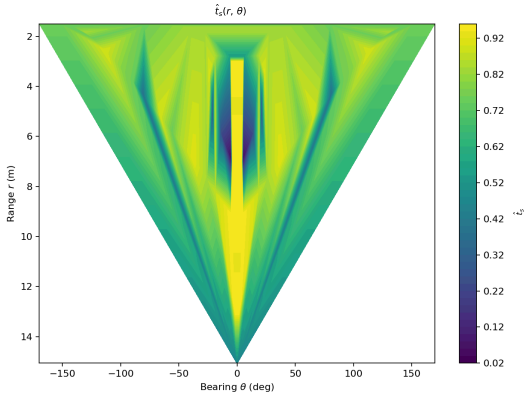


Fig. 4: Estimated true-positive probabilities for the camera

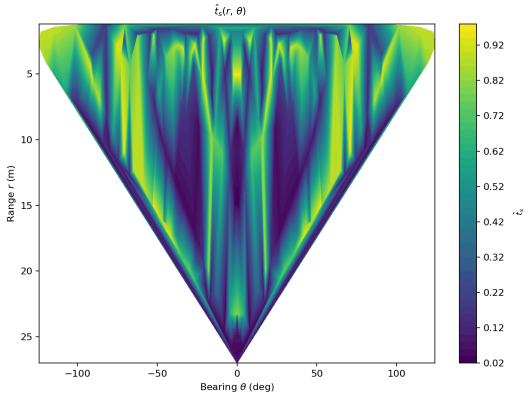


Fig. 5: Estimated true-positive probabilities for the sonar

For both sensors, data are collected by executing sweeps in which the UUV follows a motion trajectory designed to sample a wide range of relative viewing geometries. In the target-present experiments, the object is placed at fixed known location on the seafloor, and the UUV performs repeated yaw rotations combined with forward-backward translations or circular orbits to vary range and bearing with respect to target. The same motion pattern is replayed for the empty scene to ensure that the distribution of measurements is matched between the positive and negative datasets. At each step, the corresponding sensor frame (sonar polar image or RGB camera image) is recorded together with the measurement geometry.

For each sensor/dataset, the trained Mask R-CNN model is applied independently to every recorded sensor image, producing a binary measurement, indicating whether the target class is detected above a fixed confidence threshold, chosen as 0.8. Each detection is correlated with the corresponding geometry. The measurements are then discretized into bins over these geometric variables, and within each bin the empirical detection frequency is computed as a ratio of positive detections to the total number of samples. This process yields the desired geometry-dependent estimates $\hat{t}_s$ and $\hat{f}_s$. The resulting probability tables are visualized in Figure 4 for the camera and Figure 5 for the sonar. The figures use a heatmap representation that encodes how the sensor's reliability varies as a function of viewing geometry, providing a visual representation of regions with high and low detection likelihood.

C. Map and updates

Motivated by the fact that in our real SeaClear system we have access to a bathymetric model of the seafloor, we work here under the same conditions: the seafloor geometry is known as a point cloud representation, which we discretize into voxels of length 0.2m on each side. It is however unknown which voxels correspond to clean seafloor and which to litter objects. The goal is to determine this from sensor readings.

Experiments may use either a single sensor (sonar or camera), or both sensors jointly. For the latter case, because the sonar and camera operate asynchronously at different frequencies, updates may arrive from one or both sensors in a given cycle. Rather than constructing an explicit joint sensor fusion model, we treat the two sensing modalities as conditionally independent given the state of the map. When a single-sensor measurement is received, we simply run (1) while plugging in $\hat{t}_s$ and $\hat{f}_s$ for that sensor. When both sensors deliver measurements in the same cycle, the updates are applied sequentially, with the posterior from the sonar serving as the prior for the camera; the same formula (1) is used, but now twice with the two sensor models.

