An integrated low-cost system for object detection in underwater environments

Journal Title 0 (0) 1–18
IOS Press

An Integrated Low-Cost System for Object Detection in Underwater Environments

Gian Luca Foresti [a,*] and Ivan Scagnetto [a]
[a] *Department of Mathematics, Computer Science and Physics, University of Udine, Italy*

Abstract. We propose a novel low-cost integrated system prototype able to recognize objects/lifeforms in underwater environments. The system has been applied to detect unexploded ordnance materials in shallow waters. Indeed, small and agile remotely controlled vehicles with cameras can be used to detect unexploded bombs in shallow waters, more effectively and freely than complex, costly and heavy equipment, requiring several human operators and support boats. Moreover, visual techniques can be easily combined with the traditional use of magnetometers and scanning imaging sonars, to improve the effectiveness of the survey. The proposed system can be easily adapted to other scenarios (e.g., underwater archeology or visual inspection of underwater pipelines and implants), by simply replacing the Convolutional Neural Network devoted to the visual identification task. As a final outcome of our work we provide a large dataset of images of explosive materials: it can be used to compare different visual techniques on a common basis.

Keywords: Artificial Vision, Object Detection, Deep Learning, UXO, OEW, Underwater ROV

1. Introduction

The widespread availability of relatively low-cost underwater Remotely Operated Vehicles (ROVs) opens interesting new possibilities about the exploration and the (partial) automation of several tasks in underwater environments, especially in shallow water. Such duties include surveillance and repair activities of submarine gas or electrical pipelines, inspection of submerged archaeological sites, etc.

Thus, in particular, there is a wealth of new applicative scenarios for a large class of artificial vision algorithms and solutions. Indeed, besides the technological improvements that made possible the advent of sophisticated and compact underwater ROVs, there is the need of making the latter "intelligent" and hopefully autonomous, i.e., able to carry out their tasks without the constant need of a human supervision. Along this road, one of the first things to achieve is to let the ROVs to safely and efficiently explore and move in their surroundings. In submarine environments, GPS and/or Wi-Fi positioning techniques do not work (there is no possibility to use GPS or Wi-Fi signals underwater and working solutions require to combine triangulation techniques by means of acoustic signals and surface GPS). Hence, artificial vision algorithms and solutions, alongside "classic" sonar-based techniques, may come to help in detecting obstacles and recognizing several landmarks and useful objects during exploration and pathfinding tasks (thanks to the availability of high resolution optical sensors and cameras).

In this paper, we propose a low-cost yet effective integrated system for object detection in submarine environments. Its main components, from the hardware point of view are an underwater mini-ROV, equipped with sensors and cameras, and a Ground Control Station (GCS) which allows the operator to pilot the mini-ROV and to acquire, store and process images. From a software standpoint, on the other hand, we have the following components: (i) the operating system of the mini-ROV with the related navigation software, (ii) the piloting app which interfaces the user with the mini-ROV and which stores the acquired images, and (iii)

*Corresponding Author: Gian Luca Foresti, Department of Mathematics, Computer Science and Physics, University of Udine, Via delle Scienze 206, 33100 Udine, Italy; E-mail: gianluca.foresti@uniud.it.

the image processing software which applies the filtering/enhancing and classification algorithms to the set of acquired images. More details will follow in Section 3.

In particular, we addressed the problem of identifying unexploded ordnance (UXO) materials (also known as OEW, i.e., Ordnance and Explosive Waste) on the seabed (or the river/lake bottom). Indeed, this represents a serious safety issue, in particular in the summer period, in many countries which were bombed during past or recent wars, when many people bathe at the sea/river/lake.

It may happen that coastal erosion or drifting debris (caused by natural phenomena like, e.g., storms) expose partially or totally some unexploded bombs (which were released by bomber planes or fired by mortars). Findings of this type are rather common in Italy: it is sufficient to read local and national chronicle news. For instance, [2] reports the news about a World War II unexploded bomb found at less than 20 m. from the shore near Rome. Moreover, in 2017 the demining division of Italian Navy removed about 22,000 bombs across seas, lakes and rivers, and during the first half of 2019 the bomb findings were more than 10,000. The US Government's Strategic Environmental Research and Development Program (SERDP) estimates that there are more than 10 million acres of coastal waters contaminated by undetonated explosives [14], without considering also ponds, lakes, rivers etc.

Beside the direct and immediate danger of undetonated items (i.e., the possibility of devastating explosions), there are also the damaging effects to the underwater ecosystem caused by the leakage of consistent amounts of dissolved explosive compounds, due to the fact that such items rust and corrode at sea, eventually breaking their cases.

Hence, it is clear that the goal of developing an autonomous or, at least, a remotely controlled system being able to monitor and search for unexploded bombs at the sea bottom would not be only an academic diversion, but would contribute in a significant way to public and environmental safety.

In this setting, the main innovative content of the paper is represented by:

1. a low-cost integrated system, allowing one to easily control a mini-ROV able to visually detect in real-time the presence of unexploded ordnance materials in shallow waters;
2. an extensive dataset of images of bombs (in various conditions) which does not exist in the literature and which can be used to effectively compare different visual techniques on a common basis.

2. Related Work

Due to the severe implications of the problem, in the literature there are plenty of systems and proposals aimed at the surveying and detection of OEW, especially in shallow water (i.e. up to 200 feet). Indeed, as mentioned before, millions of acres of ponds, lakes, rivers, estuaries and coastal ocean areas are scattered with munitions, especially in the case they are adjacent to active and former military installations. One of the first modern accounts of this kind of systems appeared in [11], where the authors presented MUDSS (Mobile Underwater Debris Survey System), i.e., a multi-sensor system for the surveying of underwater OEW. In particular, they resorted to a combination of two different sonars with a laser line scanner, capable of 6 mm resolution, and a gradiometer. Such sensors were depressed in shallow water by a mechanical wing attached to the surface craft. The onboard controls and electronics allowed the crew to read and process in real-time the acquired sensor data. Target detection was then improved by advanced processing and data fusion techniques, because the presence of clutter is a significant problem in shallow water. A 10 kW generator was needed to power all the equipment. The technique of deploying sensors (and in particular sonars) from a shallow-draft surface vessel is rather common even in the most recent scientific literature; for instance, in [6] a sonar system, producing three-dimensional synthetic aperture sonar (SAS) imagery, allows the authors to detect fully buried underwater objects. The latter is only one of the many examples of a long sequence of applications of acoustic and sonar techniques in this field. For instance, an extensive work has been funded by the USA Strategic Environmental Research and Development Program (SERDP) and the Office of Naval Research about the acoustic detection and classification of UXO in underwater environments [5, 9, 31, 32]. Among the outcomes of this research program there is a generative relevance vector machine (RVM) trained and used for identifying rockets buried underwater. In [16], the authors propose an active learning algorithm for classifying mine-like objects, without the need of any a priori training set. Some works go even beyond the issue of detecting/recognising OEW, being able to discriminate be-

