A Real-time Underwater Object Detection Algorithm for Multi-beam Forward Looking Sonar ⋆

Enric Galceran * **Vladimir Djapic** ** **Marc Carreras** *
David P. Williams **

* *University of Girona, Underwater Robotics Research Centre, Pic de Peguera s/n, 17071 Girona, Spain (e-mail: enricgalceran@eia.udg.edu, marc.carreras@udg.edu).*
** *NATO Undersea Research Centre, Viale San Bartolomeo 400, 19126 La Spezia (SP), Italy (e-mail: djapic@nurc.nato.int, williams@nurc.nato.int)*

Abstract: A novel algorithm for the detection of underwater man-made objects in forward-looking sonar imagery is proposed. The algorithm takes advantage of the integral-image representation to quickly compute features, and progressively reduces the computational load by working on smaller portions of the image along the detection process phases. By adhering to the proposed scheme, real-time detection on sonar data onboard an autonomous vehicle is made possible. The proposed method does not require training data, as it dynamically takes into account environmental characteristics of the sensed sonar data. The proposed approach has been implemented and integrated into the software system of the Gemellina autonomous surface vehicle, and is able to run in real time. The validity of the proposed approach is demonstrated on real experiments carried out at sea with the Gemellina autonomous surface vehicle.

Keywords: Detection algorithms, forward-looking sonar, underwater object detection, acoustic image processing, image processing.

1. INTRODUCTION

Sonar can provide imaging of underwater environments even in low and zero visibility conditions. This capability has proven particularly useful for the detection of underwater man-made objects. With recent advances in maritime technology, the sonar data used to address this task can be collected by an Autonomous Surface Vehicle (ASV) or an Autonomous Underwater Vehicle (AUV) with no human intervention. Detection and intervention on man-made objects are costly tasks and imply inherent danger and time constraints. Thus, to fit the autonomous vehicle with intelligence so that it can immediately react to the data it collects is a priority. Before this goal can be realized, however, an algorithm is needed that can perform object detection in real-time onboard.

Basically, two sorts of sonar technology are suitable for automatic man-made object detection. On one hand, long range, high resolution imagery provided by side-scan sonar (SSS) and especially by synthetic aperture sonar (SAS) allows for performing detection in vast survey areas (i.e. hundreds of meters long survey tracks). On the other hand, multi-beam forward-looking sonar (FLS) allows for a closer, more in-detail inspection of possible man-made object locations. A typical approach to the problem consists in first detecting possible object locations in SSS or SAS imagery, and then performing reacquisition of these locations by means of FLS to assess that there are in fact objects of interest in such locations, and perhaps carrying out some intervention task on them.

Hayes and Gough (1992) showed that the high-resolution imagery provided by SAS is suitable for the detection of man-made objects on the seabed. In the past few years, several man-made object detection methods have been proposed for SSS and SAS imagery (Dobeck et al., 1997; Reed et al., 2003; Fawcett et al., 2006; Maussang et al., 2007; Groen et al., 2009; Williams and Groen, 2011). These methods are suitable for detecting possible locations of man-made objects laying on the seafloor of vast surveyed areas. Nonetheless, after this large-scale detection one might need to reacquire the detected targets for a closer and more in-detail inspection task, for which FLS is a more suitable option than SSS or SAS. Therefore, there is a need to perform automatic detection also with FLS in order to conduct fully autonomous detection missions with an ASV or an AUV.

On the other hand, some generic obstacle detection methods with FLS have been proposed. Lu and Sang (1998) make use of image processing techniques and the near field acoustic scattering principles of underwater targets to estimate the two-dimensional position and size/shape of nearby targets. Guo et al. (1998) use the continuous image sequences generated by an electronic scanning forward-looking sonar to achieve the aim of obstacle avoidance

⋆ This research was sponsored by the Spanish government (DPI2011-27977-C03-02) and the TRIDENT EU FP7-Project under the Grant agreement No: ICT-248497.

and visual navigation for an AUV. Here, they use a track-before-detect strategy to extract information contained in image sequences to estimate the dynamics of the AUV, then they apply a dynamic programming algorithm to solve the problem of detection. This method aims to reduce the computational cost to meet the real-time demand on obstacle avoidance and navigation of an AUV system.