D. Active search

Classically, the search for objects is done in a coverage pattern such as a lawnmower or a spiral that investigates the relevant area uniformly. Since searching with the robot costs time and energy, our objective is however to find objects faster than a coverage pattern, by focusing early on promising regions. To this end, we define a set of waypoints W and at each active search step k, we choose an optimal next waypoint by solving by enumeration the following optimization problem:

$$w_{k+1} \in \operatorname{argmax}_{w \in W} \sum_{i \in \phi(w)} \left(h_i + \alpha I(b_i \in [\underline{b}, \overline{b}]) \right) \quad (5)$$

Let us explain the elements of the objective function in turn:

- $\phi(w)$ denotes the field of view: the set of voxels i whose centers fall within the fields of view of the sensors used. Note that when we use both the camera and the sonar, ϕ is the union of the two sensors' fields of view.
- The *exploration* term h_i gives priority to voxels with larger entropy, i.e. larger uncertainty about whether they contain objects or not. If this were the only component, a coverage-like pattern would be followed to uniformly reduce the entropy over the space. However, we want to prioritize object candidates, so we add the next component.
- The *exploitation* term focuses on refining object candidates, and is given by an indicator function $I(b_i \in [\underline{b}, \overline{b}])$, which is 1 if and only if b_i is in the required interval; otherwise, it is 0. Lower bound $\underline{b} > 0.5$ sets the probability above which a voxel is considered to be part of an

object candidate, while upper bound $\bar{b} < 1$ ensures the algorithm does not unnecessarily focus on voxels that are already clearly part of an object.

- Weight α trades off exploration versus exploitation, and is the key tuning parameter of the algorithm. A smaller α focuses more on covering unseen areas, while a larger one places focuses on refining object candidates.

Each waypoint w is specified by its coordinates x, y on a plane at a given depth, together with the yaw ψ of the UUV. Once a waypoint w_{k+1} has been chosen, the UUV navigates there at a fixed velocity and samples each sensor used in the experiment at its own frequency, updating the map after each sample as expained in Section III-C above.

IV. Results

This section presents and discusses the results of our active mapping method from joint camera and sonar observations. The simulated scene covers a rectangle of size $[5, 38] \times [0, 24]\,\mathrm{m}$ and contains 11 objects grouped into three clusters: four cubes in the upper-left corner of the search area (1st cluster), four spheres in the upper right (2nd cluster), and three cubes in the lower right (3rd cluster). The operating depth is $z_{\mathrm{op}} = -18\,\mathrm{m}$. The exploitation term in (5) uses an interval $[\underline{b}, \bar{b}] = [0.7, 0.95]$, and the weight α is set to 5. The camera operates at $15\,\mathrm{Hz}$ with a $60°$ vertical FOV, and the forward-looking sonar operates at $20\,\mathrm{Hz}$ with a $120° \times 60°$ FOV and a maximum range of $30\,\mathrm{m}$. The UUV travels at a nominal velocity of $0.7\ \mathrm{m/s}$, consistent with typical UUV survey operations.

We perform two different experiments. In the first, using both the camera and sonar, we compare active search against a uniform-coverage lawnmower pattern, to check whether the method indeed finds objects faster. In the second experiment, we just use active search and compare the joint-sensor performance with sonar-only and camera-only performance. An object is considered to have been detected when the occupancy probability of at least one voxel whose center is within the ground-truth volume of the object increases above $\bar{b}$.

For the first experiment, the lawnmower follows a fixed predefined pattern with $12\,\mathrm{m}$ strip spacing. This is a wide spacing that still covers the whole search area in a small number of passes (specifically, 3 passes). For active search, the grid of waypoints is constructed by uniformly discretizing the search area with grid spacing $\Delta_x = \Delta_y = 6\,\mathrm{m}$ in the plane, and 8 uniformly spaced heading angles at $\Delta_\psi = 45°$ intervals in yaw. Figure 6 plots the number of detected objects with respect to the distance traveled. The active search strategy is clearly better, since by $27\,\mathrm{m}$ it has already found all 11/11 objects (all 3 clusters), compared with the 4 objects found by the lawnmower. The lawnmower takes $110\,\mathrm{m}$ to find all the objects. Figure 7 illustrates the trajectories performed by the UUV when using the two strategies. Note that active mapping finds the third cluster without having to travel directly on top of it.

For the second experiment, because the camera has a shorter range than the sonar, to get usable results from camera-only

