

Reconfigurable AUV for intervention missions: a case study on underwater object recovery

Mario Prats · David Ribas · Narcís Palomeras · Juan Carlos García ·
Volker Nannen · Stephan Wirth · José Javier Fernández · Joan P.
Beltrán · Ricard Campos · Pere Ridao · Pedro J. Sanz · Gabriel
Oliver · Marc Carreras · Nuno Gracias · Raúl Marín · Alberto Ortiz

Abstract Starting in January 2009, the RAUVI (Reconfigurable Autonomous Underwater Vehicle for Intervention Missions) project is a three years coordinated research action funded by the Spanish Ministry of Research and Innovation. In this paper, the state of progress after two years of continuous research is reported. As a first experimental validation of the complete system, a Search & Recovery problem is addressed, consisting of finding and recovering a flight data recorder placed at an unknown position at the bottom of a water tank. An overview of the techniques used to successfully solve the problem in an autonomous way is provided. The obtained results are very promising and are the first step toward the final test in shallow water at the end of 2011.

Keywords Underwater robotics · Intervention AUV · Autonomous underwater manipulation · Underwater computer vision · Graphical user interfaces

The RAUVI project is a multi-disciplinary approach to underwater intervention that involves different research groups at three universities: Jaume-I University, University of Girona and University of Balearic Islands

M. Prats · J. C. García · J. J. Fernández · P. J. Sanz · R. Marín

University of Jaume I, Spain
e-mail: mprats@uji.es

D. Ribas · N. Palomeras · R. Campos · P. Ridao · M. Carreras · N. Gracias

University of Girona, Spain
e-mail: dribas@udg.edu

V. Nannen · S. Wirth · J. P. Beltrán · G. Oliver · A. Ortiz
University of Balearic Islands, Spain
e-mail: vnannen@gmail.com

1 Introduction

Unmanned underwater vehicles that are used in maritime field operations often need intervention capabilities in order to complete the desired task. Typical applications include the offshore industries, where unmanned underwater vehicles dock for example to an underwater panel in order to manipulate valves with a robotic arm; marine scientists need the capability to accurately deploy and recover specialized instruments from the seabed; in the context of permanent underwater observatories that are currently under design and development, intervention capability is vital for maintenance operations; in marine rescue operations, intervention capabilities are needed to establish contact and perhaps free personal that is trapped underwater, as was for example desperately needed during the 2000 Kursk tragedy.

Currently, most intervention operations are performed by manned submersibles endowed with robotic arms or by Remotely Operated Vehicles (ROVs). Manned submersibles have the advantage of placing the operator in the field of operation with direct view to the object being manipulated. Their drawbacks are the reduced time for operation (typically in the order of a few hours) the human presence in a dangerous and hostile environment, and a very high cost of the associated oceanographic vessel. Work class ROVs are currently the preferred technology for deep water intervention. They can be remotely operated for days without problems. Nevertheless, they still need an expensive oceanographic vessel with a heavy crane and automatic Tether Management System (TMS) and a Dynamic Position

system (DP). The cognitive fatigue of the operator who has to take care of the umbilical and the ROV while cooperating with the operator of the robotic arms is remarkable.

For these reasons, some researchers have recently started to think about the natural evolution of the intervention ROV, the Intervention AUV (I-AUV). Without the need for the TMS and the DP, light I-AUVs could theoretically be operated from cheap vessels of opportunity, considerably reducing the cost of operation. Considering the fast development of battery technology, and removing the operator from the control loop, one can start to think about intervention operations that last for several days, where a ship is only needed on the first and the last day for launch and recovery.

But this fascinating scenario, where I-AUVs do the work autonomously, comes at the cost of endowing the robot with the intelligence needed to keep the operator out of the control loop. Although standard AUVs are also operated without human intervention, they are constrained to survey operations, commonly flying at a safe altitude with respect to the ocean floor while logging data. I-AUVs must be operated in the close proximity of the seabed or artificial structures. They have to be able to identify the objects to be manipulated and the intervention tasks to be undertaken, while safely moving within a cluttered work area. While I-AUVs are the natural way of technological progress, they represent an authentic research challenge for the Robotics community. Moreover, the I-AUVs that have been developed until now, and which have proven field capabilities, are heavy vehicles intended for very deep water interventions. E.g., the SAUVIM [1] and ALIVE [2] vehicles weight 6 and 3.5 ton respectively. It is a fact that science and industry are interested in the design and development of a very light I-AUV (< 300 kg) that is constrained to shallow water interventions in depths up to 300 m. The construction of an I-AUV that is able to perform intervention activities completely autonomously, and can be validated experimentally in a realistic scenario with a real prototype, would constitute a technological milestone. This is in fact the aim of the RAUVI project [3].

To foster further research and development of the project, a Search & Recovery (S&R) testbed application has been selected (see Figure 1). A typical S&R mission is the recovery of a Flight Data Recorder (FDR, also known as black-box) from a crashed airplane. Flight recorders are typically equipped with a 27-39 KHz pinger (e.g. Benthos) that periodically emits an acoustic signal that is audible up to a distance of approximately one kilometer. The acoustic beacon will begin to emit

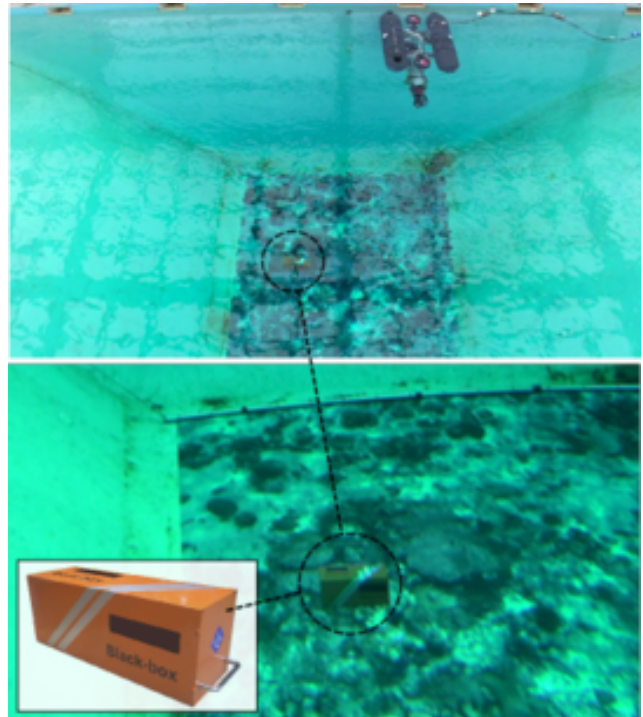


Fig. 1 The test scenario at CIRS (Univ. of Girona). An I-AUV has to autonomously search for a flight data recorder, placed at an unknown position in a water tank, and recover it

when immersed in water and the ping will last until the battery is exhausted, typically around one month later. The time limitation forces the search method to be as efficient as possible. For the experiments presented in this article, it is assumed that the FDR has already been localized within a small area. The paper is focused on the local vision-based search and recovery.

Few technical papers discuss black box recovery with the aid of an underwater intervention vehicle. All examples in the literature describe the use of ROV vehicles. To the best of the authors' knowledge, an autonomous vehicle has never been used for a black box recovery mission, likely due to the high complexity of this task. Only some theoretic papers are available that describe prospective work [4].

The remainder of this paper is organized as follows. Section 2 presents the evolution of the I-AUV concept under development and introduces details of both the vehicle and the robot arm. Section 3 shows an overview of the global control architecture. Sections 4 and 5 describe the user interface and 3D simulation module. Section 6 introduces the main characteristics of the vision system under development. Experimental results of an S&R mission are presented in Section 7. Section 8 offers a discussion and conclusive remarks.

