Engineering Applications of Artificial Intelligence 170 (2026) 114094

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Case report

The SeaClear system: An intelligent multi-robot solution for autonomous cleanup of marine debris on the seabed

Athina Ilioudi [a,*], Stefan Sosnowski [b], Elisabeth Banken [c], Petar Bevanda [b], Jan Brüdigam [b], Lucian Buşoniu [d], Yves Chardard [e], Cosmin Delea [c], Bart De Schutter [a], Antun Đuraš [f], Claudia Hertel-ten Eikelder [g], Shahab Heshmati-Alamdari [h], Vicu-Mihalis Maer [d], Ivana Palunko [f], Iva Pozniak [i], Vicko Prkačin [f], Domagoj Tolić [f,j]

[a] Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, Delft, 2628 CN, Netherlands
[b] Department of Electrical and Computer Engineering, Technical University of Munich, Arcisstr 21, Munich, 80290, Germany
[c] Fraunhofer Center for Maritime Logistics and Services CML, Blohmstraße 32, Hamburg, 21079, Germany
[d] Department of Automation, Technical University of Cluj-Napoca, Memorandumului 28, Cluj-Napoca, 400114, Romania
[e] SUBSEA TECH, 167, plage de l'Estaque, Marseille, 13016, France
[f] LARIAT-UNIDU — Laboratory of intelligent and autonomous systems at University of Dubrovnik, Ćira Carića 4, Dubrovnik, 20000, Croatia
[g] Hamburg Port Authority AöR, Neuer Wandrahm 4, Hamburg, 20457, Germany
[h] Department of Electronic Systems, Aalborg University, Fredrik Bajers Vej 7, Aalborg, 9220, Denmark
[i] Regional Development Agency Dubrovnik-Neretva County, Ul. branitelja Dubrovnika 41, Dubrovnik, 20000, Croatia
[j] RIT Croatia, Don Frana Bulića 6, Dubrovnik, 20000, Croatia

ARTICLE INFO

Keywords:
Multiple autonomous underwater vehicles
Computer vision
Deep learning
Marine debris
Control
Mapping

ABSTRACT

Marine debris poses an alarming threat to ocean environments. Conventional methods of sea and ocean cleaning rely heavily on manual collection, a process that has repeatedly demonstrated its inefficiency and extensive demand for resources. This paper presents the SeaClear system, a novel multi-robot platform designed to autonomously detect and collect marine debris, thereby offering a more efficient solution to this environmental challenge. An overview of the system is presented, followed by a detailed description of each robot's capabilities. Leveraging artificial intelligence, the system employs the deep-learning-based computer vision algorithm You Only Look Once (YOLO) for the detection of underwater litter, addressing the challenges of poor visibility and hydrodynamic disturbances of underwater environments. Additionally, the paper explores the implemented navigation and control methodologies, which are an essential part of the workflow of the system. The performance of the designed system is validated via field tests conducted in a real-world underwater environment. Finally, directions for future work are proposed.

1. Introduction

Marine environments are increasingly threatened by the accumulation of waste resulting from human activities. Plastic is the largest and most harmful fraction of marine waste, accounting for around 85% of total marine waste (McGlade et al., 2021; Isobe and Iwasaki, 2022). It is estimated that 11 million tons of plastic waste enter the ocean each year (McGlade et al., 2021). Interestingly, the majority of plastic entering the oceans (94%) ends up on the seafloor instead of the sea surface Sherrington (2016). The severity of the situation is underscored by multiple studies warning that the volume of marine debris may continue to increase, potentially surpassing the total weight of fish in the sea by 2050 if no drastic action is taken (World Economic Forum and Ellen MacArthur Foundation and McKinsey & Company, 2016).

Parallel to this environmental threat, the rapid advancements in artificial intelligence and robotics have led to their extensive application outside the lab for complex, real-world problems. Despite their potential though, the implementation of these technologies for addressing the urgent issue of marine pollution, particularly litter on the seabed, remains a challenging and largely unexplored area. The primary reasons hindering the employment of subsea robots for debris collection are related to the nature of underwater environments. Disturbances caused by hydrodynamic forces and external factors can heavily affect the performance of robotic vehicles, especially in cases that require high operating precision. Another major challenge is related to the poor visibility conditions underwater due to light absorption, scattering, and

* Corresponding author.
E-mail address: a.ilioudi@tudelft.nl (A. Ilioudi).

https://doi.org/10.1016/j.engappai.2026.114094
Received 27 May 2024; Received in revised form 31 October 2025; Accepted 3 February 2026
Available online 15 February 2026

Fig. 1. Overview of the SeaClear robotic platform.

reflection. Finally, underwater communication and sensing are imposing limitations on localization and control in the field of underwater robotics.

Artificial Intelligence (AI) adoption in underwater robotics applications could play a pivotal role in addressing the issue of marine litter. This article introduces the SeaClear system, an autonomous multi-robot platform engineered for the collection of marine debris from the seafloor. This system stands as a significant application of AI in the field of marine robotics, offering a novel solution to an increasingly concerning environmental problem. Our objective is to explain the key features of the SeaClear system and workflow, as well as to validate its feasibility and practical performance through results from field experiments. Specifically, the contributions of the current paper are:

- The SeaClear system is described: a multi-robot solution that integrates aerial, surface, and underwater vehicles to efficiently and autonomously detect, map, and collect marine debris from the seafloor.
- A sensing and control scheme is carefully designed, developed, and integrated with the hardware. To solve practical challenges that arise, novel methodological components are proposed, including e.g. an intermittent-measurement pose estimator and a procedure to locally reacquire visual contact with the litter during collection.
- We show that the state-of-the-art computer vision algorithm YOLO is suitable for real-time detection and classification of underwater litter. The adaptability and performance of this algorithm under the challenging conditions of underwater environments are assessed, providing valuable insights from its deployment in real-world scenarios.
- We confirm the practical performance of the SeaClear system. This is done by validating the system through a battery of real-world field experiments.

The paper is organized as follows: Section 2 gives an overview of the existing related work in this field. Section 3 presents the overall architecture of the platform and an in-depth description of each robot constituting the system. In Section 4 the proposed workflow of the SeaClear system is presented and analyzed. Section 5 presents the results of the simulation validation and the field experiments. Finally, Section 6 gives concluding remarks and directions for future work.

2. Related work

Despite the growing threat posed by marine litter, there are only a few integrated solutions to address it. The majority of the current strategies primarily focus on preventive measures and pollution monitoring (Watanabe et al., 2019; Babić et al., 2023), with the cleanup approaches typically targeting the sea's surface and the water column. In Cózar et al. (2014), the first global map of plastic waste in the ocean is presented. A three-dimensional real-time detection system of marine plastic litter is presented in Yang et al. (2025), which is based on deep learning techniques. An approach to detect and identify submerged debris in marine environments using forward-looking sonar imagery and convolutional neural networks is presented in Valdenegro-Toro (2016). In Jia et al. (2023), various deep-learning algorithms performing the task of visually detecting trash in underwater environments are evaluated. However, all these solutions only focus on the detection and mapping of the litter, without proposing an approach for litter collection. The employment of robotics to facilitate comprehensive solutions to the identification, gathering, and elimination of marine waste is largely concentrated on the collection of surface marine litter González-Morgado et al. (2025). These efforts include, e.g., the Ocean Cleanup (The Ocean Cleanup) initiative, which has developed an approach based on natural, oceanic currents to passively collect plastic debris in floating garbage patches. More specifically, their system comprises a floating barrier for capturing waste and a central platform that functions as a processing plant to extract, sort, and recycle the collected plastic. In addition, in Kong et al. (2021), a robot system is developed to collect floating garbage from the water surface. A deep-learning-based detection framework is adopted for the floating litter detection, while a sliding-mode control approach is implemented for the steering of the robots. In Zhang et al. (2023), an unmanned surface vehicle is proposed for autonomous water quality monitoring as well as for water surface litter detection and collection. A color-based object detection algorithm is employed for the detection of floating litter in combination with sensor fusion to accomplish the various functions of water quality monitoring and litter cleaning. Finally, a net is mounted to the surface vehicle to collect the detected floating litter.

Furthermore, the organization SeaCleaners (The SeaCleaners) has developed a specifically designed vessel to collect floating plastic debris. This vessel is also capable of removing larger pieces of floating debris from the water, such as nets or containers lost by other vessels, using two cranes. In addition, Ranmarine (RanMarine Technology) has introduced an autonomous lightweight vessel called Watershark, inspired by whale sharks. The Watershark navigates the water collecting plastics and microplastics, alien vegetation, and floating debris.

The task of underwater litter collection is currently primarily performed by human divers. In addition, the EU projects MAELSTROM (Gouttefarde et al., 2023) and Natural Seabed (Natural Seabed) are developing large cable robots[1] to pick up underwater litter, reducing dependence on heavy and expensive marine support. However, Natural Seabed's robotic system is not autonomous, and both solutions focus on large cable robots; therefore, they are not very agile and adaptable. Hence, these systems are constrained when employed in complex topographies, such as ports in which marine debris tends to accumulate, and where large vessels and platforms cannot easily enter or be installed. Furthermore, a prototype for an underwater litter

[1] Cable robots are a type of robotic manipulators where the end-effector is manipulated by multiple cables controlled by winches.

(a) SeaCat USV. (b) Collection basket. (c) Observation (mini Tortuga) ROV.

(d) Collection (Tortuga) ROV. (e) Aerial drone UAV.

Fig. 2. SeaClear system modules.

collection robot has been developed within the Rozalia Project (The Rozalia Project). This robot is operated remotely by humans. Finally, in Wang et al. (2023), a jellyfish-inspired robotic platform is presented, integrating electro-hydraulic actuators and a hybrid structure of rigid and soft components. The robot can perform a wide range of functions for diverse applications, including litter collection. This is not an integrated approach for cleaning up the sea though, as this 16 cm-diameter robot can only collect a small volume of litter, and it does not involve a stand-alone large-sized collection basket for depositing the litter.

Contrary to these solutions, the SeaClear system introduces an autonomous, integrated approach to detect and collect marine litter, featuring an adaptable architecture. To the best of our knowledge, SeaClear is the first system that consolidates all these advanced functionalities into a single solution. This emphasis on integration, adaptability, and automation significantly enhances the current state-of-the-art in marine litter collection technologies.

3. SeaClear system description

The SeaClear architecture (Fig. 1) has been developed with the aim to search for and collect marine waste in an autonomous way. The system consists of five main components: an USV that scans the sea bottom, a collection basket for storing the litter, an observation ROV that searches for litter underwater, a collection ROV that collects the litter, and an UAV, which searches for litter from the air.

Although some hardware components like sensory equipment and the aerial drone are standardized products available on the market, other parts, such as the underwater and surface vehicles, the gripper, the collection basket, as well as the respective launch and recovery systems, are specifically developed towards the SeaClear purposes and tailored to fulfill the task of underwater litter clean-up efficiently. This section describes each robotic component comprising the SeaClear system. The tables with the technical details can be found in Appendix.

3.1. Robotic modules

3.1.1. Unmanned surface vehicle - SeaCat USV

The SeaCat USV (Fig. 2(a)) is a sea-going unmanned surface vehicle originally developed for offshore wind farm inspections. It is 6.83 m in length, 3.1 m in width, and 2.15 m in height when inflated. The SeaCat USV features a range of sensors, such as a high-resolution multibeam echosounder, a Differential Global Positioning System (GPS), a Real-Time Kinematics (RTK) GPS, and sensor data interfaces including Serial, USB, and Ethernet ports, facilitating various connectivity options with the rest robotic vehicles. It serves as a central platform, offering the capability to deploy not only the other three robotic vehicles, but also a collection basket for placing the litter. Each component is tethered to the SeaCat and released by an individual Launch and Recovery System (LARS). Importantly, the SeaCat is equipped with a high-resolution multibeam echosounder. This enables it to perform bathymetry scans of the seafloor, creating accurate 3D models of the seabed. Large underwater litter may already be identified through this initial survey.

3.1.2. Collection basket

The collection basket (Fig. 2(b)) is tethered to the SeaCat USV and is lowered to the seafloor with an individual winch from the rear part of the USV. Once on the seafloor, the basket also serves as an anchor to the SeaCat USV, while the two underwater ROVs perform their tasks in detecting and collecting litter. The basket itself is designed to hold up to approximately $250\,\mathrm{kg}$ or $1\,\mathrm{m}^3$ litter per clean-up operation and is made from aluminum struts to provide sufficient stability. These struts are connected via a net with approximately 2 cm large meshes to present less risk of accidentally entrapping wildlife. Due to its funnel-shaped entrance and its bristled interface, it allows precise alignment of the collection ROV and a safe deposit of litter without the risk of it floating out of the basket again. The localization of the basket underwater is achieved through fusing data using a Kalman Filter from the included sensors on the basket, such as the Inertial Measurement Unit (IMU), the depth sensor, and the SBL localization sensor. In addition, the basket is also equipped with lights and ArUco markers for a more precise positioning of the collection ROV during the close-range maneuvering. An integrated camera inside the basket enables the operator to observe the litter deposit process, assess the litter intake capacity, and identify potential problems. For areas with dense marine litter pollution, where a collection rate of $60 \times 2\,\mathrm{L}$ per hour is needed, the total volume of the collected waste would be under the $1\,\mathrm{m}^3$ capacity of the basket (see Fig. 3).

3.1.3. Observation ROV - mini Tortuga ROV

The Observation (mini Tortuga) ROV, depicted in Fig. 2(c), is a lightweight inspection robotic vehicle that employs several sensors for underwater debris detection. This sensing system aims to distinguish among debris, fauna, and flora, ensuring that only debris is collected without harming any living organism. It combines the use of a conventional full-HD video camera with high-power LED lights to survey the surroundings primarily in clear or low-turbidity water, and an integrated dual-frequency multibeam imaging sonar to augment visibility under poor light conditions. The task of the Observation ROV

(a) Individual modules in an unloaded state.

(b) Modules when loaded on the SeaCat USV.

Fig. 3. SeaClear system components.

is to create a map of the detected litter from these sensors. During this task, the previously generated map from the SeaCat USV is further refined. The Observation ROV is connected to the SeaCat USV via a tether, more specifically, a power line communication cable. The winch for the Observation ROV is located at the bow of the SeaCat USV, where the launch and recovery are performed.

3.1.4. Collection ROV - Tortuga ROV

The Collection (Tortuga) ROV is a larger (inspection class) ROV designed to operate in strong currents and to collect debris from the seafloor thanks to an embedded gripper, as can be seen in Fig. 2(d). Similarly to the Observation ROV, the Collection ROV also features a computer-vision-based system. The Collection ROV locates debris through an integrated downward-facing camera in the gripper, which provides images of the trash to be picked up and guides the robot to the right spot for grasping it. The operation of the Collection ROV shares similarities with its smaller counterpart, the Observation ROV. The Collection ROV is also tethered to the SeaCat USV by a winch, which is situated at the rear of the SeaCat USV.

The Collection ROV collects the waste and transfers it to the collection basket. The collection mechanism consists of a gripping and a suction tool that are capable of collecting various types of litter. The suction device is aimed to assist the gripper during debris collection for diverse object types, such as plastic wrappers or bags.

Two critical considerations that have been taken into account during the design phase of the collection mechanism, include: 1. the requirement for a skid to contain the gripper and the suction device and to be mechanically connected to the Collection ROV, and 2. the need that all components minimize their negative hydrodynamic influence to retain good controllability of the Collection ROV in water. Given these considerations, the gripping and the suction tools have been designed such that they consist of two shovels for litter collection and a pump with a hose that can draw objects towards the suction cup. The gripper, featuring a streamlined and bio-inspired design, was developed and manufactured using a metal 3D printing process. This allowed the

creation of a mesh-like structure from aluminum–magnesium alloys that achieves a balance between strength and weight, while also allowing any small animals that were accidentally trapped to escape. With an inner volume of 155 cm^3 and a size of $24 \times 20 \times 12.5$ cm, the gripper can fully enclose typical half-liter bottles or cans or grasp larger objects with the front edges.