However, the critical demand for real-time signal processing and the uncertainties of the AUV's dynamics make online detection of obstacles a challenging task. Martin et al. (2000) propose an obstacle detection method where the region forward of the FLS is sub-divided into various cells. The cells are filled with the raw intensity data collected from the FLS sensor. For each filled cell, a cell signature is computed. The maximum signature cell is extracted from the grid. This cell contains transformed target information such as range, bearing to target, and cell signature.

Petillot et al. (2001) describe a framework for segmentation of sonar images, tracking of underwater objects and motion estimation. The real-time data flow (acoustic images) acquired with multi-beam FLS is first segmented and relevant features are extracted. This framework is applied to the design of an obstacle avoidance and path planning system for underwater vehicles.

However, little attention has been paid to the specific problem of man-made object detection with FLS. Moreover, the aforementioned methods suffer from several limitations that preclude them from being a suitable option for seabed man-made object detection. In particular:

- Most detection algorithms rely on training data from a different area and hence do not take into account the particular environmental conditions where the surveying takes place. Thus, such methods are not able to dynamically deal with different seabed compositions in different regions.
- Generic obstacle detection algorithms do not consider domain-specific geometrical and physical knowledge about the man-made object to be detected.

In this work, aiming to address and overcome these limitations, we propose a novel object detection algorithm for reacquisition of man-made objects on the seabed using multi-beam FLS. To achieve this objective, we borrow ideas from the detection algorithm for SAS imagery presented in (Williams and Groen, 2011). Our proposed method explicitly takes into account environmental characteristics and other geometrical knowledge about the problem domain. To the best of the authors' knowledge, this is a pioneering approach to automatic real-time man-made object detection using FLS.

The remaining sections of this paper are organized as follows. The proposed detection algorithm is described in Sec. 2. Experimental results on real autonomous missions conducted at sea with the Gemellina ASV are presented in Sec. 3. Concluding remarks and directions for future work are noted in Sec. 4.

2. PROPOSED DETECTION ALGORITHM

We propose a detection algorithm whose overall objective is to detect underwater man-made objects of interest on FLS imagery. The algorithm needs to be made fast as it is intended to run in real-time onboard an autonomous vehicle with (possibly) limited computational resources. Since no human intervention is allowed, the algorithm is designed to deal with changing environmental conditions that have a direct effect on the collected sonar data. At the same time, the proposed method addresses the limitations stated in the previous section.

The main idea of the algorithm is to locate echo highlights that are locally higher than their background. As in the object detection algorithm intended for SAS imagery presented by Williams and Groen (2011), the successive phases of the algorithm are concatenated in such a way that computational costs are minimized by operating on smaller portions of the image at each phase. Also as in the aforementioned work, the integral-image representation is used to speed up the algorithm. We take advantage of the *a priori* knowledge about the object (i.e., shape, size, etc.) by applying filtering steps in order to consider only highlights corresponding to an actual object of interest. The following sections will describe each step of the detection algorithm that resulted from these considerations. Each step will be demonstrated on an example FLS image (see Fig. 1).

2.1 Region of Interest of the Sonar Image

The proposed algorithm is designed to work on a rectangular region of interest of the sonar images. This design criteria is based on two reasons. First, it allows avoidance of noisy or poor-quality areas of the sonar image produced by some sonars at certain ranges. Second, it aids speeding up the detection by not processing the entire image, but only a subregion of it. Hence, from the very beginning, every step discussed hereafter will be applied only inside the predetermined region of interest.