tween different types of ordnance materials: in [18] a suitable matched subspace classifier (MSC) can distinguish between different classes of UXO based upon the spectral content of the sonar backscatter. There are also several approaches exploiting the visual analysis of sonar images. For instance, two acoustic lens sonars with high resolution and refresh rate are used in [7], providing a suitable solution for substituting optical systems in turbid waters. Moreover, sequences of forward-looking sonar images are used in [10] to implement a multiple tracking system, based on an application of a probability distribution called PHD (Probability Hypothesis Density): such sequences are then aligned and fused to reconstruct a 3-D map of the seabed. More recently, precise real-time or delayed real-time detection with sonar images is possible thanks to approaches like those in [41], where high-frequency data are combined with low frequency band images (limiting the number of false alarms), or [43], where issues due to object rotation, false targets and complex backgrounds are avoided by estimating the similarity of the extracted features with a previously acquired template. Other techniques applied to sonar data are features extraction and analysis by canonical correlation ([27]) and neural networks ([28]). Indeed, one of the advantages of this approach is that acoustic energy sensing does not suffer from range limitations and it can penetrate through various sources of turbidity, debris etc. in murky waters, where direct observations by means of acquired images are not viable. In particular, applying advanced machine vision techniques like, e.g., non-linear CNNs ([15]) to sonar imagery can lead to 99% of accuracy in distinguishing UXO objects from environmental clutter. On the other hand, direct optical observations, when there is a sufficient degree of visibility, clearly outperform all the other sensors (including sonars) in gaining meaningful and effective information for target localization, discrimination and identification, due to the visual cues and high level of details that can be extracted from high resolution images [3]. However, such wealth of information requires to acquire the images at short distances (in order to have high resolution and details) and without too much cluttering, thus limiting the visual perspective on the surroundings of the mapped region. Indeed, even the well-known and similar problem of object segmentation is more challenging underwater, due to the haze effect [12]. Indeed, in an underwater environment we must take into account the fact that the light is highly attenuated, distorting the colors and sometimes deforming the edges of the objects. Hence, color correction, contrast adjustment and other image enhancement techniques have been investigated and applied in [1, 17]. Other more sophisticated approaches are based upon convolutional neural networks for removing dust from images [23], on biologically inspired vision [22] or on conditional generative adversarial network-based models [20]. Preprocessing images, applying filters, is so common that in [24] a dataset of 950 images is proposed as a benchmark for comparing image enhancement algorithms and solutions of this kind. Indeed, preprocessing is usually the first step, in order to carry out object detection in images taken in an underwater environment. Several solutions address the problem of detecting fishes, using either statistical estimation techniques [8], or deep network models [29, 36, 42].

As to the detection of underwater OEW, apart from rather peculiar approaches like, e.g., the use of a tagged neutron inspection system for the detection of TNT explosives [37], one can find in the literature many sonar based approaches (see, e.g., [16, 18, 38]), because, as we mentioned before, they allow one to detect also buried or cluttered objects. In particular, in [3], the authors present a system making two steps. In the first step, underwater mosaicking techniques (applied to sonar images) are used in order to scan a large area in search of zones with a potential presence of munitions. Then, in the second step, a more careful scanning is performed only in the suspicious zones (target re-acquisition process). Applying similar mosaicking techniques to optical images can be difficult, despite the much higher resolution of the latter w.r.t. their sonar counterparts, because water turbidity heavily impacts on visual cues and, consequently, hinders the process of correctly aligning the images. However, in case of good visibility conditions, the approach allows one to make visual local searches for munitions across a wide map.

3. The Proposed System

In this section, we will provide a complete description of the proposed integrated system, including the physical architecture (see Section 3.1) and the logic architecture (see Section 3.2). From the hardware perspective, it is composed by the mini-ROV (with the payload of the external camera and lights), the ground control station (GCS, featuring a controller for the pilot), a tablet (displaying the footage acquired by the external camera), and a surface buoy (allowing the

communication between the mini-ROV and the ground control station). The software components are the piloting app (running on the GCS), and an ad-hoc app (running on the tablet to control the external camera and to apply the recognition algorithm). It is important to keep in mind that our proposal is not meant as a strict alternative to common solutions like, e.g., systems based on magnetometers, sonars, etc. Indeed, such sensors can be added to the payload of our mini-ROV as well. Moreover, we think that it can be a good complementary aid, providing users with the ability to establish a visual detection of unexploded ordnance materials, once an area has been marked as a potential site by means of some of the techniques outlined in the previous section. Indeed, our mini-ROV is small enough to explore underwater ravines or narrow tunnels and areas with too many debris to let a diver pass without some risks. As we will see in later sections, such maneuverability also allows us to pass very close to the objects of interest, greatly reducing or removing haze effects and distortions in acquired images. The novelty of our approach, on the hardware perspective, lies in the compact integration of a small vehicle with the sensor (i.e., the camera) used to acquire the data. The small dimensions of the assembled prototype allow the user to drive it in narrow spaces and to make close passes near the objects of interest. On the other hand, considering the software architecture, we have a highly parametric system, where the algorithms used in the preprocessing phase of data and the classifier can be easily replaced by alternative versions, according to specific needs. Moreover, we provide for the first time (according to our knowledge) a sufficiently large database of images of UXO materials which can be used as a common benchmark to compare different solutions in this field (see Section 4).

3.1. The Physical Architecture

A comprehensive scheme of the physical architecture is depicted in Fig. 1. As anticipated in Section 3, the reader may notice that the Ground Control Station (GCS) allows the user to control the mini-ROV by establishing a Wi-Fi connection with the buoy which can either be held on the ground or may be allowed to float on the surface of the water (like it is depicted in the previous figure). The buoy, in turn, is physically connected by means of a 100 m long tethering cable to the mini-ROV: all the commands and the video stream of the built-in onboard camera are transmitted along the cable (and then, via the Wi-Fi connection to and from the GCS).

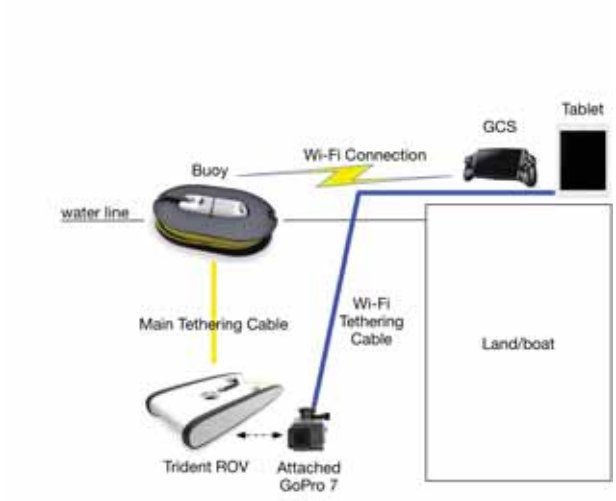

Fig. 1. An overview of the physical architecture of the proposed system.

Since the built-in onboard camera of the mini-ROV has a fixed orientation bearing forward (i.e., it is always pointing along the direction of movement of the vehicle), we exploited the possibility of attaching to the bottom of the mini-ROV another camera (henceforth denoted as external camera), pointing downwards, in order to be able to constantly monitor the seabed (or the river/lake bottom) without the need to carry out complicate maneuvers to tilt the onboard camera. Of course, in order to monitor the footage coming from this external camera we need a distinct tethering cable, in order to convey the second Wi-Fi link, avoiding dangerous interference with the mini-ROV link. This means we also need a second device (the tablet in Fig. 1) to control the external camera.

In Section 3.1.1, we will illustrate the features of the mini-ROV, of the Ground Control Station, and of the tablet with details about the two cameras used to acquire the images (see Section 3.1.2). Finally, in Section 3.1.3, we will describe the lighting system which allows us to operate in poor visibility conditions.

3.1.1. The mini-ROV, the Ground Control Station, and the tablet

As far as the mini-ROV, we opted for a low cost, small and agile vehicle in order to have a rather high degree of maneuverability, even in narrow spaces like, e.g., in shipwrecks scenarios and in submarine caves or ravines. More precisely, we customized the Trident

Underwater Drone (see [39]), whose technical specifications appear in Table 7 of the Appendix.

The role of the Ground Control Station (GCS) is carried out by the bundled Android OpenROV app, which runs on a JXD S192K controller [21]. The latter features ergonomic controls allowing the human operator to smoothly control the mini-ROV and the onboard camera and lights (with the possibility to watch in real time, on a 7" screen (or on attached video goggles), the video footage captured by the onboard camera).