The gripper and the suction device can be operated independently of each other through the Collection ROV's communication interface. The two shovel-like segments of the gripper are driven by a single motor controlled by a motor driver that is connected to the Collection ROV. The motor and its driver are contained inside a casing on which the gripping segments are mounted. The inputs to the gripper are generated by the automatic controller, although it is also possible to provide manual input.

3.1.5. Aerial drone - DJI Matrice 210 RTK V2 UAV

The UAV surveys the area in which the robotic platform operates and enables detection and mapping of locations with larger litter concentrations that are visible from the air (Sukno and Palunko, 2022). This provides initial information for a more detailed search performed by the underwater robots in the next step. The detected litter is placed in the map used to guide the Observation ROV so as to perform close-up scans. The UAV is depicted in Fig. 2(e).

In conditions where the waters are transparent, the UAV can also contribute to the localization of the ROV. This can be particularly useful given the fact that the underwater robots work in a GPS-denied environment and hence the UAV serves as an alternative source for the position information of the ROV. In this way, when the ROV is detected near the surface, it gets supplied with a GPS estimate based on the drone's RTK-GPS and the localization in its camera frame, improving the ROV's localization accuracy.

Following the initial free-flight survey, the UAV is connected to the SeaCat USV through a tether. The 40 m long tether enables continuous air operation without the need to land by supplying power and maintaining a reliable communication channel. At the same time, it imposes a constraint on the movement of the UAV, necessitating the coordination with the USV. The UAV LARS is the key component in this coordination task. Situated high at the SeaCat USV bow, it consists of a landing platform, a cable winch, and associated electronics. Hence, it functions as the power and communication interface between the aerial and surface vehicle.

4. Proposed system workflow & methodology

4.1. Overview

The mode of operation for the SeaClear system is depicted in Fig. 4. Complementary figures in the manuscript provide a representation of the main subsystems introduced in this overview, including the ROV position estimation scheme (Fig. 5), the mapping framework (Fig. 8), the control structure of the Collection ROV (Fig. 10), the control scheme of the tethered UAV (Fig. 13), and the overall software architecture (Fig. 14). When the system reaches the designated area for its mission, the SeaCat USV establishes an initial bathymetry scan of the sea floor. Large pieces of litter may already be detected in this step. If the water is clear enough, the aerial drone searches from above. The information of the USV and the aerial drone is then refined into a map, which is sent to the Observation ROV, equipped with the sonar and video imaging systems. The images captured by the Observation ROV are analyzed using object detection algorithms to differentiate between live organisms and litter fractions. The latter are marked in the initial map, which is constantly refined as the Observation ROV follows a lawnmower trajectory while searching the area. After that, the Collection ROV is deployed, navigating to the litter items marked on the map. It then collects the detected litter and deposits it in the collection basket. The litter that cannot be collected and deposited into the collection basket, e.g., due to its large size, remains marked in the map for collection with suitable equipment later on. Once the collection process is finished, all vehicles tethered to the SeaCat USV are autonomously recovered from the water and the SeaClear robotic platform returns to the shore.

4.2. Pose estimation

The pose (i.e. the position and the orientation) of the ROVs is the most challenging to estimate, since they travel underwater. We rely on sensor fusion between their IMUs, Doppler Velocity Loggers (DVLs) mounted on their undersides, and an absolute position sensor. For the latter, underwater acoustic sensors are expensive and sometimes unavailable, so as an alternative, we find the global ROV position using the camera image of the UAV (Đuraš et al., 2022). This is of course only possible when the ROV is close enough to the surface to be detected in the UAV image stream, so an interesting intermittent-measurement estimation problem arises (Liu and Goldsmith, 2004; Maer et al., 2022). We first describe the positioning algorithm in the UAV image, followed by the overall pose estimator that fuses all the available sensors.

An object tracking algorithm is executed on the UAV camera image to output the pixel position of the ROV, which is then used to obtain the normalized camera-referenced coordinates ${}^{C}\overline{\mathbf{X}}$, assuming calibrated intrinsic parameters of the camera. Further along the position estimation pipeline (Fig. 5) the UAV altitude (a_{UAV}) and the ROV depth (d_{ROV}) are combined to obtain the vertical distance to the ROV. In NED (North-East-Down) coordinates, the ROV operating plane is defined by the point $\mathbf{p}_0$ on the plane and the normal vector $\mathbf{n}$ perpendicular to it:

$$\mathbf{p}_0 = (0, 0, a_{\text{UAV}} + d_{\text{ROV}})^{\text{T}} \tag{1}$$

$$\mathbf{n} = (0, 0, 1)^{\text{T}} \tag{2}$$

We transform $\mathbf{p}_0$ and $\mathbf{n}$ from NED to the camera coordinate frame using the rotation matrix ${}^{C}R \in SO(3)$ obtained from the camera orientation angles reported by the gimbal:

$${}^{C}\mathbf{p}_0 = {}^{C}R\,\mathbf{p}_0 \qquad {}^{C}\mathbf{n} = {}^{C}R\,\mathbf{n}.$$

Intersecting the vector representing the ROV position in normalized camera coordinates ${}^{C}\overline{\mathbf{X}}$ with the ROV operating plane defined by $({}^{C}\mathbf{p}_0, {}^{C}\mathbf{n})$ allows us to recover the distance d of the ROV to the camera's optical center:

$$d = \frac{{}^{C}\mathbf{p}_0 \cdot {}^{C}\mathbf{n}}{{}^{C}\overline{\mathbf{X}} \cdot {}^{C}\mathbf{n}} \tag{3}$$

Metric coordinates of the camera-referenced ROV position ${}^{C}\mathbf{P}$ are then obtained from the normalized coordinates as:

$${}^{C}\mathbf{P} = d\,{}^{C}\overline{\mathbf{X}}. \tag{4}$$

Finally, the GPS position and georeferenced orientation of the drone are used in order to transform the camera-referenced position of the ROV into a global position estimate. For a more comprehensive description and details of transformations between coordinates, readers are encouraged to refer to the original paper (Đuraš et al., 2022).

To fuse all the sensors (onboard and, when available, UAV-based), the ROV performs pose estimation using an extended Kalman filter. The detailed implementation is described in Moore and Stouch (2016), and uses a kinematic model with 6 degrees of freedom for the robot. The equations for the model and the extended Kalman filter updates are standard, so they will not be given here, with one exception: due to the intermittent nature of the UAV-based position measurement, the measurement equation of the system switches between two possibilities. It is given by:

$$\mathbf{y}_k = C_k \mathbf{x}_k + \upsilon_k \tag{5}$$

where $\mathbf{x}_k$ is the state vector (consisting of 3D linear and angular positions and velocities, together with linear accelerations), υ_k is measurement noise, and $\mathbf{y}_k$ is the measurement. Importantly, the output

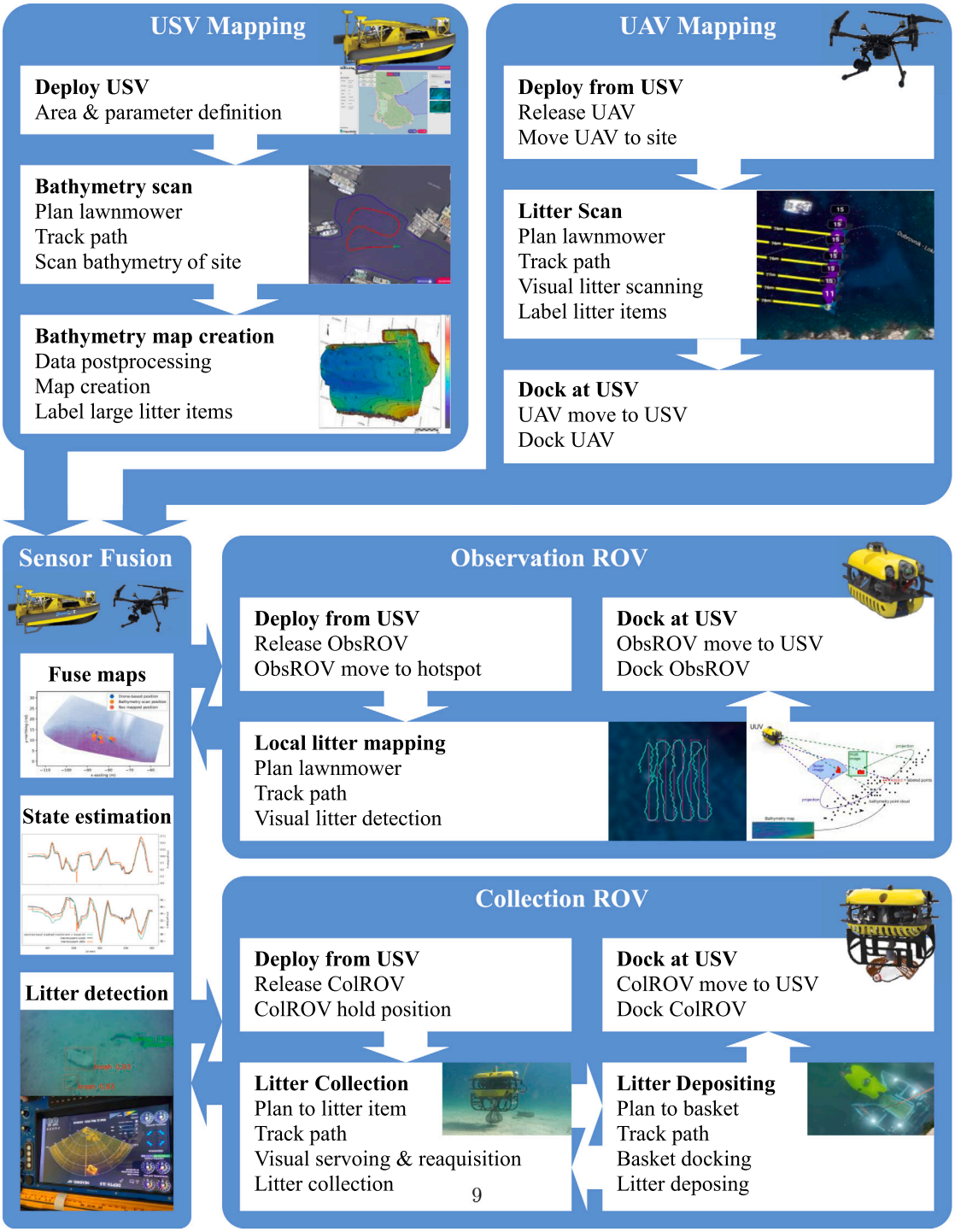


Fig. 4. Overview of the SeaClear mode of operation.

matrix C_k takes one of two values: C_o or C_f. Option C_o corresponds to on-board sensors only, so measurements include angular position and velocity from the IMU, linear velocity from the DVL, and vertical position from a water pressure sensor. Option C_f is applied at those time steps k where the UAV-based position is available. It adds two extra rows to C_o, corresponding to the position in the plane from the UAV. The output and noise dimensions increase accordingly. The performance of the estimator is evaluated later on, in Section 5.

4.3. Aerial and underwater mapping

4.3.1. Aerial debris detection

Detecting submerged marine debris from UAVs poses significant challenges stemming from various factors. The efficient absorption of light by water causes rapid attenuation with depth. This issue of limited visibility is further exacerbated by surface wave motion, resulting in motion blur and distorted images. Additionally, the considerable

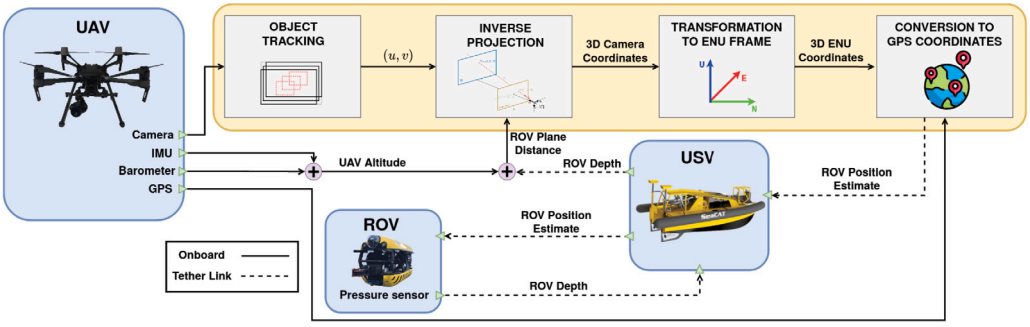


Fig. 5. Scheme of the ROV position estimation from the UAV adapted from (Đuraš et al., 2022). The ROV reports its own current depth, necessary to compute the ROV position estimate onboard the UAV. This estimate is then reported back to the ROV and used as an external input for the full ROV pose estimation algorithm.

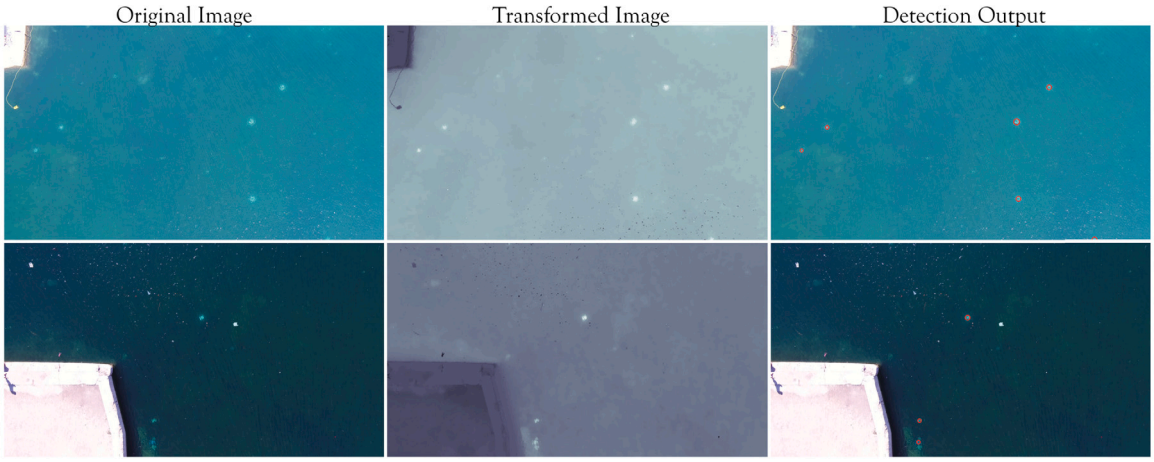


Fig. 6. Visualization of the inputs and outputs of the aerial marine debris detection pipeline. From left to right, the figure shows the captured input images, the transformed images utilized as an input for the blob detection phase, and the final detection outputs visualized on the original images.

distance between the UAV and the submerged objects often results in captured images lacking details, with objects being represented merely as small blobs. Moreover, the phenomenon of glare, caused by reflections at the water–air interface, introduces specular highlights in images, potentially leading to erroneous detection of non-existent objects.

Since there are no publicly available datasets that would facilitate learning-based methods for underwater object detection from aerial cameras, we utilize a learning-free approach based on a Difference of Gaussians (DoG) blob detector. We handcraft the inputs of the blob detection method based on experimentally observed image statistics by taking the exponential of the difference between the blue/green color channel and the red color channel:

$$I_{b-r} = \exp({}^{b}I - {}^{r}I), \tag{6}$$

$$I_{g-r} = \exp({}^{g}I - {}^{r}I), \tag{7}$$

where ${}^{c}I$ denotes the intensity of the color channel $c \in \{r, g, b\}$ of the image I.

This allows us to produce the blob detector input as a transformed single-channel image $I_{\text{input}} = \frac{(I_{b-r}+I_{g-r})}{2}$, which can be interpreted as a score map where high values indicate the possible presence of a submerged object. Intuitively, this approach based on the difference of color channels utilizes the fact that for submerged objects wavelengths in the red range are attenuated rapidly. As seen in the transformed images shown in Fig. 6, applying this transformation helps to eliminate surface objects and specular highlights (typically saturated in all color channels of the original image) which could interfere with the blob detection algorithm further along the pipeline.