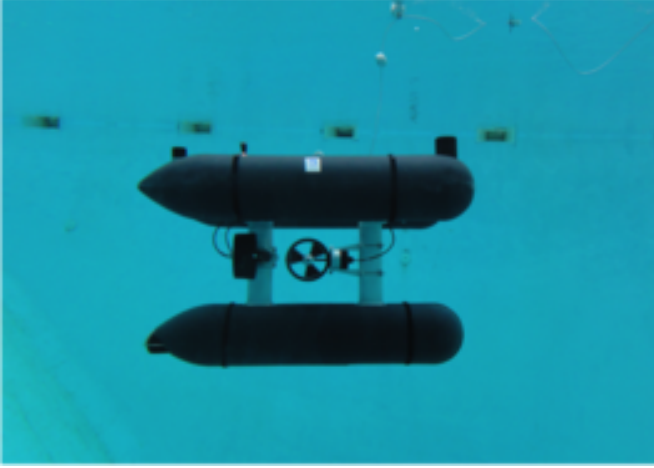


Fig. 2 The GIRONA 500 AUV in a survey configuration

2 The I-AUV developed

2.1 The autonomous underwater vehicle

The GIRONA 500 is a reconfigurable autonomous underwater vehicle (AUV) designed for a maximum operating depth of up to 500 m (see figure 2). The vehicle is composed of an aluminum frame which supports three torpedo-shaped hulls of 0.3 m in diameter and 1.5 m in length as well as other elements like the thrusters. This design offers a good hydrodynamic performance and a large space for housing the equipments while maintaining a compact size which allows operating the vehicle from small boats. The overall dimensions of the vehicle are 1 m in height, 1 m in width, 1.5 m in length and a weight of less than 200 Kg. The two upper hulls, which contain the flotation foam and the electronics housing, are positively buoyant, while the lower one contains the more heavy elements such as the batteries and the payload. This particular arrangement of the components separates the centre of gravity from the centre of buoyancy by about 11 cm, which is significantly more than found in a typical torpedo shape design. This provides the vehicle with passive stability in pitch and roll, making it suitable for tasks that will benefit from a steady platform such as interventions or imaging surveys.

The most remarkable characteristic of the GIRONA 500 is its capacity to reconfigure for different tasks. In its standard configuration, the vehicle is equipped with typical navigation sensors (DVL, AHRS, pressure gauge and USBL) and basic survey equipment (profiler sonar, side scan sonar, video camera and sound velocity sensor). In addition to these sensors, almost half the volume of the lower hull is reserved for payload equipment that can be configured according to the requirements of a particular mission. The electric arm that will be

presented in the following section is the first payload developed for the GIRONA 500. The same philosophy has been applied to the propulsion system, which is also reconfigurable. The basic layout has 4 thrusters, two vertical to actuate the heave and pitch and two horizontal for the yaw and surge. However, it is possible to reconfigure the vehicle to operate with only 3 thrusters (one vertical and two horizontal) and with up to 8 thrusters to control all the degrees of freedom.

2.2 The light-weight underwater arm

The “Light-Weight ARM 5 E” is a robotic manipulator actuated by 24V brushless DC motors. It is composed of four revolute joints, and can reach distances up to one meter. An actuated robot gripper allows for grasping small objects, and its T-shaped grooves also permit handling special tools. The arm is made of aluminium alloy partially covered with foam material in order to guarantee suitable buoyancy. The total weight in the air is about 29 kg, whereas in fresh water it decreases to 12 kg approximately. The arm is capable of lifting 12 kg at full reach, and can descend up to 300 m in water.

An underwater camera can be mounted either on the arm wrist or on the base link in order to provide a top view of the manipulation area. It is a “Bowtech DIVECAM-550C-AL” high-resolution color CCD camera, rated up to 100 m depth. The current configuration of the arm and gripper is shown in Figure 3, together with a planar projection of the manipulator workspace in Figure 4. As can be observed, the most suitable area for manipulation is around 80 cm below the arm base link. This area guarantees the highest distance to the workspace limits and is also free of arm singularities. For the experiments described here, the camera was placed next to the arm base link (denoted as C in Figure 4) and faced downwards. This configuration guarantees that there is an intersection between the camera field of view and the arm workspace that allow to visually control the arm during execution of the task.

Figure 5 shows the current integrated prototype of the I-AUV developed by the RAUVI project.

3 The Control Architecture

The I-AUV control architecture is composed of two initially independent architectures: the underwater vehicle and the manipulator architectures. Both of them have been combined into a new schema that allows for reactive and deliberative behaviors on both subsystems. Reactive actions are performed in the low-level control



Fig. 3 The Light-Weight ARM 5 E with a hook attached to the gripper. The T-shape grooves of the gripper allow attachment of different tools depending on the application

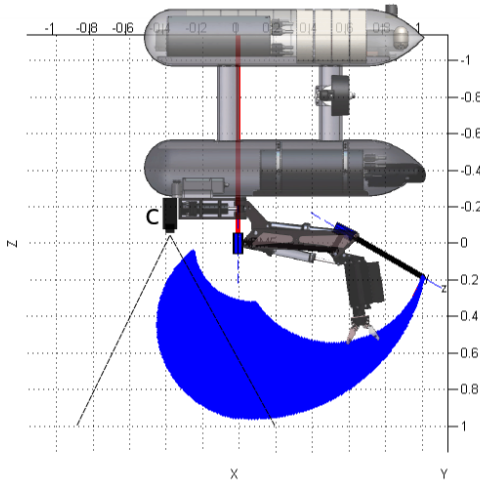


Fig. 4 A projection of the arm workspace, together with the D-H links [5] (marked as blue cylinders) and the current configuration of the camera (C)

layer that communicates with the real or simulated I-AUV via an abstraction interface. On the other hand, the whole mission is supervised at a high-level by a Mission Control System (MCS), implemented using the Petri net formalism. Visual perception services are provided by the vision module described in Section 6. The ROS Robot Operating System [6][7] is used to integrate the heterogeneous computing hardware and software of all system components, to allow for easy integration of additional mission specific components, and to record all sensor input in a suitable playback format for simulation purposes. Vehicle control, the manipulator, and the vision system are implemented as independent ROS nodes that are executed on their own independent hardware units and that communicate through ROS messages over an onboard ethernet network. The general architecture is illustrated in Figure 6. For further detail see [8].

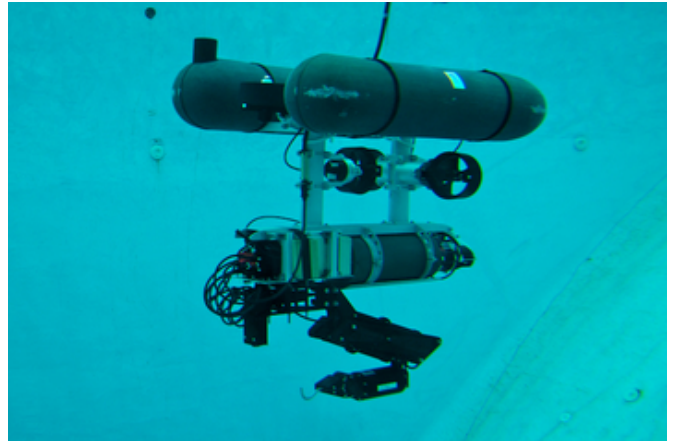


Fig. 5 The integrated I-AUV prototype in a water tank. The cable was for powering the manipulator, that was not electrically integrated with the AUV at that moment