Finally, the tablet can be any Android device: its purpose is to run an application allowing to display the footage coming from the external camera on the bottom of the mini-ROV and to apply the recognition algorithm (see Section 3.2 for the details).

3.1.2. Cameras

We have a built-in camera in the front of the mini-ROV pointing forwards: it is activated automatically when the vehicle is turned on. Video registration and image acquisition activities are controllable directly from the Android OpenROV app. The technical features of this camera are listed in Table 8 of the Appendix.

In order to leave the user free to pilot the mini-ROV, without the hassle of performing at the same time complicate maneuvers to point to the objects of interest on the seabed (or the river/lake bottom), we opted for installing a second camera on the bottom hull of the vehicle. This external camera is a GoPro 7, whose technical features are listed in Table 9 of the Appendix. It has been chosen essentially for the compact dimensions, the quality (much better than the built-in camera) and the availability of 3D-printable supports that can be easily attached to the bottom plate of the Trident mini-ROV. Moreover, such supports are fully compatible with the 60m waterproof encasing of the GoPro 7.

Other cameras can be easily added according to the operative needs of the current deployment scenario (Fig. 2).

3.1.3. Lighting System

The Trident mini-ROV is equipped with a lighting system composed by six forward facing LEDs (three on each side of the device) providing 360 lumens and with a color temperature of 4000 K. Thus, the built-in camera can benefit from those LEDs, in order to compensate to the darkness in submarine environments. Moreover, we added on the bottom part of the mini-ROV hull a lighting panel composed by 84 LEDs providing 1800 lumens with a color temperature of 5500 K. Hence, also the external camera can benefit of a lighting system pointing towards the seabed (or the river/lake bottom).

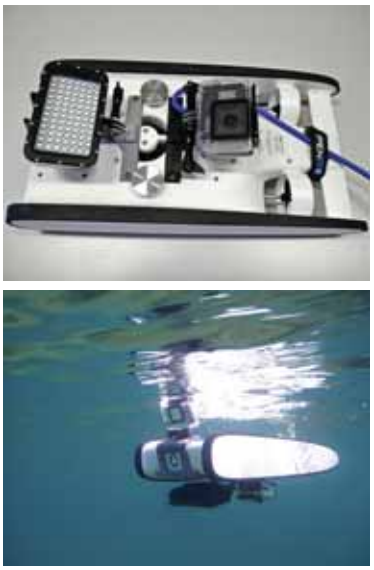

Fig. 2. The bottom of the Trident mini-ROV with the standard payload of Fig. 1 in our lab (top), and in action with an additional camera on the upper front part (bottom).

3.2. The Logical Architecture

Besides the piloting software of the mini-ROV vendor, which provides the user with an intuitive user interface allowing him to move the vehicle and to control the lights and the front camera, we implemented a tablet application receiving the video stream provided by the GoPro camera, and extracting individual frames from it. Those frames are then processed by a Convolutional Neural Network (CNN), trained to recognize the presence of unexploded ordnance materials. In the case of a positive detection, the involved frames are saved in the tablet file system for later inspection, with the detected ordnance materials visually marked, and georeferenced with the current coordinates provided by the tablet GPS. Moreover, a specific graphical element of the app UI is highlighted and an alarm sound warns the user. This data pipeline is depicted in Figure 3.

Since it is well known that underwater images are subject to distortions in shape, colors and brightness, we introduced a preliminary filtering step in the processing pipeline of the system, in order to minimize such effects, before feeding the image frames to the CNN (both in the training step and, later, during the classification activity).

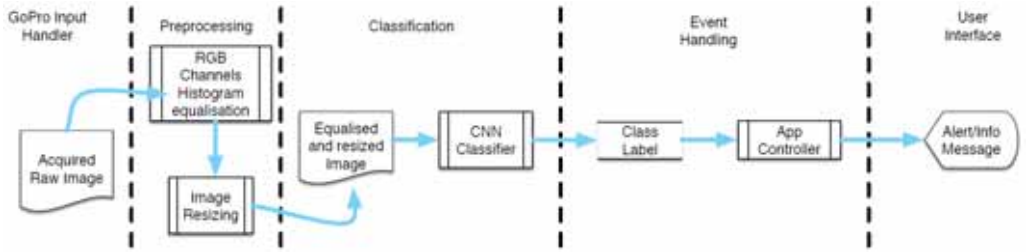

Fig. 3. The data pipeline of the proposed system.

In Fig. 4 we can see a running example of the preprocessing phase. We selected a problematic case, in order to illustrate the issues that may occur with underwater images. First of all, the agility and small dimensions of the mini-ROV allow us to go very close to the objects of interest; thus the vicinity and the powerful lights on the bottom of the mini-ROV reduce significantly all the haze effects and distortions. Then (from top to bottom in Fig. 4), we equalise the histograms of the three color channels (R, G, and B), and we resize the image to the dimension required by the CNN (i.e., 227×227). In this case the CNN classifier has been able to recognize the presence of the hand grenade[1] with a degree of confidence greater than 0.9, despite the partial occlusion due to the rock on the left, and the fact that part of the grenade has been cropped out. Moreover, the image has not being included in the training set; hence, there is no overfitting here.

Before introducing the details of the CNN we used, it is important to highlight that our system is parametric about the method (e.g., the above mentioned CNN) used to recognize unexploded ordnance materials in the processed images.

Of course, in order to teach a machine "what is a bomb", together with positive samples, we also have to provide negative samples representing "what is not a bomb". This has been a hard challenge, due to the difficulty of retrieving an adequate number of training images (see Section 4). Indeed, in order to have a balanced training set, we had to limit the negative samples (which can represent whatever) to the same size of the positive samples. Hence, we decided to exploit a transfer learning from an already trained model, namely, the BAIR/BVLC CaffeNet Model[2]

Fig. 4. An acquired image with a partially exposed and cropped UXO (top), with RGB color histograms equalised (middle) and resized (bottom), before being fed to the CNN.

(using the Caffe Framework[3] and trained on the ImageNet dataset[4] containing millions of images) which can classify 1,000 categories (not including bombs). Of course, we changed the number of outputs from 1,000 to 2 (bomb/not bomb): the resulting CNN model

[1] Grenades, although being explosives, are not classified as bombs, but for convenience, throughout the paper, with the word *bomb* we also refer to grenades.

[2] This model is a slight variation of the AlexNet CNN and it is available at https://github.com/BVLC/caffe/tree/master/models/bvlc_reference_caffenet

[3] https://caffe.berkeleyvision.org/

[4] Available at https://image-net.org/download.php.

Model	BAIR/BVLC CaffeNet Model (AlexNet)
Layers	5 Conv/ReLU, 3 FC
Dimensions of input images	227×227
Training batch size	16
Testing batch size	24
Max iterations	40,000
Number of outputs	2 (0=bomb, 1=not bomb)

Table 1
Details of the CNN.

is depicted in Fig. 5 (due to lack of space, we arranged the diagram layout in five rows), and the important details of the network are resumed in Table 1.

It is important to notice the values of two key parameters of the model, since they have a direct impact on the overall performance of the resulting classifier. In particular, the training batch size (i.e., the number of images processed together during the training phase in a single iteration) has been set (after some preliminary experiments) to 16, which is a rather low value. The reason of this choice is mainly due to the fact that working with small batches of images favors the ability of the resulting classifier to generalize beyond the specific dataset used in the training phase. In other words, it helps avoiding the overfitting phenomenon, since small input sets introduce some noise in the gradient estimation.

The other reason to keep the training batch size small is to consume less memory: this is particularly important when using GPUs to accelerate the training process. Memory consumption is also the reason of having set the value of the testing batch size to 24: this is the maximum we could allow, in order to avoid the "insufficient memory" error message during the validation phase[5].