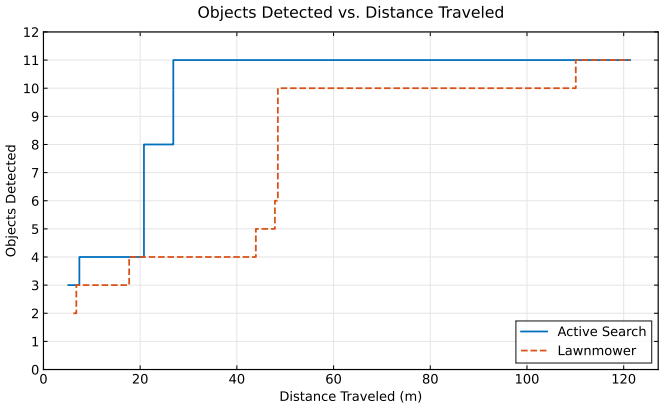


Fig. 6: Active mapping vs. lawnmower mapping using both sensors

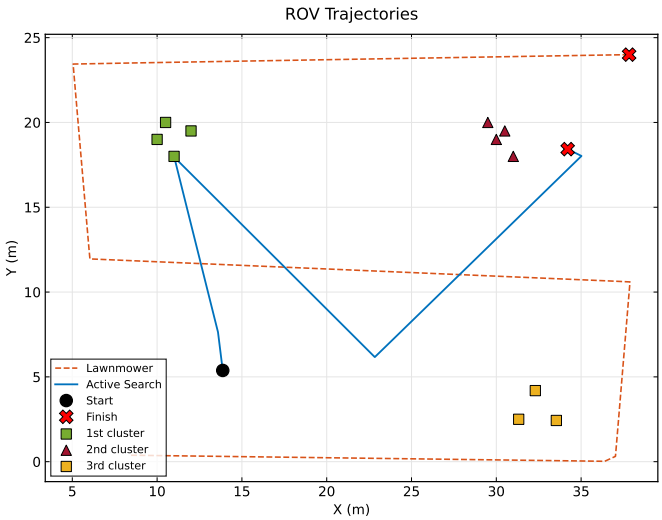


Fig. 7: UUV trajectories with the two mapping strategies

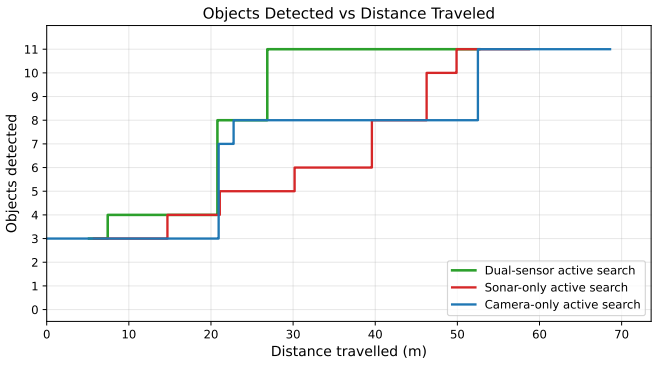


Fig. 8: Dual-sensor versus single-sensor performance during active mapping.

measurements we lowered the grid spacing to $\Delta_x = \Delta_y = 3\,\mathrm{m}$. The rest of the parameters remain the same. The results of this second experiment are shown in Figure 8, where active search with sensor fusion is roughly twice faster in finding all the objects than either the sonar or camera separately. There is no clear winner among the single-sensor setups, further illustrating that the sensors are complementary and should be used together.

Finally, we verify the robustness of the method in a different scenario where the object clusters are aligned at the 'top' of the scene. Figure 9 shows that active search maintains its advantage.

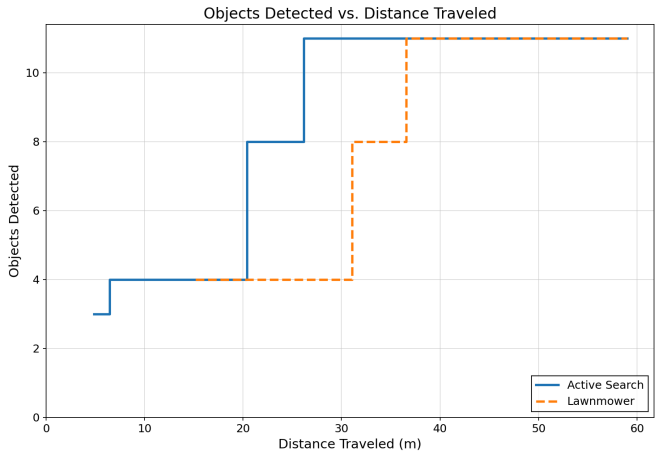


Fig. 9: Active mapping vs. lawnmower mapping for a second scenario

V. Conclusions

We presented a method for active, next-best-view search for underwater litter that fuses observations from a forward-looking sonar and a camera into a shared probabilistic map, using sensor models that depend on the range and bearing of the object relative to the sensor. Viewpoints are selected by maximizing an objective function that balances exploration of uncertain regions with exploitation of likely object candidates. Experiments are conducted in a realistic underwater environment simulator, and show that active mapping substantially outperforms predefined fixed-pattern coverage, and that fusing complementary sensor modalities works better than either sensor used separately.