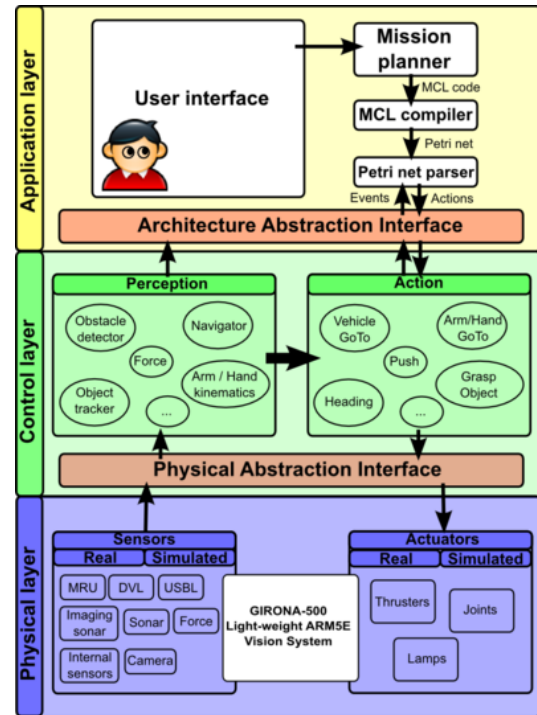


Fig. 6 An overview of the RAUVI software architecture. Communications through the network are implemented via ROS messages

3.1 The Navigation System

The vehicle relies on a dead-reckoning estimate to navigate during the execution of the mission. The estimate is produced by a Kalman filter, which is in charge of integrating the information from different sensors with the predictions from a simple kinematics model. Despite the inherent drift affecting any dead-reckoning estimate, the resulting errors have shown to be acceptable

for the application at hand, where the explored area is small. However, the navigation data may not be reliable enough in large area surveys. To address possible issues related with the accumulation of navigation errors, a framework to integrate absolute position fixes from an USBL system is currently being developed [9]

The information to be estimated by the navigation filter is stored in a state vector that contains information regarding the pose and velocity of a 4 DOF vehicle at time k :

$$\mathbf{x}_k = [x \ y \ z \ \psi \ u \ v \ w \ r]^T, \quad (1)$$

where x , y , z and ψ correspond to the 3D position and heading of the vehicle and u , v , w and r are the corresponding linear/angular velocities.

The prediction stage of the Kalman filter relies on a simple constant velocity kinematics model to predict how the state will evolve from time $k-1$ to time k :

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{n}_{k-1}), \quad (2)$$

$$\begin{bmatrix} x \\ y \\ z \\ \psi \\ u \\ v \\ w \\ r \end{bmatrix}_k = \begin{bmatrix} x + (uT + n_u \frac{T^2}{2})\cos(\psi) - (vT + n_v \frac{T^2}{2})\sin(\psi) \\ y + (uT + n_u \frac{T^2}{2})\sin(\psi) + (vT + n_v \frac{T^2}{2})\cos(\psi) \\ z + wT + n_w \frac{T^2}{2} \\ \psi + rT + n_r \frac{T^2}{2} \\ u + n_u T \\ v + n_v T \\ w + n_w T \\ r + n_r T \end{bmatrix}_{k-1}$$

where $\mathbf{n} = [n_u \ n_v \ n_w \ n_r]^T$ represents a vector of white Gaussian acceleration noises with zero mean. They are additive in the velocity terms and propagate through integration to the position. The covariance of the \mathbf{n} vector is represented by the system noise matrix \mathbf{Q}_k :

$$E[\mathbf{n}_k] = \mathbf{0}, \quad E[\mathbf{n}_k \mathbf{n}_j^T] = \delta_{kj} \mathbf{Q}_k, \quad (3)$$

The standard extended Kalman filter equations are then used to project an estimate of the state $\hat{\mathbf{x}}_k$ and its associated covariance matrix \mathbf{P}_k [10].

The vehicle is equipped with a number of sensors providing direct observations of particular elements of the state vector. The update step of the Kalman filter incorporates this information into the current prediction of the vehicle state by means of a measurement model:

$$\mathbf{z}_k = \mathbf{H}\hat{\mathbf{x}}_k + \mathbf{m}_k, \quad (4)$$

$$\begin{bmatrix} z_u \\ z_v \\ z_w \\ z_z \\ z_\psi \end{bmatrix}_k = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \psi \\ u \\ v \\ w \\ r \end{bmatrix}_k + \begin{bmatrix} m_u \\ m_v \\ m_w \\ m_z \\ m_\psi \end{bmatrix}_k$$

where the z_u , z_v and z_w are the vehicle velocities measured by the DVL, z_z is the depth measurement from the pressure sensor, z_ψ the heading of the vehicle according to the AHRS and \mathbf{m} represents a vector of white Gaussian noises with zero mean affecting the observation process. The covariance matrix of the measurement noise \mathbf{R} is given by:

$$E[\mathbf{m}_k] = \mathbf{0}, \quad E[\mathbf{m}_k \mathbf{m}_j^T] = \delta_{kj} \mathbf{R}_k, \quad (5)$$

The covariance values for the \mathbf{R}_k matrix have been assigned according to the specifications from the manufacturers of each particular sensor. Since the sensors operate asynchronously, the form of the observation matrix \mathbf{H} needs to be adapted, by adding or removing rows, to the measurements available from the sensors at each time step k . Given the proposed linear measurement model, the state is updated by means of the standard Kalman filter equations [10].

The envisioned mission requires the vehicle to follow a survey pattern in search of the object to be retrieved and then, to navigate at a particular position indicated by a human operator to begin the intervention. In this context, the navigation is achieved by defining a trajectory as a set of 2D way-points. A simple Line Of Sight (LOS) algorithm with cross tracking error [11] is employed to guide the robot towards the desired way-point. The localization data provided by the Kalman filter is used to control the path in both the Surge and Yaw degrees of freedom.

3.2 Manipulator control

The arm low-level control electronics are placed in a housing cylinder that uses a PIC micro-controller in order to (i) send/receive RS232 data packages to/from the control PC, and (ii) communicate with each motor micro-controller through a CAN bus. The RS232 communication protocol includes fixed-length motor command and sensor messages. Motor command messages are sent from the PC to the arm, and can be either a control demand in terms of position, speed or voltage, or a PID setting message. When the arm micro-controller receives a motor command message, it performs the corresponding control action and sends back to the PC a sensor message including position, speed, current and temperature of each motor as measured by the internal sensors.

Hall-effect sensors are integrated into the arm motors, thus providing very basic position information. Each motor shaft revolution corresponds to 8 position ticks that are measured with the hall-effect sensors and sent through the RS232 channel to the control PC. These position ticks are relative to the moment where

the arm is powered; they do not provide absolute position feedback. It is therefore necessary to (i) relate position ticks with respect to an absolute reference, and (ii) convert position ticks to actual joint angles and vice-versa.