The rectangular region of interest is specified by the position of its top-left corner in a Cartesian coordinate system with origin at the sonar head. This coordinate system will be used by all the subsequent phases of the algorithm. In our particular case, using a maximum range of 25 m the top-left corner (x_r, y_r) of the rectangle was located at $x_r = -11.5$ m, $y_r = 23$ m from the sonar head, with a width, $w_r = 23$ m and a height, $h_r = 11$ m (see Fig. 1(a)).

Naturally, the region of interest can be enlarged to the entirety of the image if the particular sonar images used are of good quality all along the sonar range and if the computational resources of the vehicle allow for real-time processing of entire images.

2.2 Integral Image

An integral image (Viola and Jones, 2004), also called a summed area table, is a representation of an image that allows for fast computation of the sum of pixel values in a given rectangular area of the image. In subsequent stages of the algorithm, we will exploit this image representation for quickly assessing certain distinguishing characteristics of objects, such as background and echo levels. The fast calculations allowed by the use of the integral-image representation also make real-time detection possible.

Thus, rather than operating on the pixel-based system of the sonar image, we immediately transform to an integral-image system, which contains equivalent information.

The corresponding integral image, I, of an original sonar image, A, is constructed as follows. The value at a location (x, y) in the integral image corresponds to the sum of pixel values above and to the left of (x, y), inclusive, in the original image, A. That is,

$$I(x,y) = \sum_{x' \leq x, y' \leq y} A(x', y'). \tag{1}$$

The integral image is quickly generated by applying the following recursive relation:

$$I(x,y) = I(x-1,y) + z(x,y), \tag{2}$$

where $z(x, y)$ is the cumulative sum of pixels in a row of the original image,

$$z(x,y) = z(x,y-1) + A(x,y). \tag{3}$$

As can be noticed, the integral image is computed in only one pass over the original image. Using the integral image, the sum of pixel values in any arbitrary rectangle in the image can be computed with only four array references to the integral image (without the need for referencing all the involved pixels in the original image). We will take advantage of this fact in our algorithm.

The integral-image representation corresponding to the region of interest of the sonar image in Fig. 1(a) is shown in Fig. 1(b) overlapped on the original image.

It is worth noticing that once the integral image is calulated, the algorithm uses this representation in the subsequent stages rather than the original image.

2.3 Background Estimation

The first use of the integral image, I, is in the estimation of the sonar-image background map, B. The purpose of the background map is to establish the seabed reverberation level. Once established, the reverberation level will be used subsequently to determine locations of echo highlights in the image.

The reverberation level strongly depends on the seabed composition. Thus, using a predefined threshold for all possible seabed types to determine which pixels correspond to the seabed is not a reliable option. For example, the reverberation level of a soft muddy bottom will be lower than that of a bottom of hard-packed sand (Williams and Groen, 2011).

For this reason, an approach that defines the background level according to some global intensity average over the sonar image could fail catastrophically. Rather than using some global threshold, we argue that the background estimation should be performed locally in the image.

We do this local estimation by using two different-sized, concentric sliding windows: a bigger, outer window with an inner, smaller window laying inside the bigger one. For each pixel in the region of interest of the sonar image, we compute the mean pixel value of the neighbor pixels laying in the bigger window, but we ignore the pixels laying in the inner window. Pixels in the inner window are ignored because, if an object were present, they would correspond to a high-intensity echo return related to the object.

Window sizes can be adjusted according to the object we want to search for. In our particular case, the bigger, outer window has width, $b_{ox} = 4$ m in the sonar's X axis and height, $b_{oy} = 4$ m in the sonar's Y axis; the smaller, inner window has width, $b_{ix} = 1$ m and height, $b_{iy} = 1$ m. The background score at location (x, y), $B(x, y)$, is then the mean pixel value in the bigger window centered around (x, y), ignoring pixels laying inside the smaller window.