Blob detection can be formulated as the task of finding the local extrema in the image scale-space (Lindeberg, 1998). The scale-space representation of an image is obtained by the convolution of an image $I(x, y)$ with a variable-scale Gaussian kernel $G(x, y, s)$:

$$G(x, y, s) = \frac{1}{2\pi s} \exp \frac{x^2 + y^2}{2s}$$

The scale-space can thus be defined as the following function:

$$L(x, y; s) = I(x, y) * G(x, y, s) \tag{8}$$

where $*$ represents the convolution operator.

For a scale-space representation obtained at a specific scale s, applying the Laplacian operator $\nabla^2 L = L_{xx} + L_{yy}$ results in a strong response for blobs of radius $r^2 = 2s$. Typically, multi-scale detection is achieved by simultaneously considering local extrema with respect to both space and scale by using the scale-normalized Laplacian:

$$\nabla^2_{norm} L(x, y; s) = s\nabla^2 L(x, y; s) = s(L_{xx} + L_{yy}). \tag{9}$$

The DoG operator (Lowe, 2004) is an approximation of the Laplacian operator, generalizing to the scale-normalized Laplacian case with scale levels given by $\sigma_{i+1} = k\sigma_i$:

$$\text{DoG}(x, y; s) \approx \frac{(k^2 - 1)}{2} s\nabla^2 L(x, y; s) = \frac{(k^2 - 1)}{2} \nabla^2_{norm} L(x, y; s) \tag{10}$$

We utilize the *scikit-image* implementation of the DoG blob detector, which requires the user to define the $\sigma_{\min}$ and $\sigma_{\max}$ hyperparameters related to the minimum and maximum blob size, respectively. Since σ is given in pixel units we define minimum and maximum radii r of objects of interest in meters and use the following heuristic function to

output $\sigma_{\min}$ and $\sigma_{\max}$ based on the UAV height h measurements and known intrinsic camera parameters (DFOV, image size):

$$\sigma(r,h) = r\frac{\sqrt{I_{\text{width}}^2 + I_{\text{height}}^2}}{2h\tan\frac{\text{DFOV}}{2}} \tag{11}$$

$$\sigma_{\min} = \sigma(r_{min}, h) \tag{12}$$

$$\sigma_{\max} = \sigma(r_{max}, h) \tag{13}$$

Examples of the detection output, based on two samples of experimental data collected in Marseille, France, can be observed in Fig. 6.

4.3.2. Deep-learning-based, underwater debris detection

The underwater detection of debris to further refine the map is realized by the Observation ROV. During this task, objects are detected, localized, and classified as debris, plants, animals, or ROV parts respectively, using a Convolutional Neural Network (CNN) algorithm, selected for its capability to extract hierarchical features from images, which is necessary for the detection and classification of objects in underwater environments (Ilioudi et al., 2022). Representative detector families include two-stage region-proposal networks like Faster R-CNNs, one-stage real-time detectors like the YOLO family, and end-to-end transformer-based detectors like Real-Time Detection Transformer (RT-DETR) and variants (Jian et al., 2024; Redmon et al., 2016; Ren et al., 2017; Zhao et al., 2024). Two-stage approaches can achieve high-performance object detection with pixel-level accuracy. However, the inference time for this type of CNNs is not sufficiently fast to support real-time applications when processing the video input from the ROV's camera. In addition, R-CNN requirements in terms of memory and storage have limitations when they are implemented on robotic hardware, due to the lack of large-scale cooling mechanisms in these applications, which are necessary in order to support operation under such an intensive load. In contrast, one-stage and real-time transformer detectors achieve speed–accuracy trade-offs appropriate for embedded inference (Terven et al., 2023; Carion et al., 2020; Lv et al., 2024; Wang et al., 2024). Given the real-time constraints of the SeaClear system, we adopt a YOLO-family detector, because of its performance on resource-limited platforms and its mature deployment ecosystem, despite the recent advances in RT-DETR models. Our decision is also supported by our data, as in the technical validation of the SeaClear dataset, a model of the YOLO-family achieved higher performance than Faster R-CNN (Đuraš et al., 2024a).

To properly train the YOLO neural network, a large custom dataset is generated by data collected during different data collection trials. The synthesis of a sufficiently large and good-quality dataset is not trivial, considering that it should contain observations taken in varying conditions to ensure the generalization capability of the trained neural network. This also requires accurate ground-truth annotations. Due to the scarcity of publicly available datasets for underwater debris, it was necessary to generate our own dataset (Đuraš et al., 2024b). The dataset consists of 8610 labeled image samples[2] collected using ROVs in five shallow-water locations in Croatia and France. It contains observations of 40 object categories, grouped into debris, animals, plants, and ROV parts. Annotations were created using the LabelMe tool (Russell et al., 2008), which provides polygonal segmentation masks and bounding boxes for each object instance, following the COCO format. Further details about the data collection setup, annotation taxonomy, and validation experiments can be found in Đuraš et al. (2024a). The YOLO architecture (YOLOV6 S, 17.2 M parameters) is trained and tested on this generated dataset. The training was done based on the implementation provided in the official repository of this architecture.[3] The YOLO network was trained for 200 epochs with a batch size of 32 and an input image size of 640 × 640.

The performance metrics of object detection, such as Precision (P), Recall (R), and the F1 score, are necessary for evaluating the effectiveness of the YOLO network (Goutte and Gaussier, 2005). Precision measures the accuracy of the predictions, indicating the proportion of positive detections that were actually correct, while Recall assesses the ability of the neural network to find all relevant instances in the dataset. The F1 score is the harmonic mean of Precision and Recall and it provides a single score that balances both metrics.

In addition to these metrics, the mean Average Precision (mAP) and the Average Precision at various Intersection over Union (IoU) thresholds are important in evaluating the performance of object detection techniques. The IoU criterion measures the overlap between the predicted bounding box and the ground truth, with a higher IoU indicating a more accurate prediction. The Average Precision corresponds to the area under the precision–recall curve, and it provides a summary measure that combines the precision and recall into a single metric. The mAP is the mean of the average precision values across all classes and IoU thresholds, providing a single score to represent the overall precision and recall of the neural network across different object categories (Henderson and Ferrari, 2017).

The results of training the YOLO network are presented in Table 1. The YOLO network achieves a performance equal to 93.5% mAP at 0.5 IoU threshold, while specifically for the trash, the YOLO network can achieve a performance of 92.9% mAP at 0.5 IoU threshold. It should be noted that, as anticipated, the mAP exhibited a decline at higher IoU thresholds. This is consistent with the nature of the mAP@0.5:0.95 metric, which averages the mAP over a range of IoU thresholds from 0.5 to 0.95, thereby affecting the true positive classification rate. The reduction at higher thresholds is also due to the uncertainty in object boundaries in underwater environments, where the visual conditions such as turbidity, light scattering, and partial occlusions introduce small localization deviations that impact IoU. In addition, the implementation of image enhancement methods was also explored, but they did not yield a significant increase in performance, and hence they were not incorporated into the real-time application (Ilioudi et al., 2023).

To improve the accuracy and reliability of the YOLO-based detection module, a strict confidence threshold mechanism is implemented in combination with a temporal validation method. Each detected object in YOLO is associated with a confidence score, ranging from 0 to 1, representing the neural network's certainty that the bounding box accurately includes a specific class of object. We have implemented a high confidence threshold of 0.85 in our approach. This ensures that only detections with a high likelihood of correctness are considered, thereby significantly reducing the probability of false positives. In addition to the confidence threshold, our approach incorporates a temporal validation strategy to address the challenge of repetitive or erroneous detections in successive frames. This approach involves a time-based criterion for detection validity, where a detection must persist for a certain duration, between 2 and 3 s, to be considered valid. This duration is chosen based on empirical observations, ensuring a balance between detection responsiveness and accuracy. In this way, we filter out transient detections that are likely to be false positives.

The image frames from the video stream of the ROV's camera are fed as input to the YOLO neural network, which then outputs, for each detected object, the class label, the confidence score, and the coordinates of the bounding box around the object (Fig. 7).

[2] This size is comparable with the widely used publicly available dataset TrashCan that contains 7212 images.

[3] https://github.com/meituan/YOLOv6

Table 1
Evaluation of YOLO performance.

Class	Labeled images	Labels	P@.5iou	R@.5iou	F1@.5iou	mAP@.5	mAP@.5:.95
all	1722	6321	0.925	0.89	0.907	0.935	0.711
rov	172	204	0.975	0.94	0.957	0.974	0.876
plant	84	92	0.932	0.89	0.91	0.92	0.649
trash	1542	3333	0.932	0.89	0.91	0.929	0.725
animal	685	2692	0.915	0.85	0.881	0.918	0.594

Fig. 7. Example result of the YOLO object detection algorithm.

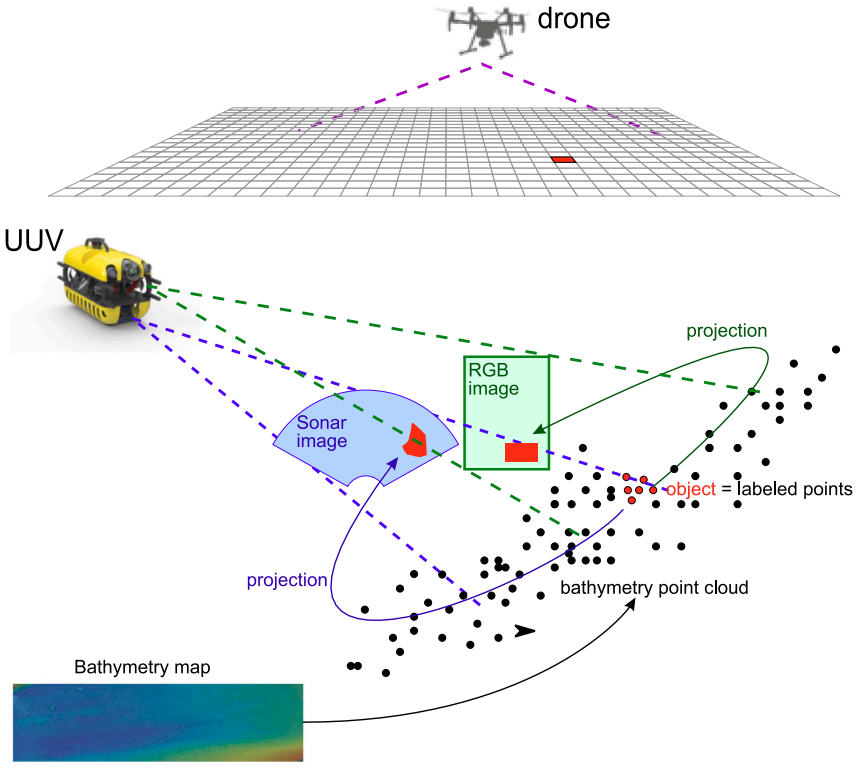


Fig. 8. Mapping framework.

4.3.3. Litter mapping

Once the objects have been detected, they must be placed on a map for later pickup by the Collection ROV. The SeaClear pipeline for doing this is shown in Fig. 8. Litter detected in the UAV camera stream is registered in a 2D occupancy grid. The ROV-based map is maintained as a point cloud, initialized from the bathymetry scan performed by the SeaCat USV. Litter detected in the Observation ROV sonar and camera streams is registered by labeling points in this point-cloud-based, reference map.

Methodologically, we perform a forward projection for detections by both the camera and the sonar on the Observation ROV, using a separate projection model for each sensor. The projection methods for the camera and sonar are illustrated in Fig. 9, left and right respectively. A pinhole camera model is used, for which the parameters are obtained

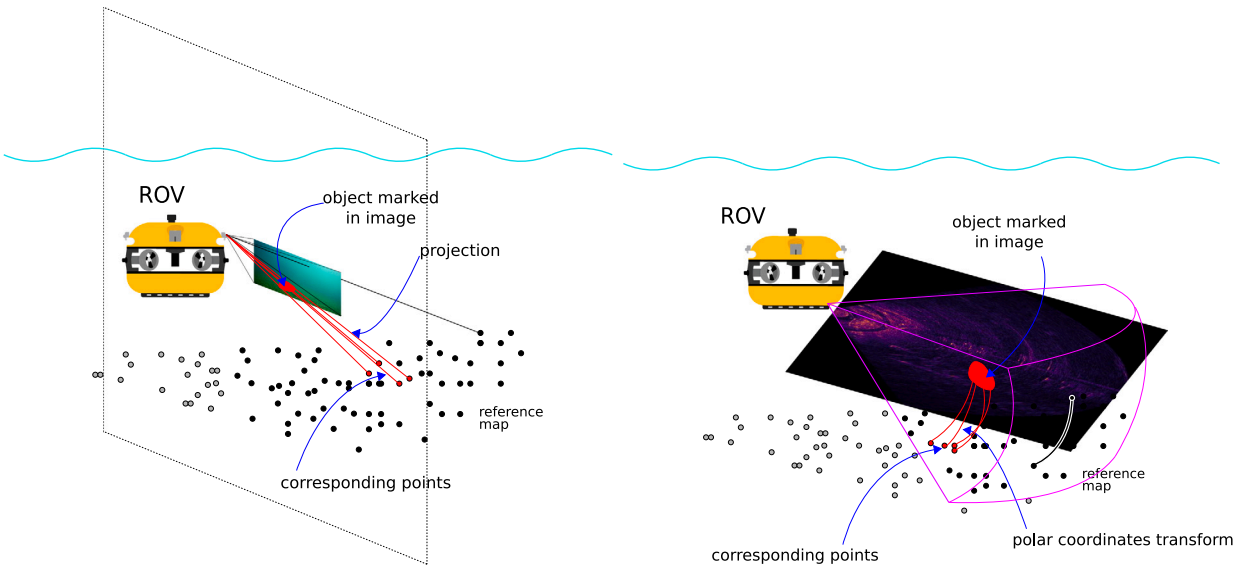


Fig. 9. Left: Mapping from a marked camera image. The pre-selected points are shown in black, and the projection from the point cloud into the image is shown by lines. The red object in the camera image corresponds via the red lines to the object points in the point cloud, also shown in red. **Right:** Mapping from a marked sonar image. The symbolism is similar to the camera mapping. One difference is that here the black points are pre-selected using the more complicated shape of the sonar field of view, shown in a red outline.

by performing a calibration in seawater. The image coordinates p are obtained from the homogeneous world coordinates of the point P_w using:

$$p = KTP_w \tag{14}$$

where T is the transform between the camera and the world coordinate frames, and K is the intrinsic matrix. For the sonar, the model assumes that a point in the volume visible by the sensor produces an image in the zero-elevation image plane at the range and azimuth of the point. The image frame produced by the sonar, in this case, is a finite, scaled, and discretized subset of the zero elevation image plane. Thus, the projected point p in the image plane can be found using the polar coordinates of this point, consisting of range R and azimuth angle θ:

$$p = R\begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix} \tag{15}$$

For both projection methods, we compute where a pre-selected part of the bathymetry point cloud would fall in the image, given the ROV pose, and subsequently, points that intersect the mask of the litter detected in the image are labeled. Point pre-selection is performed to reduce computational requirements. For the camera, the half-plane in front of the ROV is pre-selected, while for the sonar we preselect points found within the bounding azimuth and elevation angles reported by the sonar itself. These points define a cone-like shape, illustrated in magenta in Fig. 9, right. A mechanism to prevent duplicate detections from subsequent image frames is also implemented: when a litter item is detected closer than a threshold to an existing item, instead of adding a new item, a first-order filter is applied to move the existing item towards the new detection. The threshold is tuned based on the precision of the pose estimate.

A final task in mapping is to design the ROV trajectory. A simple lawnmower is applied in the field experiments, but in simulations and proof-of-concept experiments in the lab we are also investigating methods to find more efficient paths, based either on reinforcement learning (Rosynski and Buşoniu, 2022) or model-based planning (Yousuf et al., 2022).

4.4. Litter collection

Once objects have been cataloged in the reference map, they are ready for collection by the Collection ROV.