The second key parameter is the maximum number of iterations (max_iter) which we changed to 40,000 (the original value is 450,000), since during our experiments we noticed that after 10,000 iterations there are no significant improvements.

[5]The validation dataset is used to provide an unbiased evaluation of a model fit on the training dataset, allowing to tune the model parameters. As the training progresses, the "skills" on the validation dataset is incorporated into the model configuration, making the evaluation more biased.

The remaining parameters are set according to the original trained model[6]. Dataset images are resized to a 227x227 spatial resolution and divided as follows: one set for training (85% of the total set of images, including positive and negative examples of the presence of bombs) and the other (the remaining images) for validation. Thus, the former set is used to train the model, and the other one is used to calculate its accuracy. Both sets are stored as LMDB[7] databases.

In particular, a fine-tuning of the trained model is performed by continuing the backpropagation on an initial dataset of 4,600 images (2,300 images with bombs and 2,300 without bombs). We attained an accuracy in the training phases higher than 0.9, after only 10,000 iterations over the maximum of 40,000 which we programmed (see Fig. 6).

Hence, our network started to be quite effective in recognizing bombs (see Section 5), even when partially cluttered by debris or dirt (like it happens when we are searching for OEW with our mini-ROV). A complete training over our dataset of 4,600 images requires less than 30 minutes on a Linux System equipped with an Intel Core i9-10900KF CPU with 32 GB of RAM and a Nvidia RTX 2080 GPU card with 8 GB of memory and Turing microarchitecture.

4. The Dataset

Unfortunately, as far as the availability of image datasets of OEW, there are two kinds of issues to consider. First of all, there are no publicly accessible databases of this kind which can provide more than a few dozens of images. For instance, one of the largest online databases of unexploded ordnance is [40] and the underwater ordnance category provides only 49 images. The second issue is that images of underwater OEW are even rarer. Obviously, this affects negatively the training phase of supervised algorithms. Hence, we decided to build our own dataset of ordnance materials. We adopted two strategies: first of all, we started crawling the web. We developed some Python scripts using the Selenium library, in order to simulate a user

[6]The solver parameters can be found at the following URL: https://github.com/BVLC/caffe/blob/master/models/bvlc_reference_caffenet/solver.prototxt

[7]Lightning Memory-Mapped Database (LMDB) is a software library written in C with bindings available for several programming languages, providing an embedded transactional key-value store. It is very convenient to use in a multi-threaded environment where high read performances are requested.

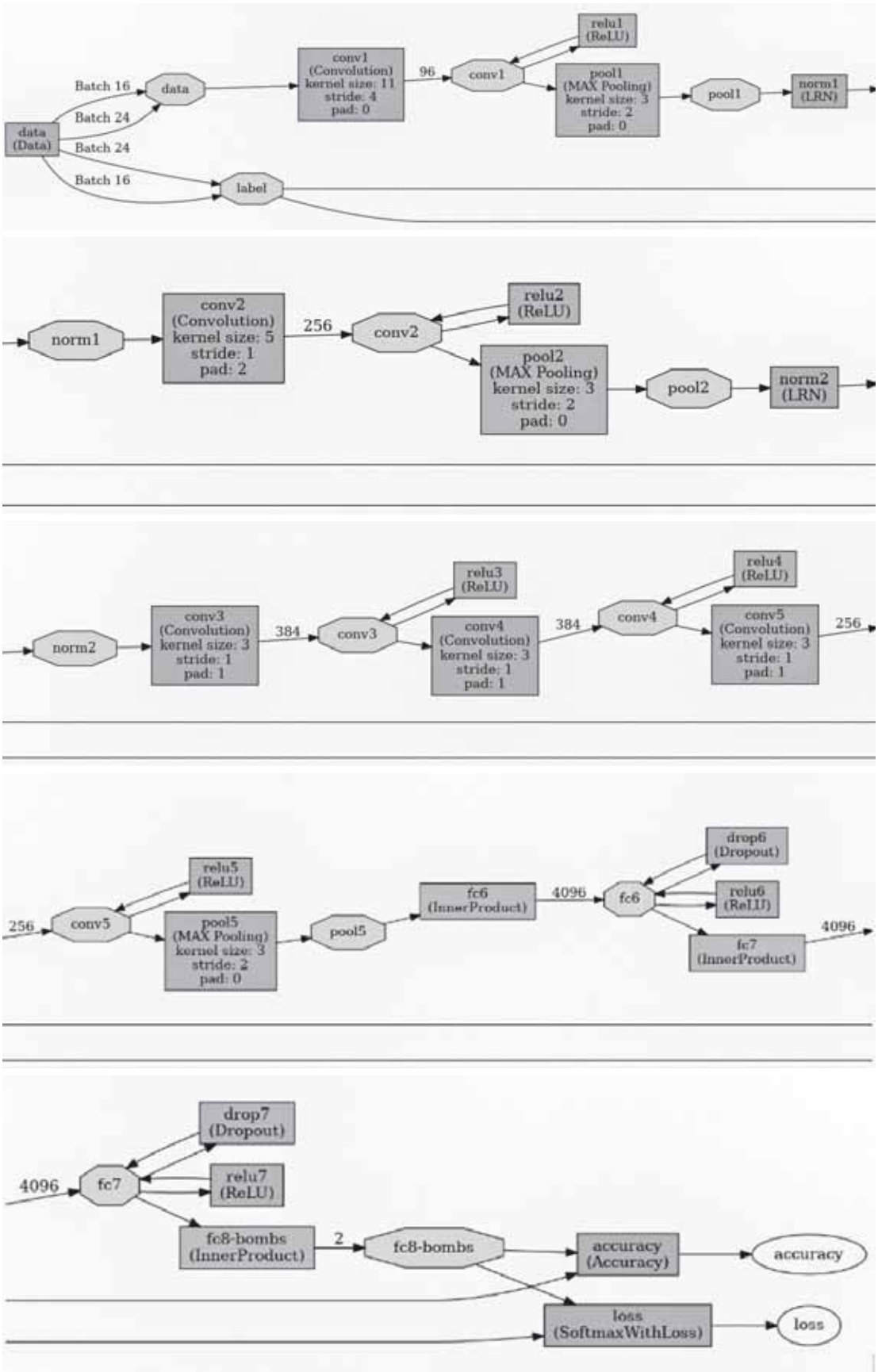

Fig. 5. Model of the proposed CNN.

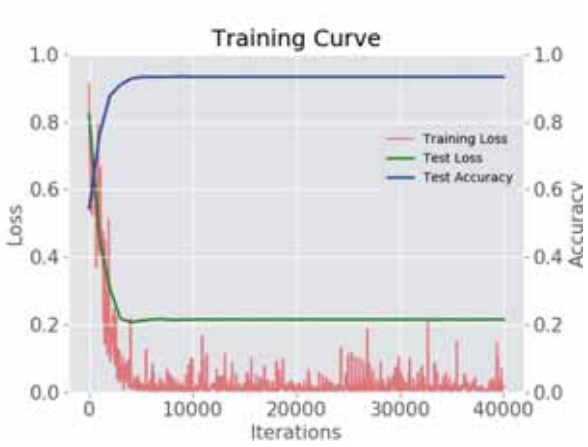

Fig. 6. Training curve with a dataset of 4,600 images.

session with common browsers (e.g., Chrome or Firefox) and querying Google Images with keywords like UXO, unexploded bombs, unexploded ordnance and so on (both in English and Italian languages, in order to exploit also websites of local newspapers[8]). The second strategy was suggested by the fact that most of the available images on the Web depict bombs pulled ashore; hence, colors and other visual features of the objects are significantly different than when underwater. Hence, we envisaged a procedure, in order to enrich our training dataset with underwater images of already recovered bomb shells. In particular, we started adding images of real bomb shells which were provided to us for the experiments by the Italian Army. Subsequently, we started acquiring images of those same bomb shells immersed in different underwater scenarios with varying conditions of visibility and cluttering. More precisely, we considered in a first series of experiments a clean water tank with clean water. Then, we started filling the tank with murky water and adding debris and various forms of cluttering. In a second phase, we replicated the experiments on a real seabed, starting from shallow waters (with clear visibility conditions) to continue with a deep seabed with poor visibility conditions. Finally, we moved to underwater regions with some debris and cluttering. Summing up, we considered the following conditions:

1. a clean tank filled with clean water;
2. a clean tank filled with murky water;
3. a cluttered tank filled with clean water;
4. a cluttered tank filled with murky water;

[8] As we noticed in the introduction, unexploded bomb findings are rather common in Italy.