The calculation of the two rectangles involved in the background value at a given location can be computed quickly using the integral-image representation. A total of only eight array references to the integral image are necessary to compute the values for the two rectangles (four references per rectangle). Specifically, the background score at pixel location (x, y) is calculated as

$$\begin{aligned} B(x,y) = (n_o - n_i)^{-1} \times \\ [I(x - \frac{\delta_{ox}}{2}, y - \frac{\delta_{oy}}{2}) - I(x - \frac{\delta_{ox}}{2}, y + \frac{\delta_{oy}}{2}) \\ + I(x + \frac{\delta_{ox}}{2}, y + \frac{\delta_{oy}}{2}) - I(x + \frac{\delta_{ox}}{2}, y - \frac{\delta_{oy}}{2})] \\ - [I(x - \frac{\delta_{ix}}{2}, y - \frac{\delta_{iy}}{2}) - I(x - \frac{\delta_{ix}}{2}, y + \frac{\delta_{iy}}{2}) \\ + I(x + \frac{\delta_{ix}}{2}, y + \frac{\delta_{iy}}{2}) - I(x + \frac{\delta_{ix}}{2}, y - \frac{\delta_{iy}}{2})], \end{aligned} \tag{4}$$

where $\delta_{ox}, \delta_{oy}, \delta_{ix}$ and δ_{iy} are the numbers of pixels that correspond to b_{ox}, b_{oy}, b_{ix} and b_{iy}, respectively, and n_o and n_i are the total numbers of pixels involved in the sums of the outer window rectangle and the inner window rectangle, respectively, so that the result is the mean pixel value of the pixels laying inside the outer window but not in the inner window [*].

The background map corresponding to the region of interest of the sonar image in Fig. 1(a) is shown in Fig. 1(c).

2.4 Echo Estimation

Once the background estimation is ready, the integral image is used again to construct an echo map. The purpose of the echo map is to help determine locations of high-intensity echo returns in the image that might have been produced by objects of interest.

The echo map is constructed using a single sliding window. The size of the sliding window used to construct the echo map is related to the size of the object we want to detect. Here, we use a window with width, $e_x = 1.5$ m and height, $e_y = 1.5$ m.

For each pixel in the region of interest of the sonar image, we compute the mean pixel value of the neighboring pixels laying in the window. The echo map value at location

[*] After the sea trials presented in this paper, we figured out that the background score calculation in Eq. 4 does not produce the expected result. Instead, the calculation should be either performed on the original image or rather on the integral image but using the four rectangles that actually contribute to the backgound scoring.

(x, y), $E(x, y)$, is then the mean pixel value in the window. Again thanks to the integral-image representation, each value can be computed quickly with only four array references to the integral image. Specifically, the echo map value at location (x, y) is calculated as

$$\begin{aligned} E(x,y) &= (n_e)^{-1} \times \\ [I(x - \frac{\delta_{ex}}{2}, y - \frac{\delta_{ey}}{2}) &- I(x - \frac{\delta_{ex}}{2}, y + \frac{\delta_{ey}}{2}) \\ +I(x + \frac{\delta_{ex}}{2}, y + \frac{\delta_{ey}}{2}) &- I(x + \frac{\delta_{ex}}{2}, y - \frac{\delta_{ey}}{2})], \end{aligned} \quad (5)$$

where δ_{ex} and δ_{ey} are the numbers of pixels that correspond to e_x and e_y, respectively, and n_e is the total number of pixels involved in the sum of the rectangle so that the final echo value is the mean pixel value inside the rectangle.

The echo map corresponding to the region of interest of the sonar image in Fig. 1(a) is shown in Fig. 1(d).

2.5 Potential Alarms Determination

After a background map, B, and an echo map, E, are constructed, the regions of the image that may possibly contain targets of interest are determined.

Any pixel for which the echo map value is sufficiently higher than the corresponding background map value is declared to be a region of interest that will receive further investigation. Specifically, if $E(x, y) > \beta B(x, y)$, then the pixel (x, y) is considered to be part of an echo highlight and therefore a potential alarm that will receive further examination. The scaling factor β adjusts the severity of the requirement for echo highlights. (Essentially, this defines that an echo highlight is when the pixel value is a certain amount more than the average value of the surrounding background.) We use $\beta = 1.2$ in our particular application.