4.4.1. Path planning

To reach a litter location safely, various path planning frameworks have been considered (Weston et al., 2024; Gammell et al., 2014). A collision-free path is planned using informed-RRT* (Gammell et al., 2014), aided by a local Nonlinear Model Predictive Control (NMPC) planner with cascaded (position and velocity) control (see Fig. 10).

The planner works directly in the Robot Operating System (ROS) and it is based on the octomap[4] that is generated during the bathymetry scan and mapping process. An octomap is based on octrees, an efficient hierarchical data structure for spatial subdivision, and uses probabilistic occupancy estimation for representing volumetric 3D environment models (Hornung et al., 2013/04/01). An example from our simulation can be found in Fig. 15 in Section 5.1.1.

4.4.2. Vision-based collection strategy

For collecting litter objects with the Collection ROV, we define a strategy with three steps after reaching the estimated litter location: *searching*, *visual-servo tracking* and *collection*, as illustrated in Fig. 11.

Searching. It should be noted that both the pose estimation during the litter mapping and the pose estimation of the Collection ROV show estimation errors (up to 1.3 m RMSE, see Section 5.2.3), which cumulatively can lead to the Collection ROV missing the true litter location and having the object outside the field of view of the downward-facing camera. To reacquire litter location, a spiral search pattern is tracked in the xy-plane with fixed height and constant velocity.

Visual-servo tracking. Upon detecting a litter object, we employ a vision-based Nonlinear Model Predictive Controller (NMPC) based on (Heshmati-Alamdari et al., 2018) for collection with the Collection ROV. Using the frame definitions shown in Fig. 11, the goal is to coincide the body-fixed frames of the gripper C_G with the object of interest C_O, tracking the object visually in the camera frame C_C (with a static transformation to C_G). This robust sensor-based control approach ensures accurate object grasping and (since it works in a local camera reference frame) does not depend on the pose accuracy in world coordinates. However, it introduces additional challenges to maintain the object within the camera's field of view (visibility

[4] https://octomap.github.io

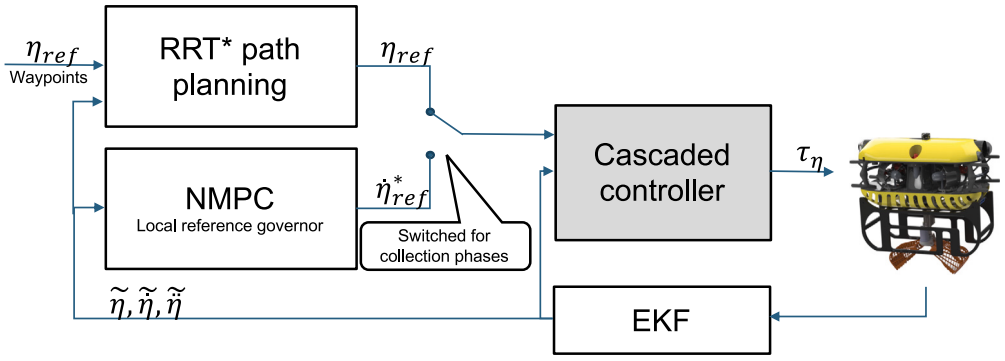


Fig. 10. Control structure of the Collection ROV with switching reference inputs during path tracking and the collection sequence.

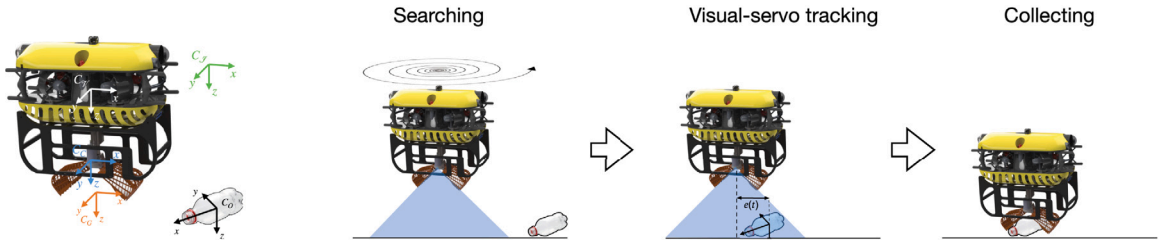


Fig. 11. (Left) Depiction of the inertial frame (C_I), the vehicle-frame (C_V), the body-fixed frames of the camera (C_C), the gripper (C_G), and the frame of the object of interest (C_o) (Right) Grasping phases: If no object is directly found at the map coordinates, a spiral search pattern is followed until a litter object is detected in the gripper-camera image. With the object in sight, visual-servo tracking centers the object in the gripper frame during the decent. Once the object is in reach it is collected for transportation to the basket..

constraint), as well as constraining the velocity of the ROV. A more detailed description is provided in the upcoming Section 4.4.3.

Collection. Once the object is centered in the camera frame and the Collection ROV descends, grasping is initiated based on an altitude threshold determined by the DVL with respect to the object and seafloor, such that the object is within reach of the gripper shovels and scraping on the seafloor is minimal (but desired to envelope objects).

Remark. Should the persistent object tracking be lost at any point during the grasping procedure, be it either by intermittent detection due to unfavorable lighting or visibility conditions for the detection algorithm, occlusion, or leaving the field of view, the last detected position is tracked via means of a complementary filter (Marantos et al., 2016). This robustifies against tracking losses in the range of a few seconds. A special case of tracking loss can happen close to the target object, with persistent visibility[5] being lost for the remainder of the approach. The algorithm then switches to an open-loop grasping phase based on the last recorded pose of the litter target utilizing the complementary filter.

4.4.3. Nonlinear Model Predictive Control

Here we provide details to the vision-NMPC controller to guide the robot to the collection position while respecting operational limitations defined below. Consider an underwater vehicle with dynamics as described in Fossen (1999), equipped with a complete Attitude and Heading Reference System (AHRS) scheme, a gripper, a camera system and an online object detection algorithm. Given the real-time pose of the camera relative to the object frame provided by the vision system, ${}^{\mathcal{O}}\boldsymbol{\eta}_C(t)$, and the fixed, known pose of the camera frame C_C with respect to the gripper frame C_G, the pose of the gripper frame relative to the object frame ${}^{\mathcal{O}}\boldsymbol{\eta}_G(t) = \left[{}^{\mathcal{O}}\boldsymbol{p}_G^T(t)\ {}^{\mathcal{O}}\boldsymbol{\phi}_G^T(t)\right]^T$ with relative position ${}^{\mathcal{O}}\boldsymbol{p}_G \in \mathbb{R}^3$ and relative orientation ${}^{\mathcal{O}}\boldsymbol{\phi}_G \in \mathbb{R}^3$ can be readily computed. Considering the collection point as the desired pose of the gripper with respect to the object frame i.e., ${}^{\mathcal{O}}\boldsymbol{\eta}_G^d$, we could define the error between the gripper and collection point as:

$$\boldsymbol{e}(t) = {}^{\mathcal{O}}\boldsymbol{\eta}_G(t) - {}^{\mathcal{O}}\boldsymbol{\eta}_G^d(t) \tag{16}$$

In sampled data NMPC, a finite-horizon optimal control problem is solved at discrete sampling time instants t_j based on the current measurements $\boldsymbol{e}(t_j)$, with prediction horizon T_P and a constant sampling time $0 < h < T_P$, such that $t_{j+1} = t_j + h,\ \forall j \geq 0$. Thus, the control signal applied in between the sampling instants is given by the solution of the following finite-horizon optimal control problem (see (Fossen, 1994)):

$$\min_{\hat{\boldsymbol{\nu}}_{\mathcal{V}}(\cdot)} J(\boldsymbol{e}(t_j), \hat{\boldsymbol{\nu}}_{\mathcal{V}}(\cdot)) = \tag{17a}$$

$$\min_{\hat{\boldsymbol{\nu}}_{\mathcal{V}}(\cdot)} \left\{ \int_{t_j}^{t_j+T_p} F\left(\hat{\boldsymbol{e}}(\tau), \hat{\boldsymbol{\nu}}_{\mathcal{V}}(\tau)\right) d\tau + E\left(\hat{\boldsymbol{e}}(t_j + T_P)\right) \right\}$$

subject to :

$$\text{System dynamics} : \dot{\hat{\boldsymbol{e}}} = f(\hat{\boldsymbol{e}}(t), \hat{\boldsymbol{\nu}}_{\mathcal{V}}(t)), \tag{17b}$$

$$\hat{\boldsymbol{e}}(t_j) = \boldsymbol{e}(t_j), \qquad \hat{\boldsymbol{\nu}}_{\mathcal{V}}(t_j) = \boldsymbol{\nu}_{\mathcal{V}}(t_j), \tag{17c}$$

$$\hat{\boldsymbol{\nu}}_{\mathcal{V}}(t) \in V_{\mathcal{V}}, \qquad {}^{\mathcal{O}}\boldsymbol{\eta}_C(t) \in H_C \tag{17d}$$

$$\hat{\boldsymbol{e}}(t_j + T_P) \in \mathscr{E}_f \tag{17e}$$

where $\mathscr{E}_f$ is a terminal region around the desired configuration and F and E are the running and terminal cost functions respectively which are both of quadratic form i.e., $F(\cdot) = \hat{\boldsymbol{e}}^\top \boldsymbol{Q} \hat{\boldsymbol{e}} + \hat{\boldsymbol{\nu}}_{\mathcal{V}}^\top \boldsymbol{R} \hat{\boldsymbol{\nu}}_{\mathcal{V}}$ and $E(\cdot) = \hat{\boldsymbol{e}}^\top \boldsymbol{P} \hat{\boldsymbol{e}}$, respectively, with $\boldsymbol{P}$, $\boldsymbol{Q}$, and $\boldsymbol{R}$ being positive definite matrices to be appropriately tuned (Heshmati-Alamdari et al., 2018). $\boldsymbol{\nu}_{\mathcal{V}} = \left[\boldsymbol{\nu}_{\mathcal{V}_1}^T\ \boldsymbol{\nu}_{\mathcal{V}_2}^T\right]^T \in \mathbb{R}^6$ is the velocity vector of the vehicle expressed in body fixed frame $\mathcal{V}$ and includes the linear (i.e., $\boldsymbol{\nu}_{\mathcal{V}_1} = \left[u_{\mathcal{V}}\ v_{\mathcal{V}}\ w_{\mathcal{V}}\right]^T$)

[5] determined by the object dimensions and the field of view of the camera: for the specific field of view of the camera used in the SeaClear system (86 deg) and debris objects up to a size of standard 1.5 liter bottles, this is typically up to 5cm outside the gripper reach, with smaller objects being fully tracked until grasped.

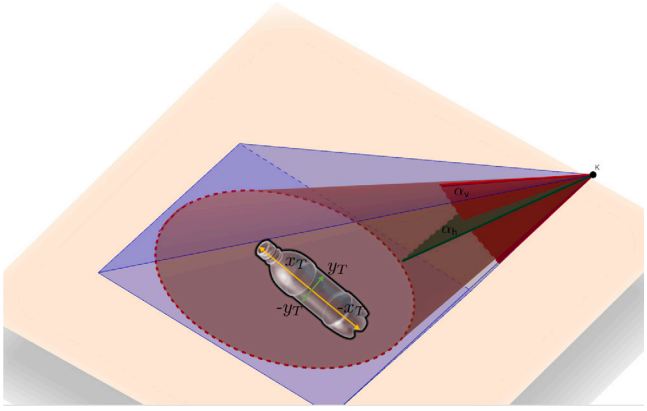


Fig. 12. Modeling of the camera visibility limitations for an object with dimensions y_T and z_T. The camera's horizontal and vertical angles of view are denoted by α_h and α_v respectively. The inner cone describes a set of soft constraints to keep the object within the field of view, whereas the pyramid forms the set of hard constraints of the camera's field of view.

and angular (i.e., $\boldsymbol{\nu}_{\mathcal{V}_2} = \left[p_{\mathcal{V}}\ q_{\mathcal{V}}\ r_{\mathcal{V}}\right]^T$) velocity vectors. Please note that $\hat{(\cdot)}$ denotes filtered estimates from an extended Kalman filter.

Field of view constraints H_C. The underwater robot should be able to always keep the object of interest within the camera field of view during the visual-servo tracking operation. Given the dimensions of the object[6] and the camera's horizontal and vertical angles of view α_h and α_v as shown in Fig. 12, a set of inequalities can be defined that must be satisfied for this. These inequalities are the 3D extended version of the field of view constraints presented in our previous work (Heshmati-Alamdari et al., 2014). A detailed description of the inequalities is provided in A.

Velocity constraints $V_{\mathcal{V}}$. In order to allow the vision tracking system to identify the object more frequently (Rozumnyi et al., 2021; Gao and Spratling, 2022), it is required to impose a prescribed smoothness to the system motion. In particular, the vehicle is required to move relatively slowly with upper bound velocity vector $\bar{\boldsymbol{\nu}}_{\mathcal{V}_1} = [\bar{u}_{\mathcal{V}},\ \bar{v}_{\mathcal{V}},\ \bar{w}_{\mathcal{V}}]^\top \in V_{\mathcal{V}} \subset \mathbb{R}^3$, where $\bar{(\cdot)}$ denotes the corresponding upper bounds for each coefficient and $V_{\mathcal{V}}$ is the constraint set given by:

$$V_{\mathcal{V}} := \left\{\boldsymbol{\nu}_{\mathcal{V}} \in \mathbb{R}^3 : |u_{\mathcal{V}}| \leq \bar{u}_{\mathcal{V}}, |v_{\mathcal{V}}| \leq \bar{v}_{\mathcal{V}}, |w_{\mathcal{V}}| \leq \bar{w}_{\mathcal{V}}\right\} \tag{18}$$

4.5. Coordination between UAV and USV during operation

Throughout the underwater mapping and collection phase, the UAV is tethered to the SeaCat USV, hovering above the survey area in order to provide position feedback for the ROVs and to enhance situational awareness. During this operation, the UAV needs to maintain the position with respect to the SeaCat USV, due to the kinematic constraint imposed by the tether. This control task is split between the UAV and the LARS.

The role of the UAV is to control the relative position with respect to the SeaCat USV. To estimate the relative position, an extended Kalman filter is implemented based on continuous RTK measurements obtained from the two vehicles. Moreover, when in sight, the estimator is provided with the intermittent relative position information obtained from the ArUco markers located on the landing platform. This additional measurement is necessary to obtain centimeter-level landing accuracy imposed by exceptionally small landing platform area (1 m^2). Given the relative position estimate, a proportional controller is employed for set-point tracking, with its output feeding into the UAV's proprietary velocity controller.

To prevent potential entanglements and excessive pulling, especially during the take-off and landing phase, the tether needs to be continuously winched in and out to match the UAV motion. This is achieved by controlling the UAV LARS motor velocity. The deployed cable length is estimated based on the LARS motor encoder data. Then, using the *L2* norm of the estimated UAV position as the length reference signal, a Proportional-Integral-Derivative control loop runs on the LARS onboard computer and outputs the motor velocity reference. This velocity reference is subsequently passed to the integrated LARS motor driver. The overall tethered UAV control scheme is shown in Fig. 13.

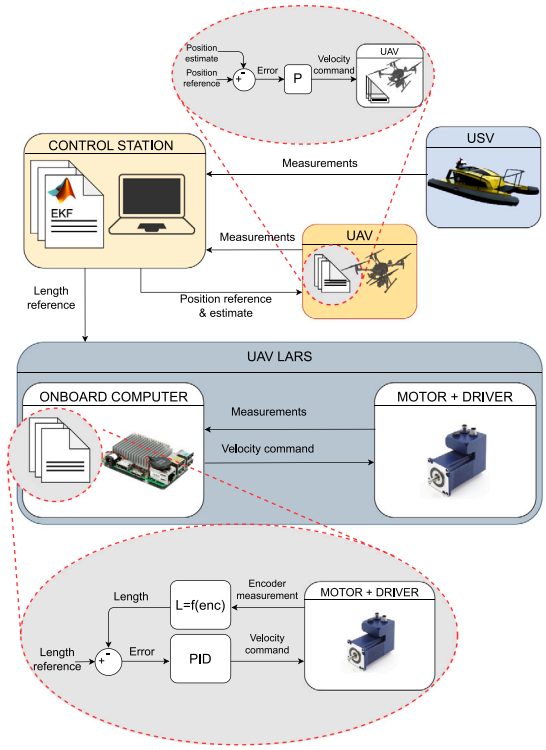


Fig. 13. Tethered UAV control scheme overview.