Fig. 7. Some bomb shells and replica images acquired ashore (upper row) and underwater (bottom row) in different conditions (on the bottom of a swimming pool and on the sea bottom in shallow waters).

5. a shallow seabed with good visibility conditions and no cluttering;
6. a cluttered shallow seabed with good visibility conditions;
7. a deep seabed in murky waters, but no cluttering;
8. a cluttered deep seabed in murky waters.

At the end we generated a database of underwater images of several ordnance materials in different conditions. Some of them are reported in Fig. 7, while Table 2 reports the composition of the whole dataset (including photos downloaded from the Web) which can be freely downloaded from [44]. Such images were then fed to the Neural Network Classifier (NNC), in order to train the network.

Thus, we aimed to "reuse" the knowledge gained while learning to visually recognize unexploded bombs in land environments, applying and modifying the previously built neural network classifier to underwater environments. So doing, we expect in the long term to improve the effectiveness of our system in detecting bombs in underwater scenarios.

After several tests, we are confident that our system can indeed be useful as a low cost aid for the detection of OEW materials in shallow waters. Fortunately,

Total images	4,600	(a+b)
Total UXO images	2,300	(a=c+d+e)
Images without UXO	2,300	(b)
UXO web images	1,957	(c)
Underwater UXO web images	283	(d)
Underwater UXO images acquired by mini-ROV	60	(e)

Table 2
Composition of the UXO images dataset.

the transfer learning from an existing model trained on the ImageNet database helped us, raising significantly the test accuracy of our CNN, while containing the test loss. We still have some false positives and false negatives with certain inputs, but they are decreasing with each iteration. Thus, the system effectiveness is heavily dependent on the training of the CNN, i.e., on the quality and size of the dataset. Hence, the usage we propose is an incremental one: start with an initial dataset and use transfer learning from an already trained model (see Section 3.2) to begin with (as we explained in the previous section). Then improve it by repeating the training phase with a new bunch of images acquired either from the field or from publicly available databases (e.g., crawling the web). Finally, if you have some empty bomb shells or even some collector's replicas you can follow the procedure previously described in this section, in order to progressively enlarge your training dataset.

5. Experimental Tests

We tested our trained CNN against new sets of images taken from bomb findings published over the Internet and images of bomb shells provided to us by the Italian Army, which were not used during the training phase. During our tests, we had a very high accuracy in detecting bombs. Indeed, the only misclassified images are rather confusing even for a human being trying to guess which object is displayed. Usually they are photos of objects with so much dirt that they can be confused with bricks, rocks or debris.

Fig. 8 shows three detections in experiments made directly using the mini-ROV and our UXO replicas, in the settings mentioned in Section 4, i.e., deploying our replicas and bomb shells underwater in different conditions. The bounding box has been generated after the classification, using a tool available in the Caffe framework. Next to it there are the classification label and the related confidence degree.

Fig. 8. Detections on images of bomb shells acquired using our mini-ROV, according to the procedure described in Section 4.

Apart from those experiments, we also checked the effectiveness of our CNN in detecting bombs in images found on the web. In particular, Table 3 (where TP, FP, TN, FN stand, respectively, for True Positive, False Positive, True Negative, and False Negative) reports the outcomes of two extensive bomb detection tests made with our network. Test 1 was conducted on a dataset of 4,264 images. Such images were randomly downloaded using the Google search engine: half of them were pictures featuring bombs of generic nature (not necessarily underwater), whereas the other half were pictures without bombs. Moreover, we carefully checked that none of the images were already used

in our training set[9]. Instead, the dataset of Test 2 was composed by underwater bombs for 50%, whereas the rest were, again, images without bombs in underwater environments (all images were again randomly downloaded from the Internet and we checked that they did not occur in the training set). As we can see, in both cases the behavior of the CNN features good scores of precision, recall, accuracy and F1-measure (always at or above 90%).

Since in our code we handle images in RGB format, we also tested the sensitivity of the accuracy of the CNN classifier, w.r.t. the insertion of gaussian noise into the three color channels of the input images. We selected a balanced dataset of 300 images downloaded from the web and acquired by our mini-ROV: 150 of them were images of UXO materials, correctly recognised by the CNN classifier, while the remaining 150 were images not related to the UXO category, correctly discarded by our CNN classifier. According to [30], Gaussian RGB noise is applied to each image color channel independently, with mean 0 and standard deviation σ. As it can be seen from Table 4 (where the value 0.0 of σ means absence of noise, i.e., the original image is left unchanged) and Fig. 9, the accuracy is very stable (above 0.9) with $\sigma \geq 100$, while it considerably degrades only for values greater than 100: in [30] the tested values of σ are less than 40 and at 35 the accuracy drops to 0.4. The good behaviour of our CNN classifier is probably due to the preprocessing phase, where we equalize the histograms of the RGB channels (see Section 3.2).

We conclude this section with the description of the last benchmark we made to test the flexibility of our system and, in particular, of the CNN classifier we adopted. Indeed, although we said that it can be easily replaced by other (possibly better) solutions, we want to convince the reader that good results can already be achieved with the current setup.

Hence, we considered an application scenario where, beside discriminating UXO materials from other kinds of objects, it is needed to also classify the latter as belonging to a certain set of categories. In order to make things more difficult, we chose to consider ImageNet classes of underwater elements which may be visually confused with UXO. Moreover, where possible, we used the EUVP dataset[10] for testing, since it provides

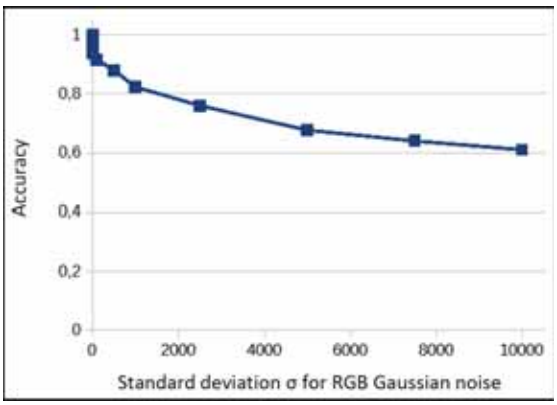

Fig. 9. Sensitivity of accuracy w.r.t. gaussian noise applied to input images.

a mix of poor quality images with enhanced (or better quality) images. The categories we considered are as follows:

1. UXO: UneXploded Ordnance materials in underwater environments, i.e., the main focus of our work;
2. Coral[11]: since UXO underwater materials are often covered by marine encrustations, corals may easily induce a classifier in error;
3. Sea Turtle[12]: this category of animals can be exchanged for grenades or other kinds of explosive materials, due to their carapace;
4. Pufferfish[13]: this kind of fish may resemble to a bomb in certain circumstances (especially when it inflates its body);
5. Torpedo[14]: another kind of animal which can be exchanged for a bomb (especially when it plunges into the sand of the seabed);
6. Sea Anemone[15]: a lifeform which may grow near or on bomb shells in the seabed;
7. Anemone Fish[16]: a kind of fish usually living among sea anemones (it has been considered to add another kind of difficulty for the CNN classifier);

[9] To guarantee that no images are present both in the training set and in the test set, we name the files with the hash values of their contents.