This test is the first data-reduction stage of the detection process. It achieves a considerable reduction in pixels that must be examined further. In practice, more than 80% of the pixels are usually removed in this step. Thus, computational costs on all the subsequent stages are greatly reduced.

The binary result of the background and echo map comparison is shown in Fig. 1(e), where white pixels correspond to regions of potential alarms that will be investigated further.

Also in this very same phase, a standard labeling algorithm is applied to the resulting binary image. Therefore, on the subsequent stages we will only work with the potential alarm "blobs" determined here. As can be noticed, there are three such blobs in Fig. 1(e) in our case.

2.6 Geometrical and Morphological Filtering

Next, the determined potential alarm regions are filtered according to their geometrical and morphological properties. The purpose of this step is to filter out potential alarm regions with a geometry and/or morphology that do not correspond to the object we want to detect.

We use the major axis length and the minor axis length of the potential alarm region to determine if it must be discarded. Specifically, a region will be discarded if any of the following conditions is met:

- Region's major axis length is longer than a certain maximum major axis length, M_{max}
- Region's major axis length is shorter than a certain minimum major axis length, M_{min}
- Region's minor axis length is longer than a certain maximum minor axis length, m_{max}
- Region's minor axis length is shorter than a certain minimum minor axis length, m_{min}

Depending on the objects we want to detect, other region properties can be used for filtering, e.g. the circularity coefficient or the area. The map of potential alarms that remain after this stage are shown in Fig. 1(f).

2.7 Echo Scoring and Thresholding

Next, the echo score is calculated for the remaining potential alarms. For each alarm, i, its echo score, s_i, is the mean pixel value of all the pixels in the region, that is:

$$s_i = \sum_{\forall (x,y) \in A_i} (x, y), \quad (6)$$

where A_i is the set of pixels, (x, y) constituting the potential alarm i.

As the potential alarm blobs are not rectangular regions in general, the integral-image representation can not be used. Here however, the amount of pixels in each blob is typically small (as ensured by the previous geometrical and morphological filtering step), and therefore the computational performance of the algorithm is not compromised.