4.6. Software architecture for system integration

The software architecture of the SeaClear system comprises two main components: the Shore Operation Center (SOC) and the Robotic System (Fig. 14). The SOC contains a collection of server-sided applications that host a web-based application accessible to registered users. Because it provides services that facilitate users to monitor and interact with the Robotic System, the web-based application is also called the SeaClear Service Layer. The SOC also serves as the access point for professional human intervention and provides high-level commands to the Robotic System. It enables direct commanding of the Robotic System and processing of user requests, by means of planning a mission, setting a target position, or enabling and disabling the controllers. It ensures a point-to-point communication with the Robotic System and, at the same time, it exchanges mission information with the web-based application. The SeaCat USV, the two ROVs, and the UAV are all connected through a ROS framework. The Robotic System is connected to the SOC via wireless communication, predominantly using 2.4 GHz WiFi. Sensory data is exchanged directly through native ROS applications, or, in some cases, through ROS-bridge TCP or websockets.

[6] Real-time bounding-box along the horizontal and vertical axes coming from the algorithm described in Section 4.3.2

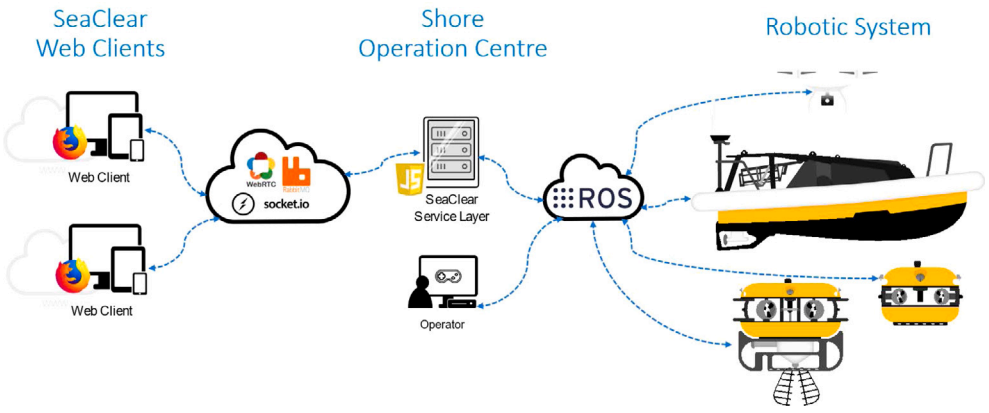


Fig. 14. Overview of the SeaClear software architecture.

Visual perception is enhanced through 1080 p video streams that are directly fed from the onboard cameras through an RTSP[7] protocol.

A digital twin of the Robotic System is connected within the aforementioned architecture similar to the real system, enabling rapid development and offering a cost-effective testing environment for control and navigation of the SeaClear Robotic System. Multiple SeaClear systems, real or digital-counterparts, could be integrated within the same SeaClear Service Layer, provided that the correct transport layer settings are made within each LAN where the fleet of robots operates.

5. Experimental results and analysis

5.1. Simulation results

To enable quick initial testing during the design phase and remote development without a real marine environment, we have used robot simulation tools such as Gazebo. This section presents the simulation results that demonstrate the performance of the SeaClear system.

5.1.1. Simulation model

We developed our own simulation environment based on `UUVsimulator` (Manhães et al., 2016)[8] for the underwater part, `Virtual RobotX` (VRX) (Bingham et al., 2019)[9] for the water-level part, and `RotorS` (Furrer et al., 2016)[10] for the aerial domain. These elements are extensions of Gazebo, a software tool for simulating physical systems such as robots, vehicles, and environments. Gazebo supports realistic sensors, kinematics, and environments via various plugins. We chose Gazebo because of its high modularity through *plug-ins* and its strong connection to ROS, which we use for our control pipeline. This allowed us to test and prototype the software pipeline without the expense of real-world testing, leaving the fine-tuning for real-life marine testing occasions. The creation of digital twins of both the robot systems and the test sites helped to lower the hurdles for transferring the algorithms to the real system and enabled the conceptual functionality test of the full system in an environmental setting mimicking the demo sites. An illustration of such a functionality test for the collection pipeline is shown in Fig. 19.

Test site digital twins were created via a fusion of bathymetry data (sonar scans) for the seabed model, and Google Maps 3D data for the surface models. For litter mapping and collection tasks, the digital twins can be randomly populated with litter models. The resulting overall model is imported as a Collada (.dae) file into the Gazebo simulation environment and used for visualization as well as a collision mesh for objects and sensors.

Visualizations of the terrain for the Petroleumhafen (petrol harbor) in Hamburg, Germany, can be seen in Fig. 16(a), and the combined render with a simulated water volume in Fig. 16(b).

The robot digital twins are created via a fusion of CAD data and kinematics of the robot and an estimate of its drag using linear and quadratic approximations. In order to capture at least the dominant dynamics of the real underwater vehicles, hydrodynamic effects are approximated from tank experiments, in which the pose of the underwater vehicle is tracked from an overhead camera for various excitations of the vehicle.[11] Since the cable has a significant influence on ROV dynamics as well, a beam-based tether dynamics model and simulation is successfully implemented for a large number of links. It uses a finite element model for modeling the movement of the umbilical in 3D. Further details are presented in Bulić et al. (2022).

Integrating a robot's geometry and actuator locations into the SeaClear simulator is very similar to an existing framework in Gazebo ROS. The description of the robot utilizes the Unified Robot Description Format (URDF) with Xacro.

Sensors are added onto a simulated vehicle by using Gazebo plugins, mostly readily available from `UUVsimulator`. Figs. 17(a) to 18(b) show visualizations of the digital twins.

To showcase the simulation, Fig. 19 displays a simulated sequence of the automated litter collection with the Collection ROV. For the sake of clarity in presentation, after deployment the simulation time between 10 s and 38 s has been cut from the plot to allow the system to settle to the initial conditions before receiving its first command. The robot follows a series of waypoints until it receives a litter location from mapping at (−1.3 m, 0.2 m). Note that this does not coincide with the true litter location at (−2.1 m, 0.9 m), so that the object is outside the field of view of the camera. Therefore, the search strategy for reacquisition of the litter item is followed until the downward-facing camera detects the litter object for at least 2 s. As long as the object is detected, the visual-servo tracking then ensures movement of the Collection ROV to bring the object to the image center and align with the gripper. At the required minimum altitude over the object, determined by the DVL, the gripper is activated to grasp the litter object. Fig. 20 shows the RMSE of the waypoint and object tracking controller of the Collection ROV. Spikes in the RMSE plot appear due to asynchronous switching of both the reference coordinate system (global or object) and the desired position in the state machine simulation. During tracking in the global frame, the RMSE never converges as new path planning goals are constantly published ahead.

[7] https://datatracker.ietf.org/doc/html/rfc2326
[8] https://github.com/uuvsimulator/
[9] https://github.com/osrf/vrx
[10] https://github.com/ethz-asl/rotors_simulator

[11] The dataset created and used for the system identification is publicly available: https://doi.org/10.4121/012fc1f6-4363-4521-8fc1-81e24f39b821.v1

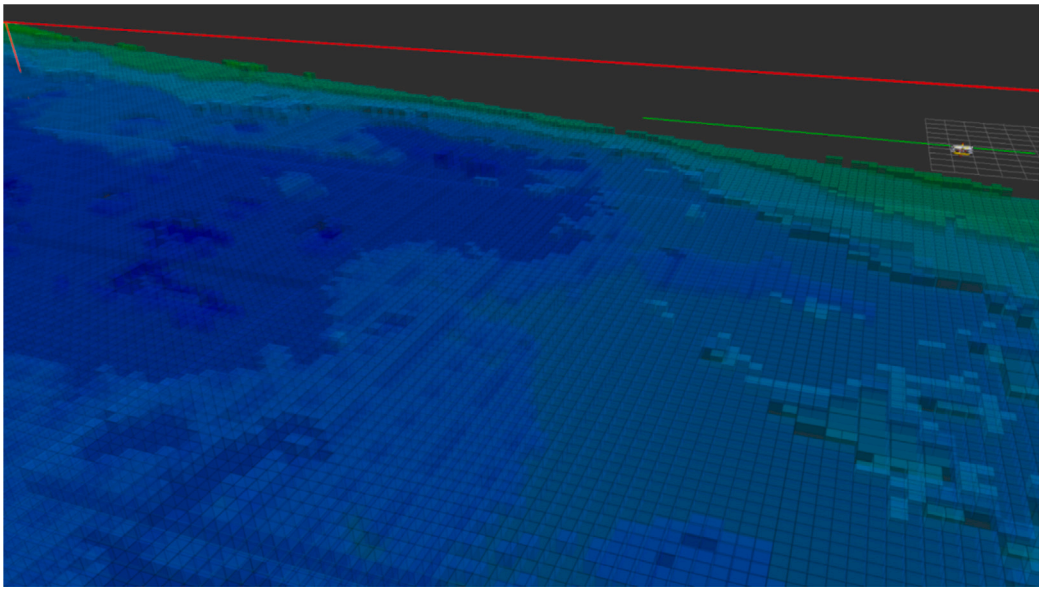
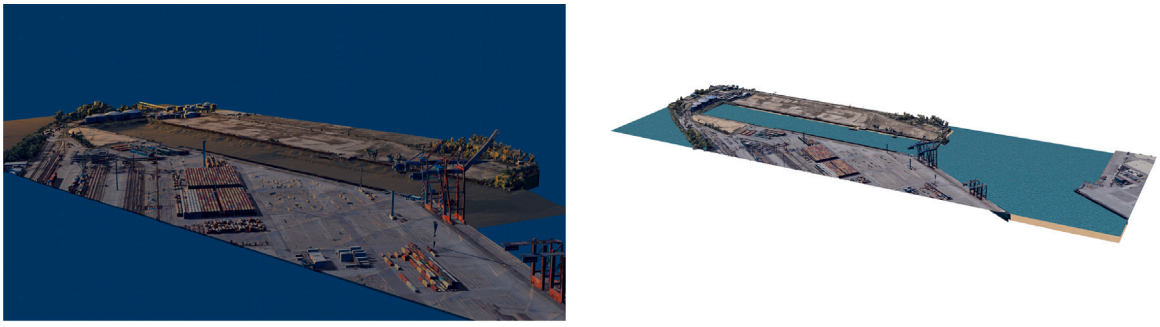


Fig. 15. Red: The planner bounds. Green: A downsampled path planned by the RRT* algorithm in the simulated octomap created from real bathymetry data.



(a) 3D terrain render (b) Gazebo water-bound render

Fig. 16. Hamburg port as a digital twin.

(a) Collection (Tortuga) ROV. (b) Observation (mini Tortuga) ROV.

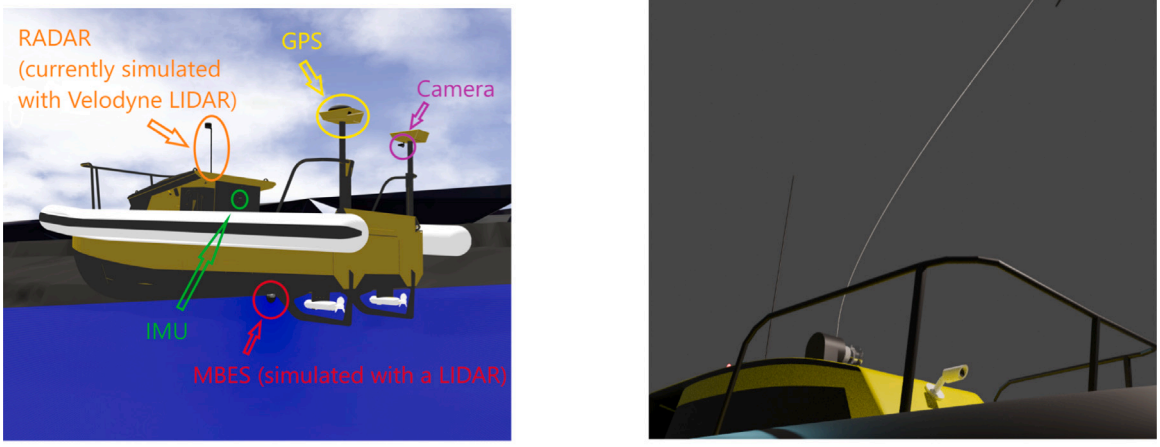
Fig. 17. ROV digital twins in Gazebo.

5.2. Field experiments

Multiple field experiments were carried out in real-world settings in Hamburg, Germany, and in Dubrovnik, Croatia (specifically Lokrum island and Bistrina Bay) to confirm the effectiveness and applicability of the system that was designed. In this section, we present some results concerning the ROV pose estimation, control, mapping, object detection, and collection obtained during these field experiments.

5.2.1. Field experiments infrastructure

The SeaClear operations have been planned taking in consideration a minimal infrastructure for deploying the robotic system. Specifically, for the Hamburg port environment a 20 ft container placed on a floating pontoon has been used as land station, with a direct sight towards the harbor basin. The wireless equipment has been mounted on the roof of the container, ensuring at least 5 m height above sea level. In the test site around Lokrum island, the equipment was placed on the lower



(a) SeaCat USV and sensor integration.

(b) Simulation of the Tethered DJI Matrice drone.

Fig. 18. Rest digital twins in Gazebo.

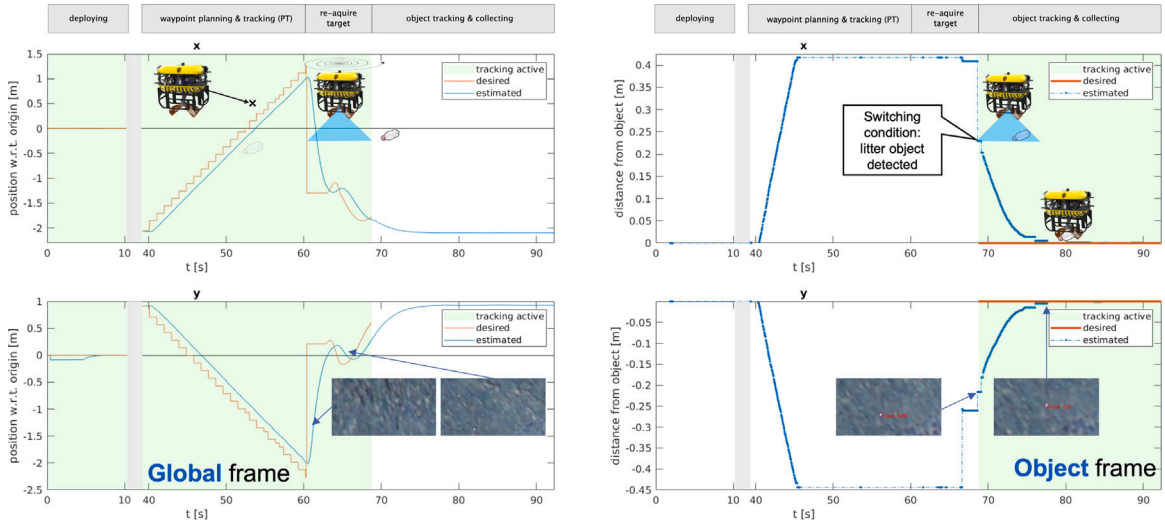


Fig. 19. Simulation of the full automated collection sequence based on live object detection. The green shaded areas highlight the active tracking in global and object coordinate frames, respectively. Upper plots show the x-coordinate, lower plots the y–coordinate in the respective frame. Both plots show convergence of tracking in the object frame towards the center of the camera frame for picking up the object with the camera-aligned gripper.

deck of the carrier ship, at a height of about 4 m above sea level. The antenna used was a 5G Hz, 19/20 dbi, 120 deg sector antenna, which is guaranteed by the manufacturer for a few kilometers, with a reasonable drop of bandwidth. The electromagnetic waves are susceptible to various environmental conditions, including temperature, humidity, precipitation and are impacted by the reflections from the water surface. In addition, the wireless communication link was limited to a single Point-To-Point connection from the USV to shore, having a frequency and channels tuned by the manufacturer to be hidden to the other wireless network clients and therefore minimizing the noise on the wireless channel. On land, the sector antenna was connected to the operational team via cable. During demonstrations, the number of clients to high-bandwidth data, such as video streams was limited to the operator monitoring the specific process. Moreover, high-payload data, such as sonar images, uncompressed images, or point cloud data have been omitted from the operational log (in the form of rosbags) and saved on the corresponding companion computers. A summary of the bandwidth allocation and the acceptable latency is provided in Table 2:

5.2.2. Litter detection

Fig. 21 presents a series of detections captured during the field test in Bistrina, Dubrovnik, Croatia. The YOLO algorithm demonstrated an 87% accuracy rate in litter detection. This validates the system's capability to effectively identify various types of marine debris. Specifically, the system recognized both larger objects and smaller items, such as rubber tires, plastic bottles, and entangled ropes, which are commonly found in underwater environments. In addition to its litter detection capabilities, the algorithm also demonstrated a high level of accuracy in recognizing marine life. This functionality to differentiate between litter and marine fauna highlights the algorithm's performance at the critical feature of preventing ecological harm during the cleanup process.