Reference [12] describes the kinematic modeling of the arm and the planning of a suitable vehicle pose that guarantees that the object is inside the arm workspace. Being \mathbf{v}_E a cartesian velocity to be achieved by the end-effector, it is transformed to arm joint velocities, $\dot{\mathbf{q}}$, via the arm end-effector jacobian Moore-Penrose pseudo-inverse \mathbf{J}_E^+ :

$$\dot{\mathbf{q}} = \mathbf{J}_E^+ \mathbf{v}_E$$

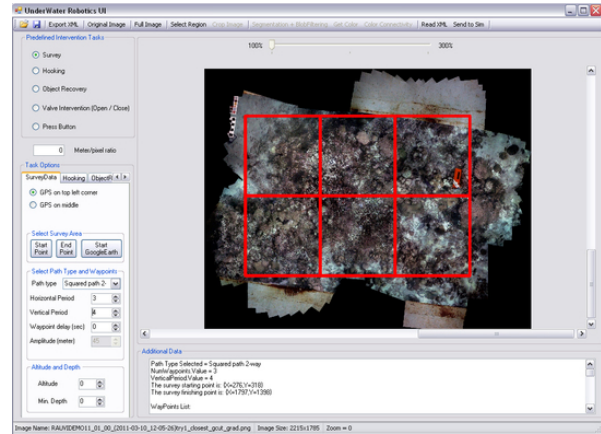
For the experiments of this paper, \mathbf{v}_E is computed proportional to the error between the current end-effector pose and the desired one, i.e. the hand moves in a straight line towards the object.

4 The User Interface

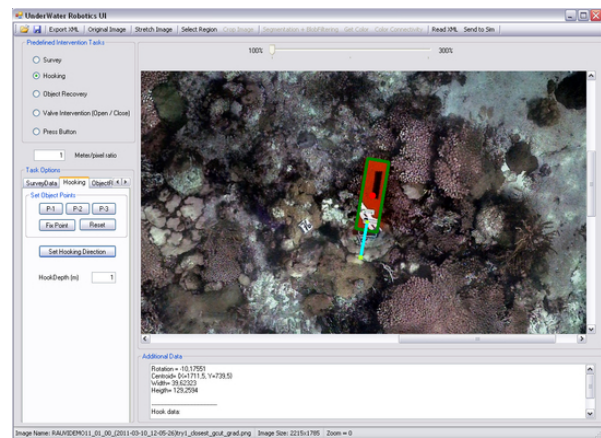
The RAUVI project proposes a two-stage strategy[3]: during the first stage, the I-AUV is programmed at the surface and receives a plan for surveying a given Region of Interest (RoI). During the survey it collects data from cameras and other sensors. At the end of this first stage, the I-AUV returns to the surface (or to an underwater docking station) where the data is retrieved and an image mosaic of the seabed is reconstructed [13]. The Target of Interest (ToI) is then identified on the mosaic and the intervention action is specified by means of a user interface described later in this section. Then, during the second stage, the I-AUV navigates again to the RoI, localizes the target and executes the intervention mission in an autonomous manner.

The Graphical User Interface (GUI) is used to specify both, the survey path and the intervention task. The former is done by loading a geo-referenced map of the area and indicating a set of waypoints (possibly using predefined grid-shaped trajectories). The waypoints are sent to the vehicle control system that guides the robot through them. Figure 7a shows an example of a grid-shaped trajectory superposed on a generated mosaic obtained during the experiments described later in this paper. Once the mosaic has been built, the user first looks for the target of interest on it. After selecting the target, the intervention task is indicated by choosing between different pre-programmed actions such as grasping, hooking, etc.

The user interface contains built-in image processing and grasp planning algorithms that automate the task specification process when possible. If automatic



(a) Survey. A grid-shaped trajectory is specified, here shown superposed on the generated mosaic.



(b) Intervention. The object is enclosed in a bounding box, and the task parameters are set.

Fig. 7 Mission specification in the GUI

methods fail, the user can always specify the task parameters manually. For the experiments described in this paper, a hooking task is considered, which is defined by enclosing the target of interest in a bounding box, and selecting the point and the direction where to attach the hook, as shown in Figure 7b. It is worth mentioning that the black box recovery is just one specific mission that can be performed under the RAUVI two-stage strategy. However, other different missions could be defined under the same umbrella.

When the specification is finished, an XML file containing the task parameters is generated. For the hooking task, this file includes:

- The image used for the specification. It is assumed that this image is geo-referenced so that it is possible to relate pixel coordinates to meters with respect to a global frame.
- The ToI bounding box origin with respect to the image origin, represented as (x, y, α) . (x, y) are pixel

coordinates and α is the orientation of the bounding box with respect to the horizontal.

- The width and height of the bounding box, both in pixels and in metric units, due to the fact that the image is geo-referenced and the camera intrinsic parameters are known, thus allowing to compute 3D dimensions from single frames.
- A hook point and direction given in pixel coordinates with respect to the bounding box origin, and also in metric units.

With the bounding box information, a template containing only the ToI is created and later used for object detection and tracking (see Sections 6 and 7.2).

5 3D Simulation & Visualization Tool

A 3D visualization environment (UWSim) has also been developed and used for two purposes: simulation of the mission before running it on the real robot, and visualization of the actual execution by reading real sensor signals. UWSim is being developed for the project, but makes use of the publicly available open source *Open-SceneGraph* and *osgOcean* libraries that allow to visualize underwater effects like silt, light attenuation, water distortion, etc. More concretely, UWSim includes:

- The I-AUV 3D kinematic model, including both the vehicle (GIRONA 500) and the arm (either the 5 DOF Light-Weight ARM 5 E, or a 7 DOF arm). Arm kinematics have been implemented, thus allowing to move the arm joints.
- A virtual camera attached to the front-bottom of the vehicle and facing downwards. Another virtual camera has been attached to the wrist of the arm. These cameras capture images of the seabed in real-time.
- A light source, placed at the bottom part of the vehicle and pointing towards the floor.
- A model of the seabed including a texture. In this particular case the model of the CIRS water tank at Girona University was used.
- A flight data recorder, lying on the seabed, which is the object that the I-AUV has to recover.

The simulation environment facilitates both the testing of the control algorithms before running them on the actual robot, and the visualization of the actual execution. The virtual sensors and actuators are interfaced through ROS topics [7]. This allows seamless integration of this tool with the rest of the architecture, thus providing realistic playback of for simulation purposes, or updating the simulated actuators with real odometric information. Figure 8 shows the visualization envi-

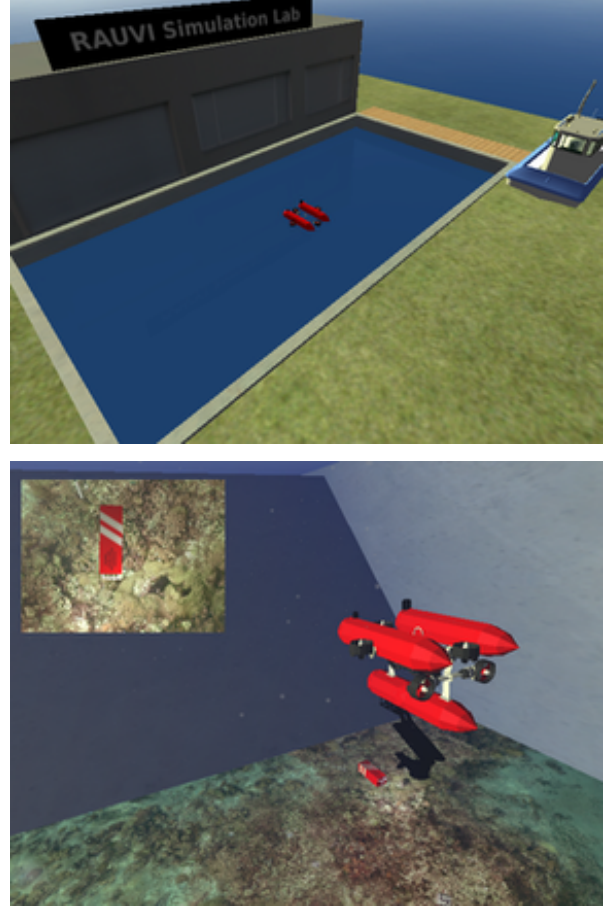


Fig. 8 Top: a virtual model of the CIRS water tank. Bottom: virtual visualization of the real execution

ronment as it reproduces in real time the actual robot motion during the experiments described in Section 7.