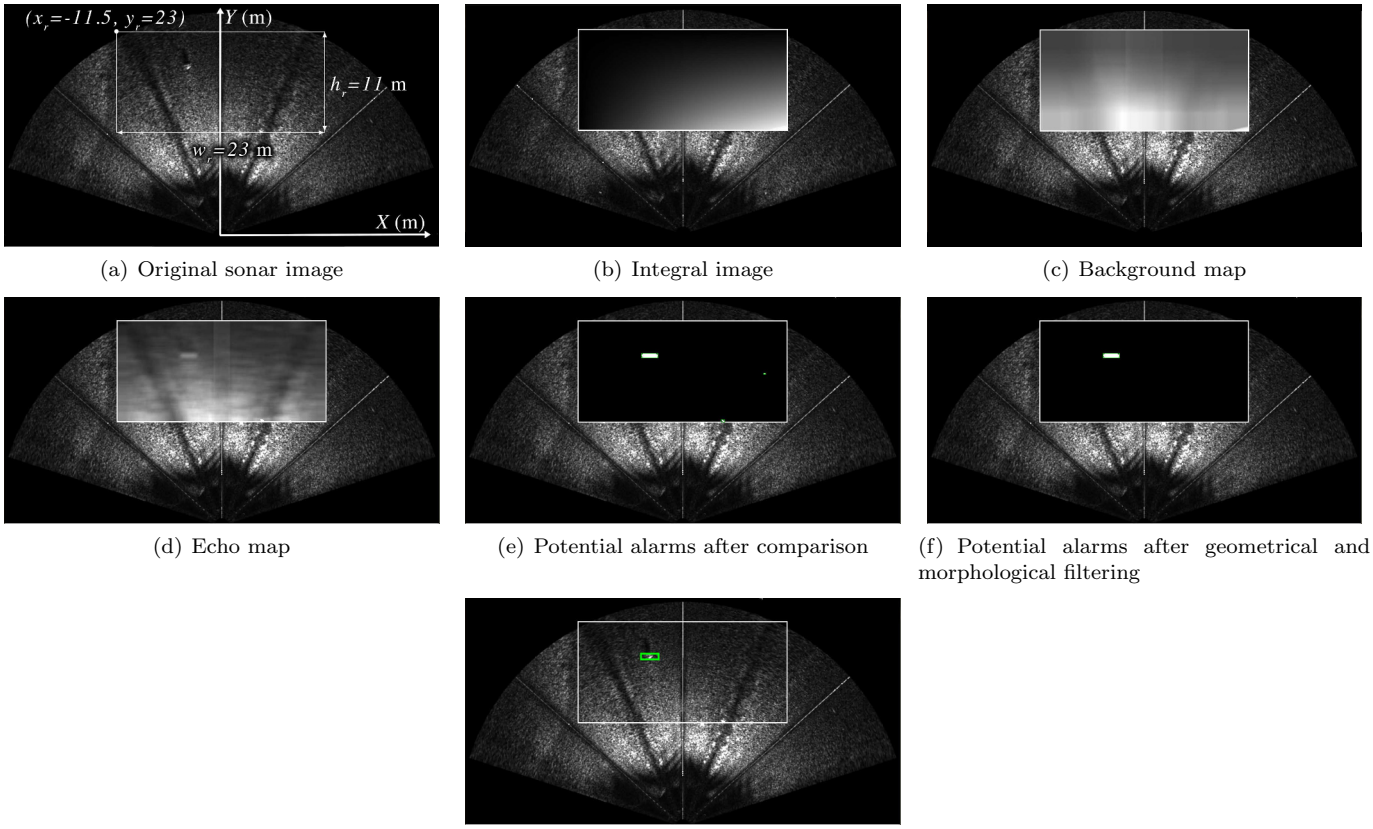
The echo scores are directly related to the intensity strength of the objects, which means that a detection threshold can be determined rigorously. Specifically, the threshold can be set such that we wish to detect any object for which the intensity strength exceeds a given level.

The final stage of the detection process removes those areas for which the echo score is below the desired threshold. The map of potential alarms that remain after this stage are shown in Fig. 1(g).

3. EXPERIMENTAL RESULTS

In October 2011, the NATO Undersea Research Centre (NURC) conducted the Autonomous Neutralization Trial (ANT'11) off the coast of Isola d'Elba, Italy. During this sea trial, four targets of two different shapes were deployed on the seafloor: cylinder shape and truncated cone shape. Targets were laying at depths ranging from 5 to 12 m. Multi-beam forward-looking sonar data were collected and processed in real time by the Gemellina ASV, which is equipped with a BlueView P900-130 900 KHz sonar. The sonar is mounted on a variable-depth pole (0 to 2 m depth) in the center of the ASV. The sonar can also be oriented by means of a pan and tilt unit (see Fig. 2).

The detection system presented in this paper was implemented and fully integrated into Gemellina's software



(a) Original sonar image

(b) Integral image

(c) Background map

(d) Echo map

(e) Potential alarms after comparison

(f) Potential alarms after geometrical and morphological filtering

(g) Final result after echo thresholding

Fig. 1. Detection algorithm. A region of interest of the original sonar image defined in the sonar coordinates system (a) is converted into its equivalent integral-image representation (b), which is then used to generate a background map (c) and an echo map (d). The comparison of the background map and the echo map generates a map of potential alarms (e). These potential alarms are then filtered according to their geometry and morphology (f). Next, an echo score is given to each remaining potential alarm. Finally, echo scores are thresholded producing discrete detections (g)