[10] Available at http://irvlab.cs.umn.edu/resources/euvp-dataset.

[11] ImageNet classes n01917289 (“brain coral”) and n09256479 (“coral reef”).

[12] ImageNet classes n01664065 (“loggerhead, loggerhead turtle, Caretta caretta”) and n01665541 (“leatherback turtle, leatherback, leathery turtle, Dermochelys coriacea”).

[13] ImageNet class n02655020 (“puffer, pufferfish, blowfish, globefish”).

[14] ImageNet classes n01496331 (“electric ray, crampfish, numbfish, torpedo”) and n01498041 (“stingray”).

[15] ImageNet class n01914609 (“sea anemone, anemone”).

[16] ImageNet class n02607072 (“anemone fish”).

	Total	TP	FP	TN	FN	Precision	Recall	Accuracy	F1-measure
Test 1	4,264	2,047	86	2,046	85	0.959	0.960	0.959	0.959
Test 2	210	99	11	94	6	0.900	0.942	0.919	0.920

Table 3

Tests of bomb detection in images downloaded from the Web.

Gaussian Noise σ	Accuracy
0.0	1
0.1	1
1.0	0,96
10.0	0,94
100.0	0,92
500.0	0,88
1000.0	0,82
2500.0	0,76
5000.0	0,68
7500.0	0,64
10000.0	0,61

Table 4

Gaussian noise σ vs. Accuracy.

8. Rock Beauty Fish[17]: a kind of fish which may be confused with anemone fishes (it has been considered to add another type of difficulty for the CNN classifier).

Having to deal with eight categories of items, we retrieved as many images as possible, either from ImageNet or from the Web (using our crawler). Thus, we set up the dataset described in Table 5 with a total of 26,880 images (3,360 for each category). As for the former experiments of binary classification, 85% of the total set of images (equally partitioned among all the eight classes) was used to train the CNN, while the remaining images were used for validation.

The test set has been taken from the ImageNet section of the EUVP dataset (100 images of poor quality and 100 images of high quality) for each of the following classes: corals, sea turtles, torpedoes, sea anemones, anemone fishes and rock beauty fishes. For the remaining classes of UXO and pufferfishes the 200 images per class have been downloaded from the Web. Thus the test set amounts to a total of 1,600 images. Every image downloaded with the crawler (for the training, validation, and test sets) has been checked and labeled by a human operator. All the images can be downloaded from [44].

[17] ImageNet classes n02606052 ("rock beauty, Holocanthus tricolor").

Table 6 provides the final statistics for this multiclass classification experiment. As we can see, accuracy is good (greater than 0.9) for all the categories taken under consideration. F1-measure is lower (but still higher than 0.8) for the following classes: corals, sea anemones and anemone fishes. Looking also at the confusion matrix reported in Appendix B, this was not an unexpected result, and the reason behind that may be the fact that the elements belonging to those categories often are present together in the same environment. Indeed, images depicting sea anemones often contain also anemone fishes and viceversa. Corals often share their environment with those categories as well.

UXO materials have the second best results for accuracy and F1-measure, directly behind rock beauty fishes which are easily recognisable, because of their peculiar combination of colors.

Looking at the state-of-the-art in the field of UXO detection and recognition by means of computer vision techniques, a closely related paper is [13], where a CNN is used to identify "improvised explosive devices" (IEDs) in "rural or built-up urban environments", i.e., an *out of the water* version of our experiments. However, the input data of the CNN are harvested by an ad-hoc sensor which is composed by a ground penetrating radar, a thermal sensor, an infrared sensor, an ultraviolet sensor and a camera. Hence, there is a wealth of data which allows one to detect buried objects too, yielding an accuracy of 98.7%, in well-lit conditions. Of course, using a simple camera in our setup, we cannot detect buried or strongly occluded objects; however, looking at Table 6, the overall accuracy of our test is still a good one (always above 90% for all classes of objects), and it achieves a score just below 98.7% for the UXO class (to be precise the actual precision value is 0.986875 which has been rounded to 0.987 in Table 6).

Other significative, but less related, works are devoted to visually recognising objects either for zone safety purposes ([34]), or for structural health assessments of infrastructures ([25]). They resort to sophisticated techniques which go beyond to the use of CNNs and they achieve precision scores of 97.2% and accuracy scores of 95.99%.

Another kind of studies focuses on damage detection in civil structures ([33]), and anomaly detection in video surveillance ([35]) or in a wider range of applications ([26]).

Finally, in the field of content based image retrieval, there are rather sophisticated studies as far as image classification is concerned. For instance, in [19], the authors adopt an ensemble learning approach: they introduce an architecture leveraging on an ensemble of CNNs. Such networks are either built on the same model, but they are then trained on different sets of images, or are trained on the same dataset, but they differ from the architectural point of view. The ensemble of CNNs thus generates a final image representation which is more general and robust for the purpose of image classification, outperforming many individual CNN architectures.

However, leaving out the above mentioned works (which are rather different and difficult to compare), if we focus on the task of recognising UXO in underwater environments and we look at the attained accuracy scores of our tests, we can say that our approach is quite effective and comparable to state-of-the-art performances, although striving for simplicity and low cost.

6. Conclusion and Future Work

We proposed a low cost system for detecting UXO (OEW) in underwater scenarios, by means of a mini-ROV (with an attached camera) and a suitably trained CNN classifier. Despite it is well-known that the lack of clarity in the acquired images makes segmentation and recognition more difficult in underwater environments, our experiments suggest that there is an advantage in applying an image-based approach to this task. Indeed, even if we cannot achieve results comparable with solutions based on georadar/sonar, we can still provide an inexpensive and effective alternative to these approaches. It is sufficient to think that a cheap sonar for underwater navigation costs around 3,000 USD, which is a price far higher than the cost of our whole prototype. Indeed, as far as experiments are concerned, the only other real cost is the training phase of the CNN (executing the resulting classifier can be done on cheap hardware, such as a common tablet), but it is possible to train the network exploiting some free test accounts of a cloud solution like, e.g., Amazon AWS or Paperspace[18] (the latter offers a free account including the availability of a GPU with a Python programming environment). Hence, the system can be affordable even for people living in third world countries (think, e.g., of the navy, coast guard, fishermen etc.). Moreover, there is also the advantage of the small dimensions and of the maneuverability of the mini-ROV which can be easily transported and deployed by a single person, without the need of huge boats and large crews.

Another contribution of this paper is the beginning of the definition of a reference dataset of UXO materials, in order to provide a common benchmarking platform for researchers working in this field. In particular, besides crawling and downloading from the Web available UXO images, we started adopting the incremental approach described in Section 4 to build a progressively larger and more representative database of OEW.

Due to the large variety of OEW (e.g., aircraft bombs, high-explosive bombs, land and water mines, hand grenades, mortar bombs, etc.), a possible future work could be to specialize the CNN to classify bombs in different categories, according to an existing and standard classification. This would help users to have a more precise evaluation of the kind of detected explosive item (e.g., detecting the presence of a hand grenade requires a complete different approach and resources to secure the area and defuse it w.r.t. the procedure involving a large high-explosive bomb). Of course, another possibility of future work will be the testing of other types of deep machine learning architectures, in order to find the best performing one for the purpose of UXO identification.