6 Visual perception aspects

Light propagating in water is subject to a variety of physical phenomena that affect the image formation [14]. Absorption and scattering dramatically reduce the effective distance of underwater vision and the contrast of the images formed under these conditions. Moreover, flora and fauna present in the scene produce variable and irregular shapes and shadows that can often hide the original appearance of objects lying on the seabed. Thus, a suitable underwater vision system has to take into account the media it works in, the nature of the images it deals with, as well as the application it is designed for.

Different solutions have been proposed concerning the configuration of lighting and gathering equipment for vision systems that are specifically designed to operate in subsea conditions. After a wide revision of the systems described in the literature and technical do-

cuments [15], a new solution has been adopted that is based on two stereo rigs. One of the stereo cameras faces forward and the other faces downward. Depending on the mission to be carried out or on the requirements of the current mission stage, each camera will be used for a different function, some of which are described below. For example, in the experiments described here the scene was flat and distance to the scene approximately constant. 3D perception was not essential, and a monocular configuration with a single downward looking camera provided satisfactory results at a significantly reduced cost of resources.

6.1 Vision System Tasks

Visual information is useful for a wide array of tasks during an AUV mission. As described in [3], the RAUVI project splits a mission into two stages: survey and intervention. In what follows, the different vision tasks that are executed during these stages are described.

During the survey stage and whenever the seabed is visible, camera images are saved on disk and tagged with a time stamp and a first order approximation of the current robot position. A set of distinct visual features is then extracted from the image and saved separately in a database. Motion estimates can be obtained by tracking sets of features over consecutive camera images. Such estimates include a reliable evaluation of the measurement error. Visual motion estimates depend on visibility and can be disrupted when not enough features can be tracked, but they are virtually drift free and can be highly accurate, forming an ideal complement to other onboard navigation sensors. Whenever the survey trajectory has points of overlap or intersection, the vision system can also accurately estimate the pose with regard to previous images from such locations. This allows the navigation unit to take corrective action if the intended point of intersection is not met. Once the survey stage finishes, the vehicle surfaces and uploads the gathered information so that the intervention stage can be specified. This specification uses a mosaic to provide a large area view of the sea floor.

The mosaic building process is detailed in [16,17]. It starts by searching for correspondences between consecutive images (referred to as consecutive image registration) to determine their homographies. By cascading these homographies it becomes possible to predict non-consecutive overlapping images and attempt to register such images. All successfully registered image pairs impose geometric constraints on the spatial arrangement of the images. Typically, as there are more image matches than images, the problem of finding the image

locations is over-constrained. A global optimization process, based on a non-linear least squares algorithm is then used to find a best fit solution to the location of all images. As a final step in the mosaic creation process, a seamless composite image is created by suitably blending the registered images [18,19]. For the intervention, a human operator selects the ToI from the mosaic, as shown in Figure 7b.

During the intervention stage, the AUV uses the image and navigation data that were obtained during the survey stage to guide the vehicle to the target. When the target area is identified, the vehicle will start to maneuver on a finer scale and a number of image analysis techniques can be applied to help the vehicle to locate the target, keep station over it, and to help the robotic arm to manipulate it. While the identification of the target area, the localization of the target itself, and the keeping of station over it can be considered to pose the same problem at different scales, they are solved with different methods. Station keeping relies on the extraction and matching of local features.

Depending on the mission, the vision system allows the target identification based on colour, texture or features, among other characteristics. Due to the colour saliency of the FDR in the images (see Figure 9, left), in the present experiment the target was identified by histograms of hue and saturation in the HSV colour space. As the scene is assumed to be static, a histogram of background colours is also used to filter the target colour histogram, reducing the number of false positives. This process results in a target model formed by the histogram containing only those colours that are significant for the target in the current scene. This information, together with the size in pixels of the target is stored and used to detect the target during the search stage.

Because the target can move during manipulation, and because the frequency and accuracy at which the navigation system and the robotic arm require updates on robot pose and target pose respectively, target localization and station keeping have to be treated as independent tasks that are optimized by different implementation choices. To help the navigation unit to correct for drift and keep the vehicle stationary, motion with regard to an arbitrary but constant reference frame at the target location is reported. To assist the robotic arm, the exact location of the target within the current view is provided.

6.2 The Vision Module Architecture

The vision module must provide the rest of the system with higher-level processing capabilities as described



Fig. 9 *Left* image from the survey stage on which the model of the target is based. *Center* model of the target based on the bidimensional HS histogram. *Right* detection, pose and size estimation of the target

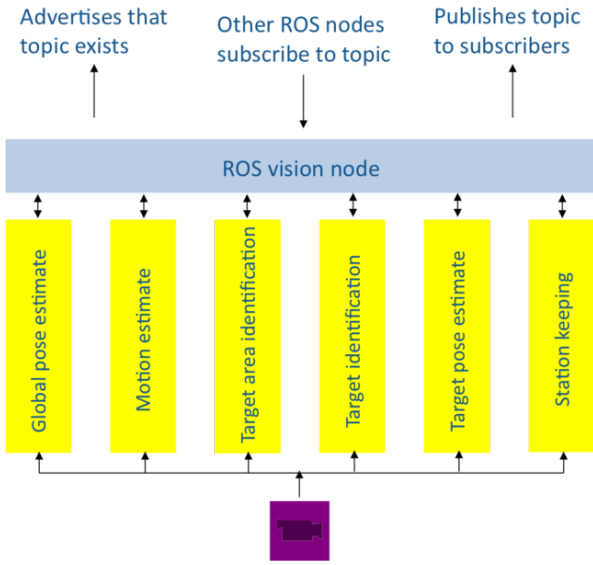


Fig. 10 Vision module architecture as a ROS node

above. To that end, this module is conceived as a ROS node on independent processing hardware and that advertises a number of topics [7] to which other ROS nodes can subscribe when needed (see Figure 10). For the planar sea floor of this experiment, a monocular two-dimensional setup is used. The visual odometer estimates robot motion and pose from image features that can be well localized and that are relatively invariant to contrast, scale, and view point [20–22]. The type of features that gives best results depends on the type of scene and can be adapted during a mission, though we find that in natural environments blobs—round areas with high contrast against the background—give more reliable estimates than edges and lines. In particular, the SURF feature descriptor [23,24] offers the best combination of speed, invariance, and configurability. SURF features allow us to calculate motion between consecutive images, identify overlap at points where the survey trajectory intersects, and to detect and localize the ToI. Images are processed only once and the ex-

tracted features are stored for reference. When stereo images are used, the distance of each feature to the camera is immediately calculated and stored as well. All further operations are performed on the extracted features. The feature descriptors of a single image typically occupy in the order of 100kB of memory, and the visual system adopts a variety of heuristics to load only those features into main memory that have a high probability to match against the next image.

For each feature, a descriptor is calculated from the two-dimensional Haar wavelet response in a number of rectangular regions that surround the feature. A match with a feature in another image or in the ToI is confirmed if the Euclidean distance between responses is below a certain threshold, and is also significantly lower than to any other features in the same image. Motion between consecutive images, as well as pose estimates with regard to intersections of the survey trajectory, with regard to an arbitrary frame during station keeping, and with regard to the ToI are all estimated from the affine homography calculated between co-planar sets of matching features.