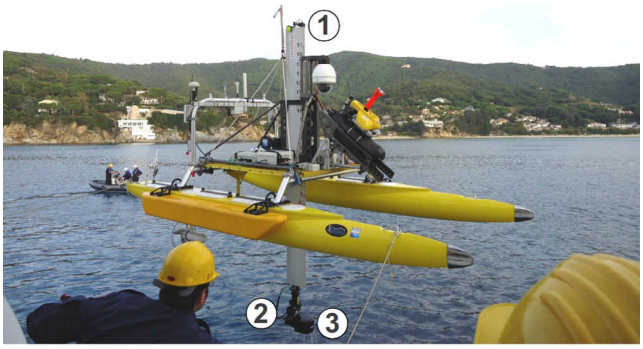


Fig. 2. The Gemellina ASV being deployed. 1: variable-depth pole; 2: pan and tilt unit; 3: BlueView P900-130 sonar

system, hence being able to run in real time. The algorithm was implemented in C++ and integrated in the Gemellina's MOOS-based (Newman, 2007) architecture.

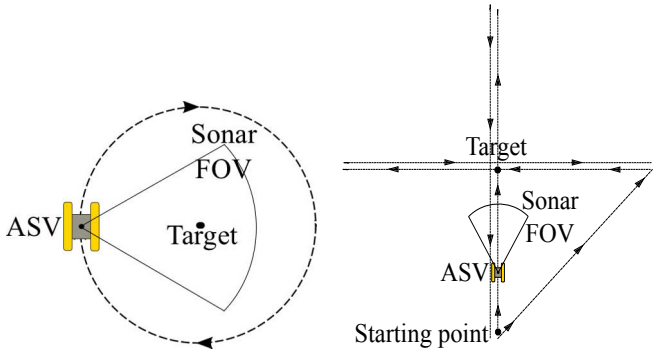
In order to test the proposed detection algorithm, several autonomous detection missions were carried out with Gemellina ASV. Those missions consisted of navigating around an *a priori* known target location tracing one of the two following target reacquisition patterns:

- The ASV circles around the target location, always keeping the target in the field of view of the sonar (see Fig. 3(a)). Circle radius is 15 m.
- The ASV traces a cross pattern centered on the target location. Each *arm* of the cross is 40 m long and the vehicle goes along each arm of the cross twice. The target is not always in the field of view of the sonar. See Fig. 3(b).

The maximum range of the sonar was set to 25 m in all missions because from the field trials in these environmental conditions we concluded that this was the limit for this type of sonar.

During every mission, the detection algorithm was run in every sonar ping, at a rate of approximately 2 pings per second. If a positive detection occurred, its location in the world coordinate frame (i.e., UTM coordinates) was stored for further examination. At the end of the mission, the mean position of all the stored detections was computed. Here we assume that most of the detections will be true positives, i.e., its location will correspond to the real target since the targets were deployed on a flat, sandy bottom without clutter around them.

A total of eight missions, four for each type of target (cylinder and truncated cone), were run to test the algo-



(a) Circle-around mission (b) Cross-pattern mission

Fig. 3. Trajectory patterns traced by the ASV around the known target locations

rithm. Table 1 shows, for each mission, the pattern traced by the ASV, the target depth and the distance error of the computed location with respect to the *a priori* known location.

Table 1. Target detection mission results

Mission pattern	Target type	Target depth	Distance error
Circle	Trunc. cone	6 m	3.8 m
Circle	Trunc. cone	10 m	1.6 m
Circle	Cylinder	5 m	1.1 m
Circle	Cylinder	12 m	3 m
Cross	Trunc. cone	6 m	2.6 m
Cross	Trunc. cone	10 m	1.4 m
Cross	Cylinder	5 m	0.6 m
Cross	Cylinder	12 m	1.6 m

4. CONCLUSION AND FURTHER WORK

A novel algorithm for the detection of underwater man-made objects on FLS imagery able to run in real time has been presented. The proposed pioneering and promising approach addresses limitations of existing generic underwater object detection algorithms by considering specific domain knowledge of the problem. By taking advantage of integral-image representations and by progressively reducing the computational load at every stage, the algorithm allows for real-time detection onboard an ASV or an AUV, as has been demonstrated in real experiments conducted at sea.