Finally, in order to make the whole system even more economical, it would be interesting to explore the possibility to use boosting techniques with the improvements suggested in [4], as they are far less demanding in terms of the hardware resources (like, e.g., GPUs) needed in the training phase.

References

[1] Ancuti C, Ancuti CO, Haber T, Bekaert P. Enhancing underwater images and videos by fusion. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. IEEE; 2012, p. 81–88.

[18] https://www.paperspace.com/.

Category	total n. of images	Origin
UXO	3,360	2,300 (our dataset: Table 2), 1,060 (Web)
Corals	3,360	2,600 (ImageNet), 760 (Web)
Sea Turtle	3,360	2,600 (ImageNet), 760 (Web)
Pufferfish	3,360	1,300 (ImageNet), 2,060 (Web)
Torpedo	3,360	2,600 (ImageNet), 760 (Web)
Sea Anemone	3,360	1,300 (ImageNet), 2,060 (Web)
Anemone Fish	3,360	1,300 (ImageNet), 2,060 (Web)
Rock Beauty Fish	3,360	970 (ImageNet), 2,390 (Web)

Table 5

Composition of the dataset used to train the CNN in the multiclass classification experiment.

	TP	TN	FP	FN	Precision	Recall	Accuracy	F1-Measure
UXO	185	1394	6	15	0.969	0.925	0.987	0.946
Corals	173	1379	21	27	0.892	0.865	0.970	0.878
Sea Turtles	178	1387	13	22	0.932	0.890	0.978	0.910
Pufferfishes	183	1375	25	17	0.880	0.915	0.974	0.897
Torpedoes	178	1388	12	22	0.937	0.890	0.979	0.913
Sea Anemones	171	1360	40	29	0.810	0.855	0.957	0.832
Anemone Fishes	177	1374	26	23	0.872	0.885	0.969	0.878
Rock Beauty Fishes	199	1387	13	1	0.939	0.995	0.991	0.966

Table 6

Multiclass classification of underwater images.

[2] Ansa R. Anvcg, in mare bombe guerra inesplose - Puglia [Internet]. ANSA.it. 2018 [cited 2022 Jan 20]. Available from: http://www.ansa.it/puglia/notizie/2018/06/29/anvcg-in-mare-bombe-guerra-inesplose_229c0974-4d22-4ede-8773-76bb8bbc1fa6.html.

[3] Beaujean P-PJ, Brisson LN, Negahdaripour S. High-resolution imaging sonar and video technologies for detection and classification of underwater munitions. Mar Technol Soc J. 2011;45(6):62–74, doi:10.4031/mtsj.45.6.6.

[4] Buenaposada JM, Baumela L. Improving multi-class Boosting-based object detection. Integr Comput Aided Eng. 2020;28(1):81–96, doi:10.3233/ica-200636.

[5] Bucaro JA, Houston BH, Saniga M, Nelson H, Yoder T, Kraus L, Carin L. Wide area detection and identification of under- water UXO using structural acoustic sensors. Naval Research Lab, Washington DC; 2007.

[6] Brown DC, Johnson SF, Gerg ID, Brownstead CF. Simulation and testing results for a sub-bottom imaging sonar. In: 177th Meeting of the Acoustical Society of America. ASA; 2019, doi:10.1121/2.0001012.

[7] Belcher E, Matsuyama B, Trimble G. Object identification with acoustic lenses. In: MTS/IEEE Oceans 2001 An Ocean Odyssey Conference Proceedings (IEEE Cat No01CH37295). Marine Technol. Soc; 2002;1:6–11, doi:10.1109/OCEANS.2001.968656.

[8] Boudhane M, Nsiri B. Underwater image processing method for fish localization and detection in submarine environment. J Vis Commun Image Represent. 2016;39:226–238, doi:10.1016/j.jvcir.2016.05.017.

[9] Bucaro JA, Waters ZJ, Houston BH, Simpson HJ, Sarkissian A, Dey S, et al. Acoustic identification of buried underwater unexploded ordnance using a numerically trained classifier (L). J Acoust Soc Am [Internet]. 2012;132(6):3614–7, doi:10.1121/1.4763997.

[10] Clark DE, Bell J. Bayesian multiple target tracking in forward scan sonar images using the PHD filter. IEEE Proc. Radar Sonar Navig. 2005;152(5):327–334, doi:10.1049/ip-rsn:20045068.

[11] Carroll PJ, Lathrop JD, McCormick JF, Summey DC. Mobile underwater debris survey system (MUDSS). In: IGARSS '98 Sensing and Managing the Environment 1998 IEEE International Geoscience and Remote Sensing Symposium Proceedings (Cat No98CH36174). IEEE; 1998, doi:10.1109/IGARSS.1998.699528.

[12] Chen Z, Sun Y, Gu Y, Wang H, Qian H, Zheng H. Underwater object segmentation integrating transmission and saliency features. IEEE Access. 2019;7:72420–30, doi:10.1109/access.2019.2919711.

[13] Colreavy-Donnelly S, Caraffini F, Kuhn S, Gongora M, Florez-Lozano J, Parra C. Shallow buried improvised explosive device detection via convolutional neural networks. Integr Comput Aided Eng. 2020;27(4):403–416, doi:10.3233/ica-200638.

[14] CSIRO. New sensor detects bombs on sea floor [Internet]. CSIROscope. 2012 [cited 2022 Jan 20]. Available from: https://blog.csiro.au/new-sensor-detects-bombs-on-sea-floor/.

[15] Dzieciuch I, Gebhardt D, Barngrover C, Parikh K. Non-linear convolutional neural network for automatic detection of mine-like objects in sonar imagery. In: Lecture Notes in Networks and Systems. Cham: Springer International Publishing; 2017. p. 309–14.

[16] Dura E, Zhang Y, Liao X, Dobeck GJ, Carin L. Active learning for detection of mine-like objects in side-

scan sonar imagery. IEEE J Ocean Eng. 2005;30(2):360–71, doi:10.1109/joe.2005.850931.

[17] Abdul Ghani AS, Mat Isa NA. Underwater image quality enhancement through composition of dual-intensity images and Rayleigh-stretching. Springerplus. 2014;3(1):757, doi:10.1186/2193-1801-3-757.

[18] Hall JJ, Azimi-Sadjadi MR, Kargl SG, Zhao Y, Williams KL. Underwater unexploded ordnance (UXO) classification using a matched subspace classifier with adaptive dictionaries. IEEE J Ocean Eng. 2019;44(3):739–52, doi:10.1109/joe.2018.2835538.

[19] Hamreras S, Boucheham B, Molina-Cabello MA, Benıtez-Rochel R, Lopez-Rubio E. Content-based image retrieval by ensembles of deep learning object classifiers. Integrated Computer-Aided Engineering. 2020;27(3):317–31.

[20] Islam MJ, Xia Y, Sattar J. Fast underwater image enhancement for improved visual perception, CoRR. CoRR; 2019.

[21] JXD official online store [Internet]. Jxdofficial.com. [cited 2022 Jan 20]. Available from: http://www.jxdofficial.com/.

[22] Li C, Guo C, Ren W, Cong R, Hou J, Kwong S, et al. An underwater image enhancement benchmark dataset and beyond, CoRR. CoRR; 2019.

[23] Li Y, Zhang Y, Xu X, He L, Serikawa S, Kim H. Dust removal from high turbid underwater images using convolutional neural networks. Opt Laser Technol [Internet]. 2019;110:2–6, doi:10.1016/j.optlastec.2017.09.017.

[24] Gao S-B, Zhang M, Zhao Q, Zhang X-S, Li Y-J. Underwater image enhancement using adaptive retinal mechanisms. IEEE Trans Image Process. 2019;28(11):5580–5595, doi:10.1109/TIP.2019.2919947.