Future work includes developing a joint sensor model that includes couplings between sonar and camera likelihoods; investigating alternate active mapping objectives and non-myopic planners; as well as evaluating the active search pipeline in field trials as a part of the SeaClear2.0 project.

References

[1] S. Bai, T. Shan, F. Chen, L. Liu, and B. Englot, "Information-driven path planning," *Current Robotics Reports*, vol. 2, no. 2, pp. 177–188, 2021.

[2] B. Charrow, G. Kahn, S. Patil, S. Liu, K. Goldberg, P. Abbeel, N. Michael, and V. Kumar, "Information-theoretic planning with trajectory optimization for dense 3D mapping," in *Robotics: Science and Systems*, vol. 11, 2015, pp. 3–12.

[3] P. Dames and V. Kumar, "Autonomous localization of an unknown number of targets without data association using teams of mobile sensors," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 3, pp. 850–864, 2015.

[4] P. M. Dames, "Distributed multi-target search and tracking using the PHD filter," *Autonomous Robots*, vol. 44, no. 3, pp. 673–689, 2020.

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2961–2969.

[6] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, 2013.

[7] A. Ilioudi, S. Sosnowski, E. Banken, P. Bevanda, J. Brüdigam, L. Buşoniu, Y. Chardard, C. Delea, B. De Schutter, A. Djuras, C. Hertel-ten Eikelder, S. Heshmati-Alamdari, V.-M. Maer, I. Palunko, I. Pozniak, V. Prkacin, and D. Tolic, "The SeaClear system: An intelligent multi-robot solution for autonomous cleanup of marine debris on the seabed," *Engineering Applications of Artificial Intelligence*, vol. 170, p. 114094, 2026.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, 2012.

[9] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[10] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014, pp. 740–755.

[11] I. Loncar, J. Obradovix, N. Krasevac, L. Mandic, I. Kvasic, F. Ferreira, V. Slosic, D. Nadj, and N. MiSkovic, "Marus-a marine robotics simulator," in *OCEANS 2022, hampton roads*. IEEE, 2022, pp. 1–7.

[12] J. Long, E. Shelhamer, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[13] M. Popović, T. Vidal-Calleja, G. Hitz, J. J. Chung, I. Sa, R. Siegwart, and J. Nieto, "An informative path planning framework for UAV-based terrain monitoring," *Autonomous Robots*, vol. 44, no. 6, pp. 889–911, 2020.

[14] S. Rahman, A. Quattrini Li, and I. Rekleitis, "SVIn2: A multi-sensor fusion-based underwater SLAM system," *The International Journal of Robotics Research*, vol. 41, no. 11-12, pp. 1022–1042, 2022.

[15] 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*, 2015.

[16] J. Rückin, L. Jin, and M. Popović, "Adaptive informative path planning using deep reinforcement learning for UAV-based active sensing," in *2022 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 4473–4479.

[17] A. V. Sethuraman, P. Baldoni, K. A. Skinner, and J. McMahon, "Learning which side to scan: Multi-view informed active perception with side scan sonar for autonomous underwater vehicles," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 8348–8354.

[18] J. Shin, S. Chang, J. Weaver, J. C. Isaacs, B. Fu, and S. Ferrari, "Informative multiview planning for underwater sensors," *IEEE Journal of Oceanic Engineering*, vol. 47, no. 3, pp. 780–798, 2022.

[19] Subsea Tech, "Mini TORTUGA – inspection class ROV datasheet," https://www.subsea-tech.com/mini-tortuga/.

[20] S. R. Sudha, M. Popović, and E. M. Coates, "An informative planning framework for target tracking and active mapping in dynamic environments with ASVs," *IEEE Robotics and Automation Letters*, vol. 11, no. 3, pp. 2690–2697, 2026.

[21] Y. Sung and P. Tokekar, "GM-PHD filter for searching and tracking an unknown number of targets with a mobile sensor with limited FOV," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 2122–2134, 2021.

[22] B. Yousuf, R. Herzal, Z. Lendek, and L. Buşoniu, "Multi-agent active multi-target search with intermittent measurements," *Control Engineering Practice*, vol. 153, p. 106094, 2024.