However, it is important to acknowledge the algorithm's limitations in adapting to varied environmental conditions. The tests near Lokrum Island, conducted in shallow waters approximately 15 m deep, revealed that changing lighting conditions significantly affected the algorithm's performance. This limitation was particularly evident when the environmental conditions deviated from those of the training dataset, indicating a need for further development in this area.

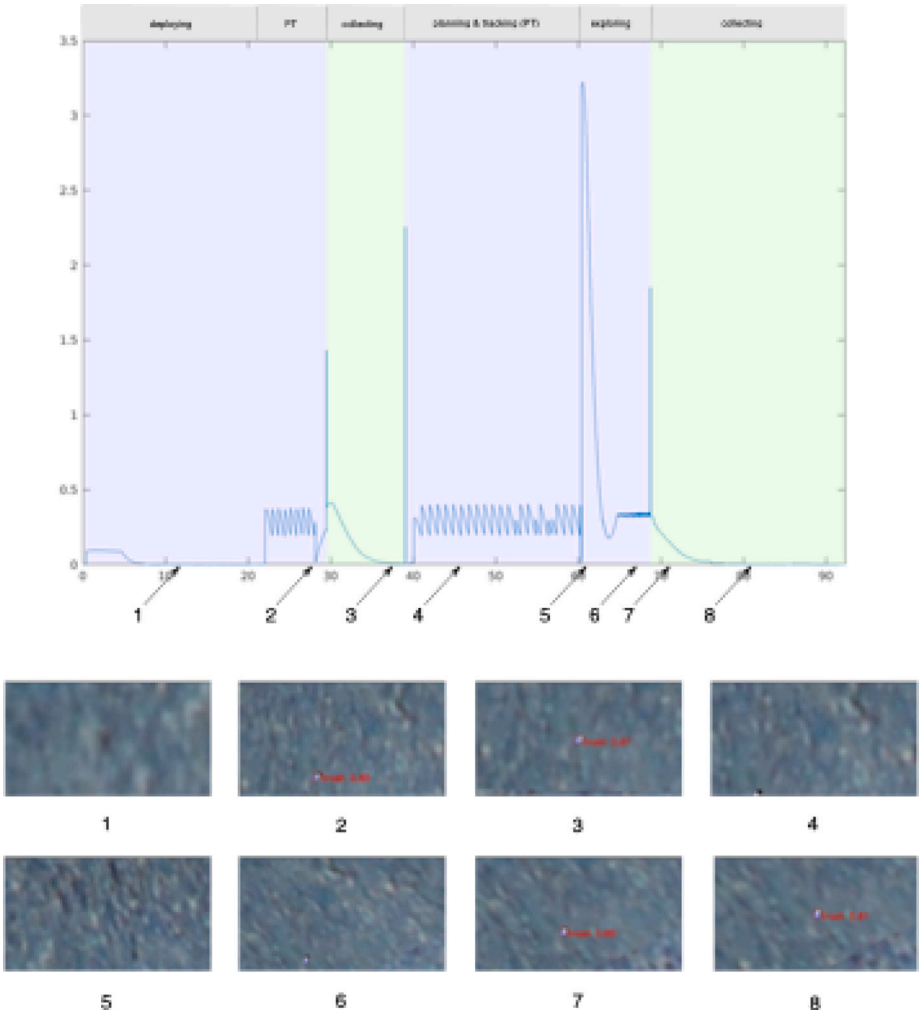


Fig. 20. Plot of the RMSE for trajectory tracking and visual-servo object centering during the collection sequence. The numbered frames represent snapshots of the downward-facing camera on the Collection ROV, indicating if a litter object is visible and detected. During successful detection, the visual servo tracking in the object frame is active (highlighted in green), converging to the true object position.

Table 2
Wireless communication system bandwidth allocation and acceptable latencies (tested within LAN).

Data Type	Frequency	Bandwidth Allocation [Mbit]	Latency [ms]
Text	1 Hz–1 kHz	0.8	40–60
PointCloud	300 000 pps	20	1000
RTSP HD VideoStream (Operator)	30 fps	50	150
Web HD VideoStream (Web client)	30 fps	50	3000

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

Fig. 21. Object detection results during the field experiment.

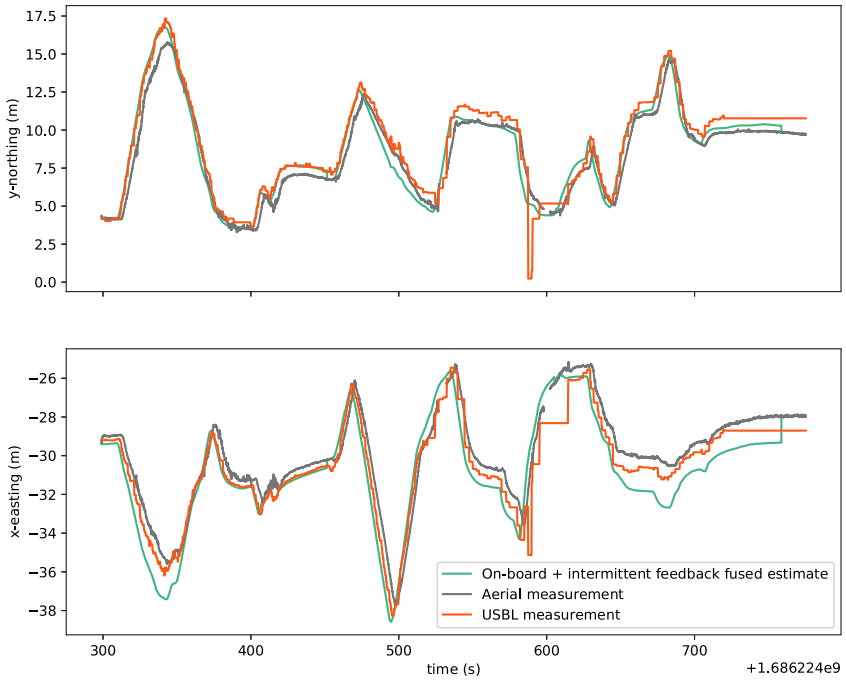


Fig. 22. Pose estimation result: evolution over time of the planar positions X and Y. The horizontal axis depicts Unix epoch time in seconds.

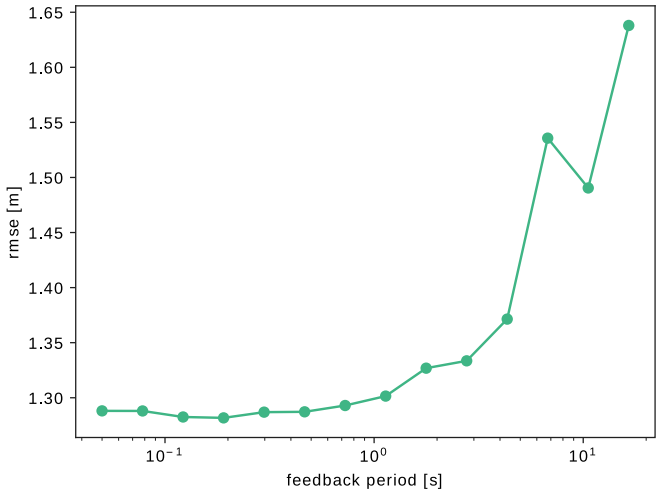


Fig. 23. RMSE between pose estimate and SBL obtained for a range of UAV feedback period choices.

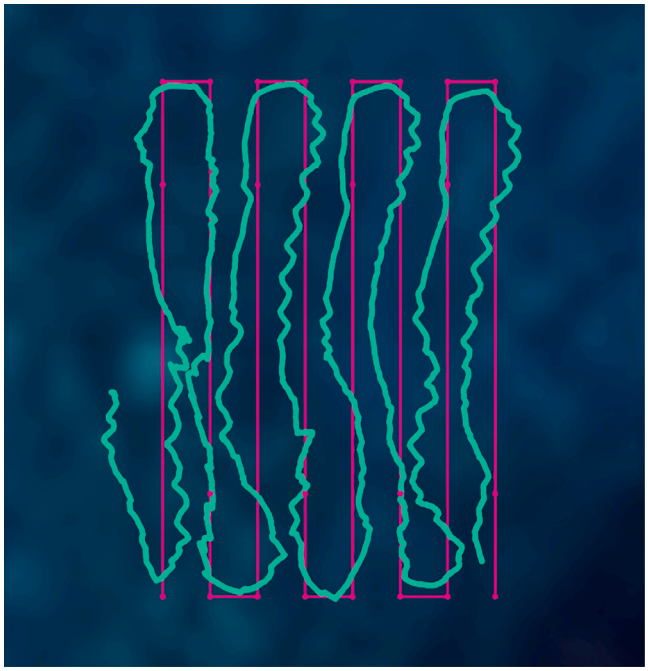


Fig. 24. Planned waypoints (magenta disks) connected by lines, and actual trajectory (teal) of the ROV.

5.2.3. ROV pose estimation

Fig. 22 shows the estimated planar position of the ROV in an experiment run in Hamburg, Germany. The UAV-based position is shown in black, and this is fed back into the ROV pose estimator with a period of 153 s, where it is fused with onboard sensors, and the resulting estimate is shown in green in the figure. To investigate the accuracy of this estimate, we compare it against SBL acoustic positioning, shown in orange, and the RMSE obtained is 1.3 m. It should be noted that to avoid missing litter objects despite this relatively large error, the Collection ROV performs a local, spiral search pattern to reacquire visual contact with the object once reaching its approximate location.

Fig. 23 shows the influence of the period with which the UAV-based position is read on the RMSE compared to the SBL reference. As expected, errors generally increase with larger periods.

5.2.4. ROV control

A series of waypoints describing a lawnmower trajectory was defined for the Observation ROV to follow so as to perform mapping of a litter hotspot area detected in Dubrovnik, Croatia. Fig. 24 shows these waypoints as small disks and the real trajectory of the ROV. Note that the waypoints are also connected by lines to clarify things visually, but in reality the ROV runs a setpoint regulator that aims to reach the next waypoint, rather than trajectory tracking. The curved trajectories followed in-between waypoints are due to the fact that the robot is strifing laterally, and unmodeled tether dynamics and

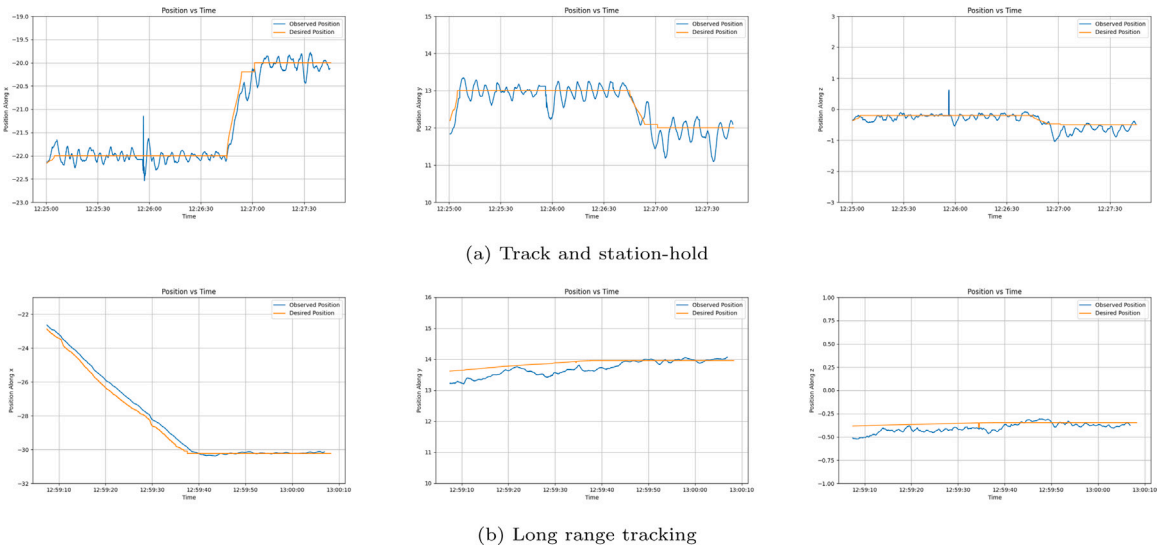


Fig. 25. Tracking performance of the Collection ROV from Hamburg port demonstrations.

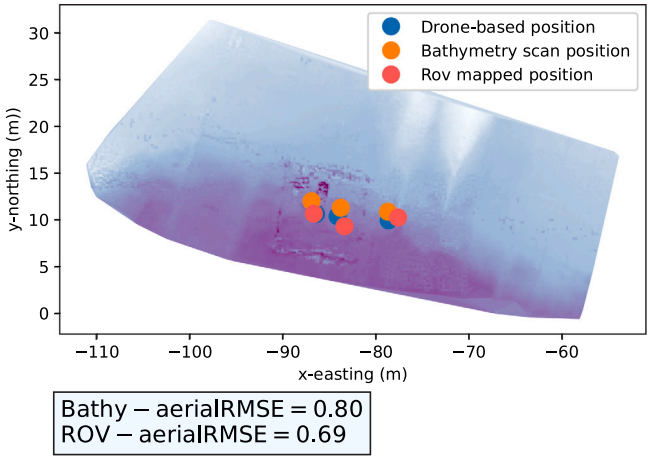


Fig. 26. Mapped litter positions.

dynamical couplings are affecting the control; nevertheless they are useful to cover more area with the sensors.

During SeaClear demonstrations at the Hamburg test site, data was collected from position tracking during experiments with the Collection ROV. Raw data was collected from sensory, state estimation, and control input data to track errors in both position and velocity. Figs. 25(a) and 25(b) show the desired positional references and observed positions via odometry from an extended Kalman filter, respectively. The observed error consistently remains below 0.5 m and the RMSE values for positions consistently stay below 0.3 m.

5.2.5. Litter mapping

Fig. 26 shows the mapping results in an experiment run in the same location as the pose estimation experiment above. The bathymetry point cloud is plotted, on top of which we show the positions of three identified litter items. Three versions of each position are shown: as manually identified on the bathymetry scan, as detected in the UAV image, and finally, as automatically mapped in the sonar image through the ROV mapping pipeline described earlier. The UAV detection serves as a means to validate the others; note that in general, finding underwater objects on the UAV image will not be possible. All three sets of positions are in good agreement with each other, with the ROV detections being slightly closer to the UAV ones.