[25] Luo C, Yu L, Yan J, Li Z, Ren P, Bai X, et al. Autonomous detection of damage to multiple steel surfaces from 360° panoramas using deep neural networks. Comput-aided civ infrastruct eng. 2021;36(12):1585–1599, doi:10.1111/mice.12686.

[26] Mishra P, Piciarelli C, Foresti GL. A neural network for image anomaly detection with deep pyramidal representations and dynamic routing. Int J Neural Syst. 2020;30(10):2050060, doi:10.1142/s0129065720500604.

[27] Pezeshki A, Azimi-Sadjadi MR, Scharf LL, Robinson M. Underwater target classification using canonical correlations. In: Oceans 2003 Celebrating the Past. Teaming Toward the Future (IEEE Cat No03CH37492). IEEE; 2003;4:1906–1911.

[28] Perry SW, Guan L. Pulse-length-tolerant features and detectors for sector-scan sonar imagery. IEEE J Ocean Eng. 2004;29(1):138–56, doi:10.1109/joe.2003.819312.

[29] Qin H, Li X, Liang J, Peng Y, Zhang C. DeepFish: Accurate underwater live fish recognition with a deep architecture. Neurocomputing. 2016;187:49–58, doi:10.1016/j.neucom.2015.10.122.

[30] Rodner E, Simon M, Fisher R, Denzler J. Fine-grained recognition in the noisy wild: Sensitivity analysis of convolutional neural networks approaches. In: Procedings of the British Machine Vision Conference 2016. British Machine Vision Association; 2016.

[31] SERDP/Office of Naval Research. Workshop on Acoustic Detection and Classification of UXO in the Underwater Environment. Final Report, U.S. Department of Defense; 2013.

[32] SERDP/Office of Naval Research. Workshop on Acoustic Detection and Classification of Munitions in the Underwater Environment. Final Report, U.S. Department of Defense; 2018.

[33] Sarmadi H, Yuen K-V. Early damage detection by an innovative unsupervised learning method based on kernel null space and peak-over-threshold. Comput-aided civ infrastruct eng. 2021;36(9):1150–1167, doi:10.1111/mice.12635.

[34] Shen J, Yan W, Li P, Xiong X. Deep learning-based object identification with instance segmentation and pseudo-LiDAR point cloud for work zone safety. Comput-aided civ infrastruct eng. 2021;36(12):1549–1567, doi:10.1111/mice.12749.

[35] Shin W, Bu S-J, Cho S-B. 3D-convolutional neural network with generative adversarial network and autoencoder for robust anomaly detection in video surveillance. Int J Neural Syst. 2020;30(6):2050034, doi:10.1142/S0129065720500343.

[36] Sun X, Shi J, Liu L, Dong J, Plant C, Wang X, et al. Transferring deep knowledge for object recognition in Low-quality underwater videos. Neurocomputing. 2018;275:897–908, doi:10.1016/j.neucom.2017.09.044.

[37] Sudac D, Valkovic V, Nad K, Obhodas J. The underwater detection of TNT explosive. IEEE Trans Nucl Sci. 2011;58(2):547–551, doi:10.1109/tns.2011.2112671.

[38] Thompson B, Cartmill J, Azimi-Sadjadi MR, Schock SG. A multichannel canonical correlation analysis feature extraction with application to buried underwater target classification. In: The 2006 IEEE International Joint Conference on Neural Network Proceedings. IEEE; 2006, p. 4413–4420.

[39] Sofar Trident Underwater Drone [YouTube link]. San Francisco: Sofar Ocean; 2019. [updated 2019 Mar 19; cited 2022 Jan 20]. Available from: https://www.youtube.com/watch?v=iX7zVPX3oBs.

[40] UXO Photo Gallery [Internet]. Uxoinfo.com. [cited 2022 Jan 20]. Available from: http://uxoinfo.com/blogcfc/client/includes/uxopages/Photo-Gallery-VLB.cfm.

[41] Williams DP. The Mondrian detection algorithm for sonar imagery. IEEE Trans Geosci Remote Sens. 2018;56(2):1091–102, doi: 10.1109/tgrs.2017.2758808.

[42] Wang X, Ouyang J, Li D, Zhang G. Underwater object recognition based on deep encoding-decoding network. J Ocean Univ China. 2019;18(2):376–82, doi:10.1007/s11802-019-3858-x.

[43] Zhu J, Yu S, Han Z, Tang Y, Wu C. Underwater object recognition using transformable template matching based on prior knowledge. Math Probl Eng. 2019;2019:1–11, doi:10.1155/2019/2892975.

[44] UXO dataset of images [cited 2022 Jan 20]. Available from: https://lambda-iot.uniud.it/uxo_dataset.

Appendix A. Technical Specifications of Hardware

The following tables report some technical data about the Trident mini-ROV and the two cameras used in our experiments. Data are taken from the websites of the respective producers.

External dimensions [L X W X H]	410 mm x 205 mm x 86 mm
Weight (ballasted for freshwater)	3.4 kg
Weight (ballasted for seawater)	3.5 kg
Depth rating	100 m
Thrusters	3 brushless motors
Maximum speed	2 m/s
Battery architecture	3S4P Li-NMC 18650 cells with built-in PCM
Capacity	95 WH
Charge time	1.5 hr from 20% to 80%, 3 hr from 0% to 100%
Nominal run time	3-4 hours, normal operation
Power consumption	30 W nominal
Charge power requirements	120VAC to 240VAC
Topside WiFi interface	802.11 b/g/n
IMU	3-axis magnetometer, 3-axis gyro, 3-axis accelerometer
Depth/temperature	1cm-resolution depth sensor with temperature calibration and display
Other sensors	Internal barometer, battery meter
System requirements	Android (min. 5.1) device through the OpenRov App.
Vehicle payload data interface	Wi-Fi
Operating water temperature rating	-2°C to 40°C
Storage temperature	0°C to 25°C
Chemical resistance	Seawater, diluted Chlorine

Table 7
Trident Technical Specifications.

Resolution	1080p @ 30fps recorded, 720p @ 30fps live on device
Video Latency	∼120 ms
Video compression	H.264
Video export format	MP4
Features	High color rendition and dynamic range, optimized for low-light underwater, wide angle field-of-view and scratch-proof sapphire window. 100m Waterproof

Table 8
Onboard Camera Technical Specifications.

Resolution	4K @ 60fps, 2.7K1 @ 20fps, and 1080p @ 240fps. Capture 12MP up to 30 fps
Video Latency	∼16.7 ms
Video compression	H.264
Video export format	MP4
Field of View (FOV)	https://gopro.com/help/articles/question_answer/hero7-field-of-view-fov-information?sf96748270=1
Features	HyperSmooth Video Stabilization, SuperPhoto Auto HDR Photo Enhancement, 10m Waterproof without a Housing, 60m Waterproof with Housing

Table 9
External Camera (GoPro Hero 7 Black) Technical Specifications.

Appendix B. Multiclass underwater classification data

The following table reports the confusion matrix of the multiclass classification experiment described at the end of Section 5:

	UXO	Coral	Sea Turtle	Pufferfish	Torpedo	Sea Anemone	Anemone Fish	Rock Beauty
Pred. UXO	185	1	1	2	1	0	0	1
Pred. Coral	8	173	1	2	5	4	1	0
Pred. Sea Turtle	5	0	178	6	2	0	0	0
Pred. Pufferfish	0	4	12	183	9	0	0	0
Pred. Torpedo	1	2	6	3	178	0	0	0
Pred. Sea Anemone	0	19	0	2	1	171	18	0
Pred. Anemone Fish	0	0	0	2	0	24	177	0
Pred. Rock Beauty	1	1	2	0	4	1	4	199