Affine homographies with only four degrees of freedom (lateral translation, yaw, and scale) are used. Despite the fact that the vehicle cannot completely prevent pitch and roll, inclusion of these additional degrees of freedoms in the calculation of the homography leads to a decrease in accuracy, in particular when motion estimates are calculated over long series of images. RANSAC (RANdom SAmple Consensus [25]) is extensively used, both to filter out the large number of mismatches between features, as well as to prevent poorly localized features from influencing the pose estimate.

The accuracy of our visual motion estimates was evaluated by comparing the affine homographies between 10,000 pairs of images to what can be established by matching images to the original poster image. Overlap between images pairs was between two third of the image area to 97%. While the homographies between pairs of images were calculated in real time from

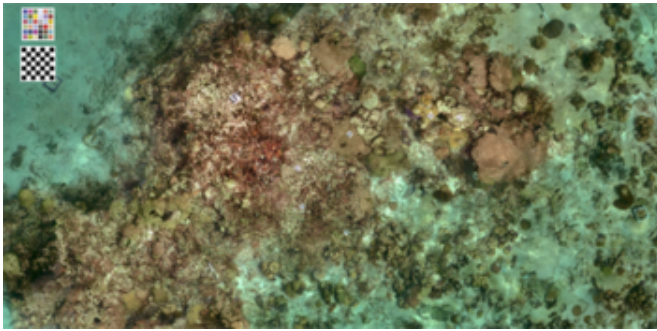


Fig. 11 The original sea floor image placed at the bottom of the CIRS water tank

between 6 and 40 matching pairs of features per pair of images, the homographies between images and the image poster were calculated offline, made use of several extensive but slow search methods, and are typically based on 60 matching pairs of features. The error in translation, yaw, and scale all follow a normal distribution around zero and are all correlated with a correlation coefficient of about 0.5. The variance of the error in x and y direction of the camera frame is 5 millimeter at a distance of one meter from the floor. The variance of the error in yaw is 0.001 degree. The variance of the error in scale is 0.00004, a value that would seem unlikely low if not for the fact that scale is almost constant one with variance 0.0002.

7 Experimental validation: the Search & Recovery problem

To experimentally validate the system described above, a real Search & Recovery problem is considered: finding and retrieving a flight data recorder. The experiments were carried out at the CIRS water tank (University of Girona). A digital image of a real sea floor (see Figure 11) was printed in a 4×8 m poster and placed at the bottom of the water tank, as can be appreciated in Figure 1. A mockup of a black box (of size $13 \times 15 \times 40$ cm) was placed at an unknown position at the floor of the water tank. The experiment was divided into two stages: a survey stage where the robot had to build a photo-mosaic of the ground, and an intervention stage, where the FDR was actually recovered.

7.1 Survey

In order to properly cover the search area, the robot was programmed to survey the bottom of the water tank along a grid shaped trajectory with one meter distance between parallel swaths. At the commanded altitude of

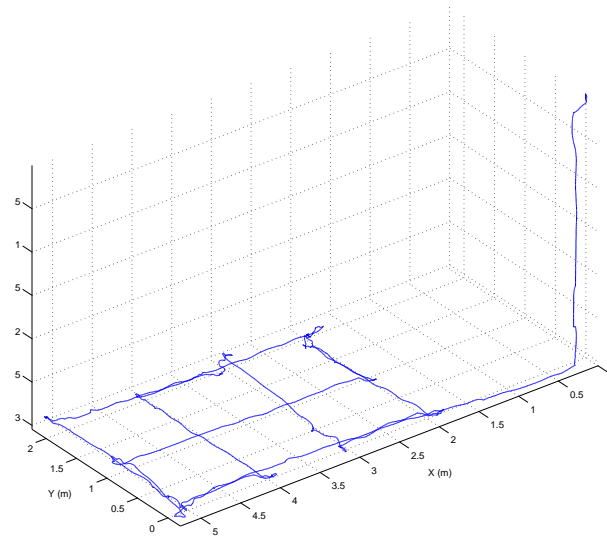


Fig. 12 The trajectory followed by the vehicle during the survey

one meter, this grid resolution ensures full camera coverage of the explored area and avoids gaps in the final mosaic. To perform the trajectory, the vehicle started at a known position at the border of the water tank and navigated through the area using the dead-reckoning estimate from the on-board Kalman filter, which merges the information from the DVL (a 600kHz Teledyne-RDI explorer PA), the pressure sensor (a Valeport miniSVS) and the fiber optics gyro enhanced AHRS (a Tritech iGC combined with a Tritech iFG) [26]. The resulting estimated trajectory can be seen in Figure 12.

Once the navigation data and the acquired images have been retrieved, a simple preliminary mosaic can be built by projecting the images using the measured vehicle position and altitude over the floor. Both the images and the navigation data have consistent time stamps which makes possible combining them. That preliminary mosaic allows to rapidly explore the visual map in search for the object to recover. Alternatively, the complete mosaic can be built, to provide better image quality and higher precision. However, due to the computational complexity involved, it requires an additional processing time of a few hours. Figure 13 shows the resulting mosaic for the water tank experiment in which the position of the FDR (at the top-right side of the image) can be determined prior to the intervention. This mosaic can be compared with the original image shown in Figure 11.

An important advantage of the experimental pool setup of this paper, is that the texture in the bottom is known a priori. By performing direct image to poster image registration, it becomes possible to estimate the pose of the vehicle with significantly higher ac-

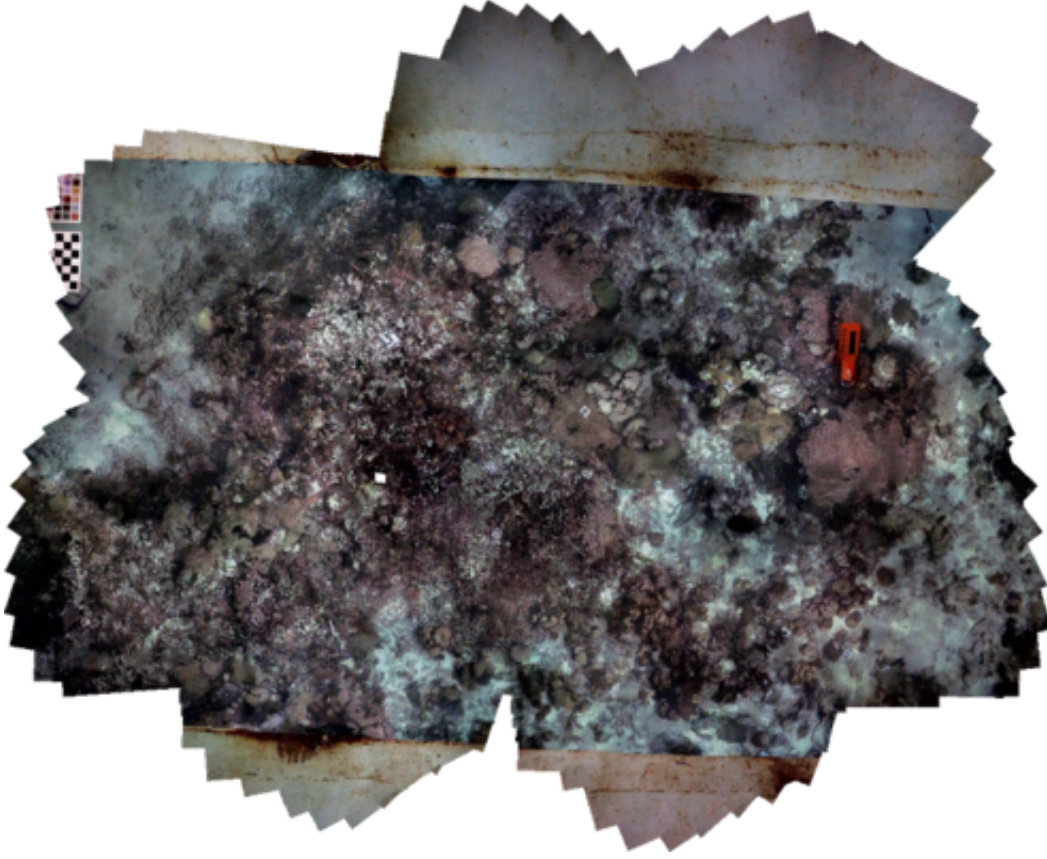


Fig. 13 The mosaic generated after the survey. Compare with Figure 11. The black box can be appreciated on the top-right corner

curacy than using acoustic sensing methods. Although not explored in this paper, such estimate can be used as ground truth for benchmarking other localization modalities. An example of mosaic based pose estimation is given in Figure 14, using the maximum-likelihood method of [27]. A first order approximation is used to propagate the covariance from the correspondences to the pose.