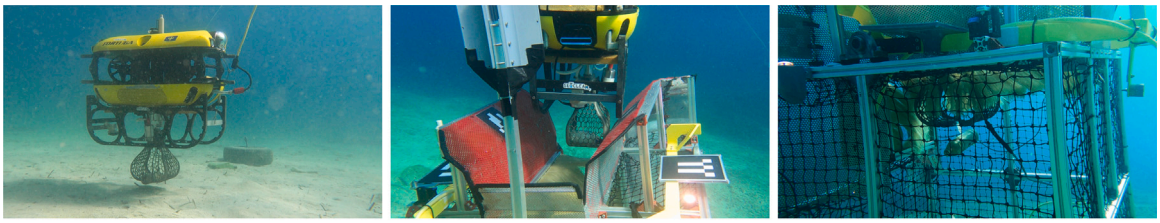
5.2.6. Litter collection

Fig. 27 shows the collection and deposition of litter during the field test in Lokrum, Dubrovnik, Croatia. The Collection ROV showcased sufficiently accurate path following (within the magnitude of the localization errors) to reach the mapped targets. The visual-servoing approach managed to keep detected objects in the field of view of the downward-facing camera, leading to successful grasping and drop-off of the litter in the basket. The autonomous grasping procedure was fully validated in the field tests in Bistrina, with 3 successful automatic collections of a plastic bottle and two sandals. One collection attempt failed as – quite close to the litter target – a part of the seafloor structure was identified as litter with a higher confidence, leading the tracking to switch to that and grab the empty floor. In the Lokrum trials, the system operated under visual operator support to account for variable environmental and visual conditions. Here we performed 5 collection attempts, collecting 4 objects (bottles, can, glass jar). One collection sequence was reinitiated when the litter item was insufficiently secured in the gripper during lift-off and was completed on the second try. All target litter objects were successfully collected, resulting in a 100% collection success rate. It should be noted that due to a very limited 30-minute collection time-slot for the tests in Lokrum's nature reserve (governmental restrictions), 4 out of the 7 mapped items were collected and deposited in the basket.

Remark. The first failure case was not previously seen in the conducted simulated collection attempts due to the much finer-grained simulated seafloor texture. Visual structures on that texture were small enough to not be identified as objects even at a close view of the ROV camera. The second failure case with the insufficiently stable grasp also occurred in simulation and showed the necessity for mitigation through reinitializing the collection attempt once the object was not detected by the camera anymore during transportation.

5.2.7. Tethered UAV control

The effectiveness of the proposed control structure for the tethered UAV was showcased during the final SeaClear demonstrations in Dubrovnik, Croatia. The UAV LARS system was deployed from the support vessel *Naše more*. The system demonstrated its capability to autonomously execute the entire flight task, therefore validating the UAV control approach.



(a) Litter collection (b) Basket entry (c) Litter deposition

Fig. 27. Litter collection sequence during field trials at the Lokrum, Dubrovnik, Croatia test site. Figure (a) shows the grasping of a bottle on the seafloor after identification with the in-gripper camera. In Figure (b), the collection ROV enters the funnel of the basket, which helps for guidance and a stable fit of the ROV on top of the basket. Aruco markers indicate the entry position and orientation of the funnel. Figure (c) shows the ROV stationary on the basket. The gripper entered the litter retainer and is opening to deposit the bottle.

Table B.3
SeaCat USV.

Type	Catamaran, aluminum hull, inflatable lateral tubes
Dimensions	L: 6.83 m, W: 3.1 m, H: 2.15 m (inflated) L: 5.78 m, W: 2.15 m, H: 2.15 m (deflated)
Draught	0.73 m (empty), 0.9 m with max payload
Weight	1075 kg without payload
Payload	500 kg
Max. speed	6 knots
Max sea state	4 (operation), 6 (transit)
Power supply	Generator, 8 kW diesel engine
Tank capacity	2 x 200 L for up to 8 days autonomy
Propulsion system	2 unidirectional Aziprop electrical thrusters
Mode of operation	Manual or auto navigation
Navigation sensors	High-resolution multibeam echosounder, 2 full HD video cameras, DGPS (RTK in option), Gyrocompass, Radar, AIS, INS
Communication	2.4 GHz WiFi antennas with 5 km range, 2 back-up UHF antennas (900/1200 MHz and 430 MHz), 4 underwater GPS receivers
Sensor data interfaces	Serial, USB, Ethernet

Table B.4
Observation (mini Tortuga) ROV.

Type	Inspection class ROV
Dimensions	L: 672 mm, W: 310 mm, H: 381 m
Weight	20 kg in air, neutral in water
Payload	5 kg (wet weight) and 10 kg (dry weight)
Max. speed	4 knots
Power supply	220 Vac (main power or generator)
Propulsion system	4 horizontal thrusters (3 kgf per thruster) with azimuthal adjustment and 2 vertical thrusters (3 kgf per thruster)
Mode of operation	Manual or auto navigation
Sensors	Full-HD video front camera pan & tilt (1920 × 1080 p), LED lights (2x 10 000 lumen), compass, pressure (depth) sensor, temperature, leak detector, AHRS (inertial navigation system), multibeam sonar Oculus 750d, USBL, altimeter, DVL Teledyne Wayfinder, WaterLinked A50
Tether	500 m, 9.5 mm neutral/8 mm negative buoyant, breaking strength 500 kg
Communication	Power line communication (PLC)
Sensor data interfaces	Serial, USB, Ethernet

5.3. Cost analysis

A preliminary cost-effectiveness study estimates a daily operating cost of 2584 € for one SeaClear team (considering two operators) and 2000 €/day for a two-diver team (Hertel-ten Eikelder, 2021). In addition, ship-based alternative approaches are priced at 2–3 €/m^2. For the Hamburg port for an area of 38000 m^2, it is estimated that SeaClear will cost approximately €11365 € – 22730 € while for a ship-based approach, the cost can reach 76000 €. In Dubrovnik, for an area of 8400 m^2 SeaClear estimated cost is 3390 € – 9040 €, while divers are 4667 € at 10 m depth and 8660€ at 20 m depth. Further information about the cost-effectiveness of the SeaClear approach can be found in Hertel-ten Eikelder (2021).

6. Conclusions and future work

This paper provided a comprehensive description and analysis of the SeaClear system, the first autonomous multi-robot system to collect litter from the seabed using advanced control and AI methods. This description includes a detailed explanation of the system's infrastructure, the function of each module, and the design strategy for the sensing and control scheme of the multi-robot platform. Field tests in Hamburg, Germany, and near Dubrovnik, Croatia, demonstrated autonomous detection, mapping, and collection capabilities, showcasing the system's performance as well as the potential of robotics technology to assist or reduce human involvement, enhance operational efficiency, and minimize the risk associated with marine litter collection.

Table B.5
Collection (Tortuga) ROV.

Type	Inspection class ROV
Dimensions	L: 996 mm, W: 430 mm, H: 461 mm (without gripper)
Weight	40 kg in air, neutral in water (without gripper)
Payload	5 kg (wet weight) and 10 kg (dry weight)
Max. speed	4 knots
Power supply	220 Vac (main power or generator)
Propulsion system	4 horizontal thrusters (3 kgf per thruster) with azimuthal adjustment and 2 vertical thrusters (3 kgf per thruster)
Mode of operation	Manual or auto navigation
Sensors	Full-HD video front camera pan & tilt (1920 × 1080 p), rear camera pan & tilt, compass, pressure (depth) sensor, temperature, leak detector, AHRS (inertial navigation system), multibeam sonar, USBL, altimeter, DVL
Auto functions	Heading, depth, altitude
Tether	500 m, 9.5 mm neutral/8 mm negative buoyant, breaking strength 500 kg
Communication	Power line communication (PLC)
Sensor data interfaces	Serial, USB, Ethernet

Table B.6
DJI Matrice 210 RTK V2 UAV.

Dimensions (unfolded)	L: 883 mm, W: 886 mm, H: 427 mm
Weight	4.69 kg
Payload	1.23 kg
Max. speed	Ascent: 5 m/s Descent: 3 m/s
Max. wind resistance	12 m/s
Flight time	No limit due to umbilical power supply
Integrated sensors	Dual barometer and dual IMU, upward infrared sensor, first-person view (FPV) forward vision system, aircraft status indicator, antennas (relay aircraft control and video signals)
Hovering accuracy (GPS)	Vertical: ± 0.5 m Horizontal: ± 1.5 m
Hovering accuracy (D-RTK2)	Vertical ± 0.1 m Horizontal: ± 0.1 m
Tether length	40 m
Camera	Zenmuse X5S (3-axis stabilization)

As lessons learned, it is critical to highlight the inherent difficulties encountered in accurate sensing within underwater environments. Precise estimation of the position and orientation of the ROVs poses a non-trivial challenge, yet these metrics are essential for tasks like mapping, path planning, and picking up small objects. Moreover, the efficacy of object detection is heavily reliant on sensor quality, while underwater conditions, such as water properties and lighting, often differ significantly from those in the training set, creating a discrepancy between the algorithm's training environment and its operational setting. In addition, persistence of detection over a longer timespan with shifting points of view and lighting conditions is much more challenging than object detection in single image frames, but crucial for the continuous tracking during visual-servo control. Finally, it should be emphasized that significant challenges arise in the transition from simulation to real-world testing in validating the architecture due to the constrained and often simplified environment and hardware representations. While the fidelity of the simulation is still insufficient for a full evaluation of the system, it proved to be helpful for the prototyping and validation of the logic of the system-wide workflow and the state transitions. Transferring results from the simulator to field experiments turned out to be especially challenging with respect to optical properties of water, sediment disturbance, or biological growth that affects perception. With recent developments in bridging close-to-photorealistic game engines with ROS (e.g., Unity3D), future work will focus on improving especially the visual, camera-based, environmental representation to evaluate diverse conditions prior to field experiments. Furthermore, the digital twins of the test sites capture only relatively favorable conditions for the operation of an underwater system, such as good lighting, shallow environments and limited currents. More challenging, deeper environments, as well as utilizing techniques such as domain randomization, will likely exhibit a larger sim2real gap then the one currently observed, with additional robustification requirements on the system algorithms and controllers.

Future work will focus on extending the current computer vision architecture and integrating it with additional state-of-the-art computer vision algorithms. Moreover, further analysis is necessary to enhance the interpretability of the neural networks in use, which is critical for validating their decisions in underwater environments (Selvaraju et al., 2017). Developing effective data augmentation strategies is also crucial to address the variability encountered in different underwater environments. Another important topic for future research concerns the quantification of the effect of environmental variability on the perception performance. This will be achieved by integrating uncertainty-aware strategies to weight sensor measurements under varying turbidity and lighting conditions, enabling the system to adapt to unseen degradation in diverse underwater conditions (Tian et al., 2020).

To enhance perception capabilities, future developments will enable the fusion of camera and sonar data.[12] This fusion aims to exploit the complementary strengths of both sensing modalities, where sonar provides reliable structural information in highly turbid or low-light conditions, and the camera captures detailed texture and color information under favorable visibility. By combining these sources, the system will be capable of maintaining robust object detection even under severe visual degradation or strong currents. In addition, the collection of a more diverse and representative dataset covering a wide range of environmental, geographic, and depth conditions will be a key objective. This will include data acquisition in varying marine environments to evaluate and improve the system's generalization ability.

A few important next steps to improve mapping are: better pose estimation by using better sensors and algorithms; using octtree litter representations instead of point clouds, together with Bayesian fusion from cameras and sonars; and developing a self-localization and mapping (SLAM) pipeline. For precise control of the Collection ROV during the grasping operation, we plan on including online learning-based control strategies that account for uncertainties in the environment, such as currents, as well as unknown dynamics of the system, e.g., mass and drag changes from picking up litter. Furthermore, we plan to increase the capabilities of the system for lifting larger and heavier items. In addition, we intend to enhance the robustness of ROV tracking from the UAV through the implementation of tracking failure detection mechanisms.

Ultimately, the paramount goal is to develop a comprehensive strategy to tackle the complete life cycle of marine litter. This can be achieved not only by scaling up the current SeaClear architecture to multiple teams of autonomous robots, and validating it under harsher operating conditions, such as deeper, high-turbidity waters, stronger currents, but also by empowering community involvement and taking steps to raise public awareness about marine litter as we currently do in the following SeaClear2 project. We hope that this research will pave the way for future investigation and advancements in the field, contributing to maintaining cleaner seas.

CRediT authorship contribution statement

Athina Ilioudi: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Stefan Sosnowski:** Writing – review & editing, Writing – original draft, Visualization, Supervision, Resources, Funding acquisition, Conceptualization. **Elisabeth Banken:** Writing – original draft, Validation, Methodology, Investigation. **Petar Bevanda:** Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Jan Brüdigam:** Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Lucian Buşoniu:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Conceptualization. **Yves Chardard:** Resources, Project administration, Funding acquisition. **Cosmin Delea:** Writing – original draft, Visualization, Software, Resources, Methodology, Investigation, Funding acquisition, Conceptualization. **Bart De Schutter:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition. **Antun Đuraš:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Claudia Hertel-ten Eikelder:** Resources, Project administration, Funding acquisition. **Shahab Heshmati-Alamdari:** Writing – original draft, Supervision. **Vicu-Mihalis Maer:** Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization.

[12] https://www.seaclear2.eu/

Ivana Palunko: Supervision, Resources, Project administration, Funding acquisition, Conceptualization. **Iva Pozniak:** Resources, Project administration, Funding acquisition. **Vicko Prkačin:** Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Domagoj Tolić:** Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by European Union's Horizon 2020 research and innovation programme under the grant agreement no. 871295 "SeaClear" (SEarch, identificAtion and Collection of marine Litter with Autonomous Robots), and by European Union's Horizon Europe programme under the grant agreement no. 101093822 "SeaClear2.0". We would like to thank Michael Vogt for his technical support on the simulator.

Appendix A. Field of view requirements

Let us first define vectors ${}^{\mathcal{O}}\boldsymbol{p}_c = [{}^{\mathcal{O}}x_c, {}^{\mathcal{O}}y_c, {}^{\mathcal{O}}z_c]^\top$ and ${}^{\mathcal{O}}\boldsymbol{\phi}_c = [{}^{\mathcal{O}}\phi_c, {}^{\mathcal{O}}\theta_c, {}^{\mathcal{O}}\psi_c]^\top$ describing the pose of the camera frame C_c with respect to the object frame $C_{\mathcal{O}}$, and α_h and α_v the field of view angles in horizontal and vertical axes (see Fig. 12).

Considering y_T and z_T to be the dimensions of the object with respect to the object frame, the following inequalities describe the field of view constraints of the systems, that must be satisfied in order to always maintain the object within the camera field of view. These requirements are captured by the constraint set H_C:

$$ {}^{\mathcal{O}}\boldsymbol{\eta}_C = \left[{}^{\mathcal{O}}\boldsymbol{p}_C^T \; {}^{\mathcal{O}}\boldsymbol{\phi}_C^T\right]^T \in H_C \tag{A.1}$$

which can be defined as:

$$ -{}^{\mathcal{O}}y_c + {}^{\mathcal{O}}z_c \tan\left(-{}^{\mathcal{O}}\phi_c - \frac{\alpha_h}{2}\right) - \frac{y_T}{2} \geq 0 \tag{A.2a}$$

$$ {}^{\mathcal{O}}y_c - {}^{\mathcal{O}}z_c \tan\left(-{}^{\mathcal{O}}\phi_c + \frac{\alpha_h}{2}\right) - \frac{y_T}{2} \geq 0 \tag{A.2b}$$

$$ {}^{\mathcal{O}}x_c + {}^{\mathcal{O}}z_c \tan\left({}^{\mathcal{O}}\theta_c - \frac{\alpha_v}{2}\right) - \frac{z_T}{2} \geq 0 \tag{A.2c}$$

$$ -{}^{\mathcal{O}}x_c - {}^{\mathcal{O}}z_c \tan\left({}^{\mathcal{O}}\theta_c + \frac{\alpha_v}{2}\right) - \frac{z_T}{2} \geq 0 \tag{A.2d}$$

For locally grasping the identified object of interest, we consider that the vision tracking algorithm provides in real-time the pose of camera with respect to the local inertial frame $\mathcal{I}$ i.e., ${}^{\mathcal{I}}\boldsymbol{p}_c$ and ${}^{\mathcal{I}}\boldsymbol{\phi}_c$ with $C_I \approx C_{\mathcal{O}}$, as well as an estimation of the object dimensions, i.e., y_T and z_T as a bounding box along the horizontal and vertical axes on the object frame. In the SeaClear system, this is fulfilled by the object detection algorithm updating the bounding box and center of geometry with 30 fps.