Related ongoing work that will be continued in the future will investigate the possibility of adding density filtering to the algorithm, that is, to ignore alarms concentrated on a certain small area, here assuming that the objects we want to detect will not be laying in cluttered patterns.

Future work will concentrate on reducing the number of false alarms by considering shadows casted by objects laying on the seabed. In this sense, another step might be added to the algorithm's cascaded architecture to make only objects that cast a shadow prevail.

Finally, we believe that the presented algorithm can be extended to perform detection and tracking of a mid-water target, such as a moving Unmanned Underwater Vehicle (UUV). Promising preliminary experimental results were already obtained in this regard during the ANT'11 sea trials.

ACKNOWLEDGEMENTS

The authors are gratefully thankful to Hans Groen and Warren Fox from NURC for their assistance during the algorithm development phase and to Alberto Grati and Stefano Fioravanti (also from NURC) for their effort during the implementation onboard of the ASV and the testing phase.

REFERENCES

Dobeck, G., Hyland, J., , and Smedley, L. (1997). Automated detection/classification of seamines in sonar imagery. In *Proc. SPIE International Society of Optics*, volume 3079, 90–110.

Fawcett, J., Crawford, A., Hopkin, D., Myers, V., and Zerr, B. (2006). Computer-aided detection of targets from the citadel trial klein sonar data. Technical report, Defence R & D Canada - Atlantic, Canada.

Groen, J., Coiras, E., and Williams, D. (2009). Detection rate statistics in synthetic aperture sonar images. In *Proc. Intl. Conf. & Exh. Underwater Acoustic Measurements*, 367–374.

Guo, J., Cheng, S.W., and Liu, T.C. (1998). Auv obstacle avoidance and navigation using image sequences of a sector-scanning sonar. In *Proc. of the 1998 International Symposium on Underwater Technology*.

Hayes, M. and Gough, P. (1992). Broad-band synthetic aperture sonar. *IEEE Journal of Oceanic Engineering*, 17(1), 80–94.

Lu, Y. and Sang, E. (1998). Underwater target's size/shape dynamic analysis for fast target recognition using sonar images. In *Proceedings of the 1998 International Symposium on Underwater Technology*.

Martin, A., An, E., Nelson, K., and Smith, S. (2000). Obstacle detection by a forward looking sonar integrated in an autonomous underwater vehicle. In *Proc. OCEANS 2000 MTS/IEEE Conf. and Exhibition*, volume 1, 337–341. doi:10.1109/OCEANS.2000.881281.

Maussang, F., Chanussot, J., Hétet, A., and Amate, M. (2007). Mean-standard deviation representation of sonar images for echo detection: Application to sas images. *IEEE Journal of Oceanic Engineering*, 32(4), 956–970.

Newman, P. (2007). Introduction to programming with moos.

Petillot, Y., Ruiz, I.T., and Lane, D.M. (2001). Underwater vehicle obstacle avoidance and path planning using a muli-beam forward looking sonar. *IEEE Journal of Oceanic Engineering*, 26.

Reed, S., Petillot, Y., and Bell, J. (2003). An automatic approach to the detection and extraction of mine features in sidescan sonar. *IEEE Journal of Oceanic Engineering*, 28, 90–105.

Viola, P. and Jones, M. (2004). Robust real-time object detection. *International Journal of Computer Vision*, 57(2), 137–154.

Williams, D.P. and Groen, J. (2011). A fast physics-based, environmentally adaptive underwater object detection algorithm. In *Proc. IEEE - Spain OCEANS*, 1–7. doi: 10.1109/Oceans-Spain.2011.6003424.