7.2 Intervention

For the intervention stage the robot was relaunched and it guided itself autonomously to the pre-programmed position where the black box was found (see Figure 15). There, the robot kept its position and attitude with visual feedback from the target object. While keeping station, the arm was able to autonomously retrieve the object in different trials.

Vision-based station keeping was performed with two degrees of freedom: the horizontal motion of the vehicle was controlled in order to keep the tracked template origin close to a desired position in the current view. Vertical motion was controlled with the altimeter feedback in order to keep a suitable distance to the

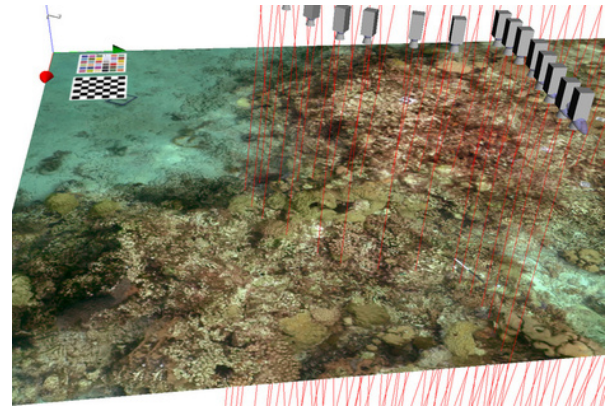


Fig. 14 3D representation of part of the AUV trajectory during survey, obtained by direct image to poster image registration. The blue ellipsoids (at the base of the camera icons) represent the uncertainty volumes on the vehicle location at 95% probability. To allow better visualization, the ellipsoid axis were enlarged by a factor of five

floor of around one meter, measured from the base of the arm. Figure 16 shows the evolution of the error in image pixels between the actual and the desired object position. The system for vision-based station keeping was active during the entire manipulation action. Note

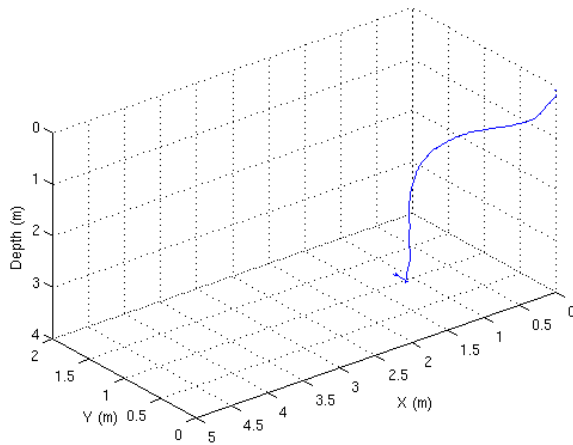


Fig. 15 The trajectory followed by the vehicle for the intervention. The small displacement and the end of the trajectory is due to the visual station keeping

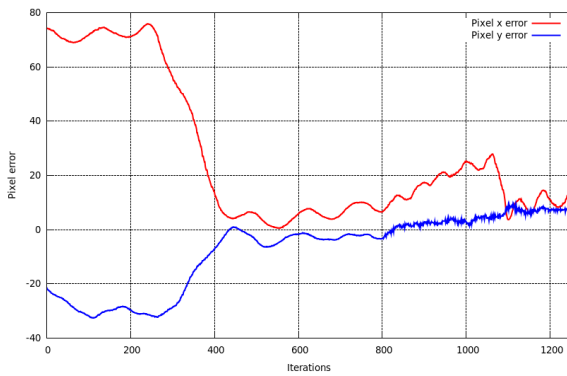


Fig. 16 Error in pixels between actual object position in the image and desired object position

the quickly decreasing error in object position, from an initial state that was far from desired, to virtually zero at measurement iteration 1100. The disturbances towards the very end of the sequence may be due to the dynamic effects of arm motion on vehicle position. It is expected to improve these results by generating smoother arm trajectories with very low accelerations.

Figures 17 and 18 show how the hook is successfully attached to the FDR. A template tracking algorithm was in charge of following the object motion in the image and computing its 3D pose that was later used to perform Resolved Motion Rate Control (RMRC) of the manipulator as described in Section 3.2 and detailed in [12][28].

The mission finished with the retrieval of the FDR. The effects of the vehicle mass change were compensated by the PID controllers in charge of the depth and pitch degrees of freedom. However, it is worth mentioning that the mass of the mock-up FDR is small and that the capacity of the vehicle to lift heavier objects is yet to be studied.

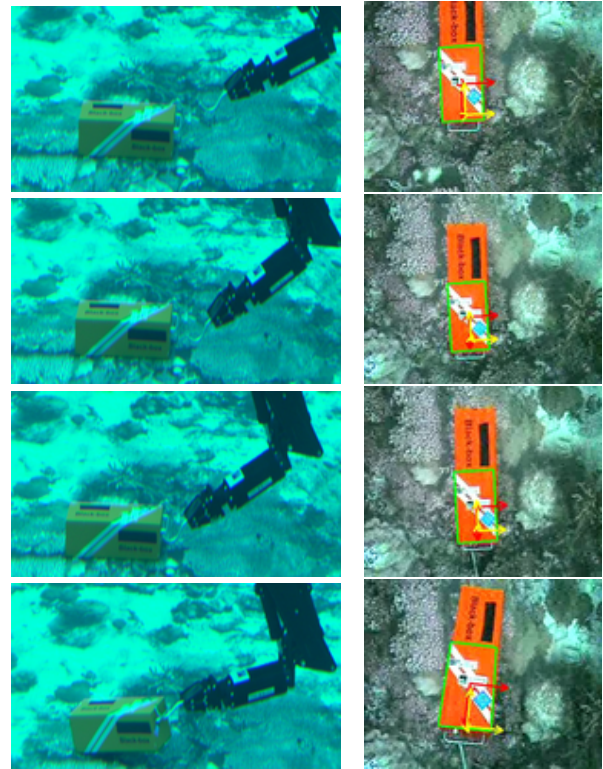


Fig. 17 *Left column* the underwater arm autonomously attaching the hook. *Right column* object tracking from the vehicle camera

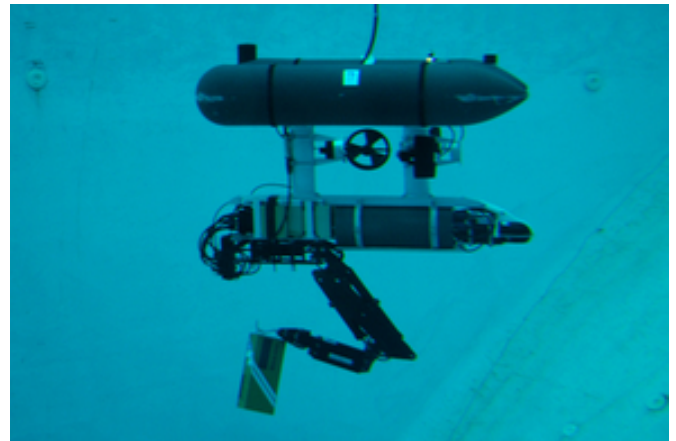


Fig. 18 The vehicle returns to the surface with the successfully retrieved black box