Appendix. Technical specifications

See Tables B.3–B.6.

Data availability

Data will be made available on request.

References

Babić, A., Ferreira, F., Kapetanović, N., Mišković, N., Bibuli, M., Bruzzone, G., Motta, C., Ferretti, R., Odetti, A., Caccia, M., Aracri, S., De Pascalis, F., 2023. Cooperative marine litter detection and environmental monitoring using heterogeneous robotic agents. In: OCEANS 2023 - Limerick. pp. 1–6. http://dx.doi.org/10.1109/OCEANSLimerick52467.2023.10244657.

Bingham, B., Agüero, C., McCarrin, M., Klamo, J., Malia, J., Allen, K., Lum, T., Rawson, M., Waqar, R., 2019. Toward maritime robotic simulation in gazebo. In: OCEANS 2019 MTS/IEEE SEATTLE. pp. 1–10. http://dx.doi.org/10.23919/OCEANS40490.2019.8962724.

Bulić, D., Tolić, D., Palunko, I., 2022. Beam-based tether dynamics and simulations using finite element model. In: 6th IFAC Conference on Intelligent Control and Automation Sciences ICONS 2022. pp. 154–159. http://dx.doi.org/10.1016/j.ifacol.2022.07.624, URL: https://www.sciencedirect.com/science/article/pii/S2405896322010357.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., 2020. End-to-end object detection with transformers. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (Eds.), Computer Vision – ECCV 2020. Springer International Publishing, Cham, pp. 213–229.

Cózar, A., Echevarría, F., González-Gordillo, J.I., Irigoien, X., Úbeda, B., Hernández-León, S., Palma, A.T., Navarro, S., García-de Lomasx, J., Ruiz, A., de Puelles María L., F., Duarte, C.M., 2014. Plastic debris in the open ocean. Proc. Natl. Acad. Sci. 111 (28), 10239–10244. http://dx.doi.org/10.1073/pnas.1314705111, URL: https://www.pnas.org/doi/abs/10.1073/pnas.1314705111 arXiv:https://www.pnas.org/doi/pdf/10.1073/pnas.1314705111.

Hertel-ten Eikelder, C., 2021. Technical Report D2.3, SeaClear Consortium, URL: https://seaclear-project.eu/results/public-deliverables.

Fossen, T., 1994. Guidance and Control of Ocean Vehicles. Wiley, New York.

Fossen, T.I., 1999. Guidance and control of ocean vehicles. (Doctors Thesis). University of Trondheim, Norway, ISBN: 0 471 94113 1. Printed by John Wiley & Sons, Chichester, England.

Furrer, F., Burri, M., Achtelik, M., Siegwart, R., 2016. Robot operating system (ROS): The complete reference (volume 1). Springer International Publishing, Cham, pp. 595–625. http://dx.doi.org/10.1007/978-3-319-26054-9_23.

Gammell, J.D., Srinivasa, S.S., Barfoot, T.D., 2014. Informed rrt*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 2997–3004. http://dx.doi.org/10.1109/IROS.2014.6942976.

Gao, B., Spratling, M.W., 2022. More robust object tracking via shape and motion cue integration. Signal Process. 108628.

González-Morgado, A., Smits, S., Heredia, G., Ollero, A., Krupa, A., Chaumette, F., Spindler, F., Franchi, A., Gabellieri, C., 2025. Multi-robot aerial soft manipulator for floating litter collection. arXiv:2507.03517. URL: https://arxiv.org/abs/2507.03517 arXiv:2507.03517.

Goutte, C., Gaussier, E., 2005. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In: Losada, D.E., Fernández-Luna, J.M. (Eds.), Advances in Information Retrieval. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 345–359.

Gouttefarde, M., Rodriguez, M., Barrelet, C., Hervé, P.-E., Creuze, V., Gorrotxategi, J., Oyarzabal, A., Culla, D., Sallé, D., Tempier, O., Ferrari, N., Chaumont, M., Subsol, G., 2023. The robotic seabed cleaning platform: An underwater cable-driven parallel robot for marine litter removal. In: Caro, S., Pott, A., Bruckmann, T. (Eds.), Cable-Driven Parallel Robots. Springer, Cham, pp. 430–441.

Henderson, P., Ferrari, V., 2017. End-to-end training of object class detectors for mean average precision. In: Lai, S.-H., Lepetit, V., Nishino, K., Sato, Y. (Eds.), Computer Vision – ACCV 2016. Springer International Publishing, Cham, pp. 198–213.

Heshmati-Alamdari, S., Eqtami, A., Karras, G.C., Dimarogonas, D.V., Kyriakopoulos, K.J., 2014. A self-triggered visual servoing model predictive control scheme for under-actuated underwater robotic vehicles. In: 2014 IEEE International Conference on Robotics and Automation. ICRA, IEEE, pp. 3826–3831.

Heshmati-Alamdari, S., Karras, G.C., Marantos, P., Kyriakopoulos, K.J., 2018. A robust model predictive control approach for autonomous underwater vehicles operating in a constrained workspace. In: 2018 IEEE International Conference on Robotics and Automation. ICRA, pp. 6183–6188. http://dx.doi.org/10.1109/ICRA.2018.8460918.

Hornung, A., Wurm, K.M., Bennewitz, M., Stachniss, C., Burgard, W., 2013/04/01. OctoMap: an efficient probabilistic 3D mapping framework based on octrees. Auton. Robots 34 (3), 189–206. http://dx.doi.org/10.1007/s10514-012-9321-0.

Ilioudi, A., Dabiri, A., Wolf, B.J., De Schutter, B., 2022. Deep learning for object detection and segmentation in videos: Toward an integration with domain knowledge. IEEE Access 10, 34562–34576. http://dx.doi.org/10.1109/ACCESS.2022.3162827.

Ilioudi, A., Wolf, B.J., Dabiri, A., De Schutter, B., 2023. Towards establishing an automated selection framework for underwater image enhancement methods. In: OCEANS 2023 - Limerick. pp. 1–6. http://dx.doi.org/10.1109/OCEANSLimerick52467.2023.10244710.

Isobe, A., Iwasaki, S., 2022. The fate of missing ocean plastics: Are they just a marine environmental problem? Sci. Total Environ. 825, 153935. http://dx.doi.org/10.1016/j.scitotenv.2022.153935, URL: https://www.sciencedirect.com/science/article/pii/S0048969722010270.

Jia, T., Kapelan, Z., de Vries, R., Vriend, P., Peereboom, E.C., Okkerman, I., Taormina, R., 2023. Deep learning for detecting macroplastic litter in water bodies: A review. Water Res. 231, 119632. http://dx.doi.org/10.1016/j.watres.2023.119632.

Jian, M., Yang, N., Tao, C., Zhi, H., Luo, H., 2024. Underwater object detection and datasets: a survey. Intell. Mar. Technol. Syst. 2 (1), 9. http://dx.doi.org/10.1007/s44295-024-00023-6.

Kong, S., Tian, M., Qiu, C., Wu, Z., Yu, J., 2021. IWSCR: An intelligent water surface cleaner robot for collecting floating garbage. IEEE Trans. Syst. Man Cybern.: Syst. 51 (10), 6358–6368. http://dx.doi.org/10.1109/TSMC.2019.2961687.

Lindeberg, T., 1998. Feature detection with automatic scale selection. Int. J. Comput. Vis. 30 (2), 79–116. http://dx.doi.org/10.1023/a:1008045108935.

Liu, X., Goldsmith, A., 2004. Kalman filtering with partial observation losses. CDC, In: 2004 43rd IEEE Conference on Decision and Control, vol. 4, pp. 4180–4186.

Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. 60 (2), 91–110. http://dx.doi.org/10.1023/b:visi.0000029664.99615.94.

Lv, W., Zhao, Y., Chang, Q., Huang, K., Wang, G., Liu, Y., 2024. RT-DETRv2: Improved baseline with bag-of-freebies for real-time detection transformer. URL: https://arxiv.org/abs/2407.17140 arXiv:2407.17140.

Maer, V.-M., Tamás, L., Buşoniu, L., 2022. Underwater robot pose estimation using acoustic methods and intermittent position measurements at the surface. In: 2022 IEEE International Conference on Automation, Quality and Testing, Robotics. AQTR, pp. 1–6.

Manhães, M.M.M., Scherer, S.A., Voss, M., Douat, L.R., Rauschenbach, T., 2016. UUV simulator: A gazebo-based package for underwater intervention and multi-robot simulation. In: OCEANS 2016 MTS/IEEE Monterey. pp. 1–8. http://dx.doi.org/10.1109/OCEANS.2016.7761080.

Marantos, P., Koveos, Y., Kyriakopoulos, K.J., 2016. UAV state estimation using adaptive complementary filters. IEEE Trans. Control Syst. Technol. 24 (4), 1214–1226. http://dx.doi.org/10.1109/TCST.2015.2480012.

McGlade, J., Samy Fahim, I., Green, D., Landrigan, P., Andrady, A., Costa, M., Geyer, R., Gomes, R., Tan Shau Hwai, A., Jambeck, J., Li, D., Rochman, C., Ryan, P., Thiel, M., Thompson, R., Townsend, K., Turra, A., 2021. From pollution to solution: A global assessment of marine litter and plastic pollution. Technical Report, United Nations Environment Programme.

Moore, T., Stouch, D., 2016. A generalized extended Kalman filter implementation for the robot operating system. In: Advances in Intelligent Systems and Computing, vol. 302, pp. 335–348. http://dx.doi.org/10.1007/978-3-319-08338-4_25.

Natural Seabed, 2024. https://naturalseabed.com/en/. (Accessed 25 March 2024).

RanMarine Technology, 2024. https://www.ranmarine.io/. (Accessed 25 March 2024).

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 779–788. http://dx.doi.org/10.1109/CVPR.2016.91.

Ren, S., He, K., Girshick, R., Sun, J., 2017. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. 39 (6), 1137–1149. http://dx.doi.org/10.1109/TPAMI.2016.2577031.

Rosynski, M., Buşoniu, L., 2022. A simulator and first reinforcement learning results for underwater mapping. Sensors 22 (14), 5384.

Rozumnyi, D., Matas, J., Šroubek, F., Pollefeys, M., Oswald, M.R., 2021. Fmodetect: Robust detection of fast moving objects. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3541–3549.

Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T., 2008. Labelme: A database and web-based tool for image annotation. Int. J. Comput. Vis. 77 (1), 157–173. http://dx.doi.org/10.1007/s11263-007-0090-8.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D., 2017. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In: 2017 IEEE International Conference on Computer Vision. ICCV, pp. 618–626. http://dx.doi.org/10.1109/ICCV.2017.74.

Sherrington, C., 2016. Plastics in the Marine Environment. Technical Report, Eunomia Research and Consulting Ltd, URL: https://eunomia.eco/reports/plastics-in-the-marine-environment/.

Sukno, M., Palunko, I., 2022. Hand-crafted features for floating plastic detection. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS, pp. 3378–3383. http://dx.doi.org/10.1109/IROS47612.2022.9981320.

Terven, J., Córdova-Esparza, D.-M., Romero-González, J.-A., 2023. A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS. Mach. Learn. Knowl. Extr. 5 (4), 1680–1716. http://dx.doi.org/10.3390/make5040083, URL: https://www.mdpi.com/2504-4990/5/4/83.

The Ocean Cleanup, 2024. https://theoceancleanup.com/. (Accessed 25 March 2024).

The Rozalia Project, 2024. https://www.rozaliaproject.org. (Accessed 25 March 2024).

The SeaCleaners, 2024. https://www.theseacleaners.org. (Accessed 25 March 2024).

Tian, J., Cheung, W., Glaser, N., Liu, Y.-C., Kira, Z., 2020. UNO: Uncertainty-aware noisy-or multimodal fusion for unanticipated input degradation. In: 2020 IEEE International Conference on Robotics and Automation. ICRA, pp. 5716–5723. http://dx.doi.org/10.1109/ICRA40945.2020.9197266.

Đuraš, A., Ilioudi, A., Wolf, B., Palunko, I., De Schutter, B., 2024b. Seaclear marine debris detection and segmentation dataset. http://dx.doi.org/10.4121/4F1DFF25-E157-4399-A5D4-478055461689.V1, URL: https://data.4tu.nl/datasets/4f1dff25-e157-4399-a5d4-478055461689/1.

Đuraš, A., Sukno, M., Palunko, I., 2022. Recovering the 3D UUV position using UAV imagery in shallow-water environments. In: 2022 International Conference on Unmanned Aircraft Systems. ICUAS, pp. 948–954. http://dx.doi.org/10.1109/ICUAS54217.2022.9836195.

Đuraš, A., Wolf, B.J., Ilioudi, A., Palunko, I., De Schutter, B., 2024a. A dataset for detection and segmentation of underwater marine debris in shallow waters. Sci. Data 11 (1), 921. http://dx.doi.org/10.1038/s41597-024-03759-2.

Valdenegro-Toro, M., 2016. Submerged marine debris detection with autonomous underwater vehicles. In: 2016 International Conference on Robotics and Automation for Humanitarian Applications. RAHA, pp. 1–7. http://dx.doi.org/10.1109/RAHA.2016.7931907.

Wang, T., Joo, H.-J., Song, S., Hu, W., Keplinger, C., Sitti, M., 2023. A versatile jellyfish-like robotic platform for effective underwater propulsion and manipulation. Sci. Adv. 9 (15), eadg0292. http://dx.doi.org/10.1126/sciadv.adg0292, URL: https://www.science.org/doi/abs/10.1126/sciadv.adg0292. arXiv:https://www.science.org/doi/pdf/10.1126/sciadv.adg0292.

Wang, S., Xia, C., Lv, F., Shi, Y., 2024. RT-DETRv3: Real-time end-to-end object detection with hierarchical dense positive supervision. URL: https://arxiv.org/abs/2409.08475 arXiv:2409.08475.

Watanabe, J.-I., Shao, Y., Miura, N., 2019. Underwater and airborne monitoring of marine ecosystems and debris. J. Appl. Remote. Sens. 13 (4), 044509. http://dx.doi.org/10.1117/1.JRS.13.044509.

Weston, J., Tolić, D., Palunko, I., 2024. Application of hamilton-Jacobi-bellman equation/pontryagin's principle for constrained optimal control. J. Optim. Theory Appl. 200 (2), 437–462. http://dx.doi.org/10.1007/S10957-023-02364-4.

World Economic Forum and Ellen MacArthur Foundation and McKinsey & Company, 2016. The New Plastics Economy: Rethinking the future of plastics. [Online]. Available: https://www.ellenmacarthurfoundation.org/the-new-plastics-economy-rethinking-the-future-of-plastics.

Yang, X., Chen, Y., Zhou, Y., Tong, F., 2025. A three-dimensional marine plastic litter real-time detection embedded system based on deep learning. Marine Poll. Bull. 213, 117603. http://dx.doi.org/10.1016/j.marpolbul.2025.117603.

Yousuf, B., Lendek, Z., Buşoniu, L., 2022. Exploration-based search for an unknown number of targets using a UAV. IFAC-PapersOnLine 55 (15), 93–98.

Zhang, Y., Huang, Z., Chen, C., Wu, X., Xie, S., Zhou, H., Gou, Y., Gu, L., Ma, M., 2023. A spiral-propulsion amphibious intelligent robot for land garbage cleaning and sea garbage cleaning. J. Mar. Sci. Eng. 11 (8), http://dx.doi.org/10.3390/jmse11081482, URL: https://www.mdpi.com/2077-1312/11/8/1482.

Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., Chen, J., 2024. DETRs beat YOLOs on real-time object detection. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 16965–16974. http://dx.doi.org/10.1109/CVPR52733.2024.01605.