8 Conclusions & Future work

The most recent progress of the RAUVI project has been presented. An Autonomous Underwater Vehicle for Intervention (I-AUV) has been developed and was successfully tested under the relatively realistic conditions that can be created in a water tank. A Search & Recovery task has been considered for the experimental validation. Specifically, the capability to autonomously search for a flight data recorder and to re-

trieve it by means of an underwater robotic arm has been demonstrated. To this end, the underwater vehicle first surveyed the seabed where it collected images and odometric information. The collected data was used to build a photo-mosaic, which was then loaded into a GUI where the target object was localized and the retrieval task was specified. Next, the I-AUV autonomously navigated to a position on top of the target object, and kept station with visual feedback. Meanwhile, the target pose was computed in real time, and used to control the manipulator, which recovered the flight data recorder. Notably, this experiment has demonstrated the feasibility and reliability of the RAUVI project, which envisioned the coordinated effort of many different resources from both the human and mechatronics (hardware and software) point of view.

For future work it is expected to improve the vision-based station keeping by implementing a full image-based visual approach that allows visual control of all degrees of freedom of the vehicle. Another task to be addressed is to fully integrate the visual odometry with the inertial and acoustic systems to improve the robot localization. Regarding manipulation, further improvements can be made by generating smooth velocity and acceleration trajectories, and by implementing error recovery actions when the manipulation action fails. It is also planned to integrate the GUI and the 3D Simulator into a single software package, and to apply augmented reality techniques in order to improve the interaction with the user and to assist with the specification and supervision of the intervention mission. Further work on using acoustic modems to rapidly localize the black box will be also addressed. Finally, these promising results encourage us to follow with the next step: a shallow water test of RAUVI by the end of 2011.

Acknowledgements This research was partly supported by Spanish Ministry of Research and Innovation DPI2008-06548-C01-02-03 (RAUVI Project), by the European Commissions Seventh Framework Programme FP7/2007-2013 under grant agreement 248497 (TRIDENT Project), by Fundació Caixa Castelló-Bancaixa P1-1B2009-50, and by the Spanish Ministry of Education and Science (MCINN grant CTM2010-15216), Ramon Y Cajal Program (N. Gracias) and the FPI program (R. Campos).

References

1. Marani G, Choi SK (2010). Underwater target localization. *IEEE Robotics & Automation Magazine* 17(1):18
2. Evans J, Redmond P, Plakas C, Hamilton K, Lane D (2003). Autonomous docking for intervention-auvs using sonar and video-based real-time 3d pose estimation. In: *OCEANS 2003*, vol. 4, pp. 2201–2210
3. Sanz PJ, Prats M, Ridao P, Ribas D, Oliver G, Orti A (2010). Recent progress in the RAUVI project. a reconfigurable autonomous underwater vehicle for intervention. In: *52-th International Symposium ELMAR-2010*, pp. 471–474. Zadar, Croatia
4. Weiss P, Mascarell J, Badica M, Labbe D, Brignone L, Lapierre L (2003). Freesub: Modular control system for intervention auvs (iauv). In: *13th International Symposium on Unmanned Untethered Submersible Technology (UUST03)*. Durham, New Hampshire, USA
5. Denavit J, Hartenberg RS (1955). A kinematic notation for lower-pair mechanisms based on matrices. *Trans ASME J Appl Mech* 23:215–221
6. Quigley M, Gerkey B, Conley K, Faust J, Foote T, Leibs J, Berger E, Wheeler R, Ng A (2009). ROS: an open-source Robot Operating System. In: *ICRA Workshop on Open Source Software*
7. Cousins S (2010). Welcome to ROS topics. *IEEE Robotics & Automation Magazine* 17(1):13–14
8. Palomeras N, García JC, Prats M, Fernández JJ, Sanz PJ, Ridao P (2010). A distributed architecture for enabling autonomous underwater intervention missions. In: *Systems Conference, 2010 4th Annual IEEE*, pp. 159–164
9. Ridao P, Ribas D, Hernández E, Rusu A (2011). USBL/DVL navigation through delayed position fixes. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2344–2349. Shanghai, China
10. Maybeck P (1982). *Stochastic models, estimation and control*, vol. 1. Academic Press
11. Healey AJ (2006). Guidance laws, obstacle avoidance and artificial potential functions. In: *Geoff Roberts RS, ed., Advances in Unmanned Marine Vehicles*, vol. 3, pp. 43–66. The Institution of Electrical Engineers
12. Prats M, García JC, Fernández JJ, Marín R, Sanz PJ (2011). Towards specification, planning and sensor-based control of autonomous underwater intervention. In: *IFAC 2011*. Milano, Italy. To be published
13. Nicosevici T, Gracias N, Negahdaripour S, Garcia R (2009). Efficient three-dimensional scene modeling and mosaicing. *J Field Robot* 26:759–788
14. Mobley CD (1994). *Light and water, radiative transfer in natural waters*. Academic Press
15. Bonin F, Burguera A, Oliver G (2011). Imaging systems for advanced underwater vehicles. *Journal of Maritime Research* To be published
16. Ferrer J, Elibol A, Delaunoy O, Gracias N, García R (2007). Large-area photo-mosaics using global alignment and navigation data. In: *Oceans MTS/IEEE*, pp. 1–9. Vancouver, Canada
17. Lirman D, Gracias N, Gintert B, Gleason A, Reid RP, Negahdaripour S, Kramer P (2007). Development and application of a video-mosaic survey technology to document the status of coral reef communities. *Environmental Monitoring and Assessment* 1–3(125):59–73
18. Gracias N, Gleason A, Negahdaripour S, Mahoor M (2009). Fast image blending using watersheds and graph cuts. *Image and Vision Computing* 27(5):597–607
19. Prados R, Neumann L, Cufi X, Garcia R (2007). Visually pleasant blending techniques in underwater mosaicing. *Instrumentation Viewpoint* 6
20. Kosecka J, Li F, Yang X (2005). Global localization and relative positioning based on scale-invariant keypoints. *Robotics and Autonomous Systems* 52(1):27–38. *Advances in Robot Vision*
21. Kosecka J, Li F (2004). Vision based topological Markov localization. In: *Proceedings of 2004 IEEE International Conference on Robotics and Automation, ICRA'04*, vol. 2, pp. 1481–1486

-
22. Lowe DG (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60:91–110
 23. Bay H, Tuytelaars T, van Gool L (2008). Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding* 110:346–359
 24. Evans C (2009). Notes on the OpenSURF library. Tech. Rep. CSTR-09-001, University of Bristol
 25. Fischler MA, Bolles RC (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun ACM* 24:381–395
 26. Ribas D, Ridao P, Neira J (2010). Underwater SLAM for Structured Environments Using an Imaging Sonar. No. 65 in *Springer Tracts in Advanced Robotics*. Springer Verlag, Heidelberg, Germany
 27. Gracias N, Zwaan S, Bernardino A, Santos-Victor J (2003). Mosaic based Navigation for Autonomous Underwater Vehicles. *Journal of Oceanic Engineering* 28(4)
 28. Whitney DE (1969). Resolved motion rate control of manipulators and human prostheses. *IEEE Transactions on Man Machine Systems* 10(2):47–53