# Video Image Enhancement and Machine Learning Pipeline for Underwater Animal Detection and Classification at Cabled Observatories

**Vanesa Lopez-Vazquez** [1,*], **Jose Manuel Lopez-Guede** [2], **Simone Marini** [3,4],
**Emanuela Fanelli** [4,5], **Espen Johnsen** [6] **and Jacopo Aguzzi** [4,7]

[1] DS Labs, R+D+I unit of Deusto Sistemas S.A., 01015 Vitoria-Gasteiz, Spain
[2] Department of System Engineering and Automation Control, Faculty of Engineering of Vitoria-Gasteiz, University of the Basque Country (UPV/EHU), Nieves Cano, 12, 01006 Vitoria-Gasteiz, Spain; jm.lopez@ehu.es
[3] Institute of Marine Sciences, National Research Council of Italy (CNR), 19032 La Spezia, Italy; simone.marini@sp.ismar.cnr.it
[4] Stazione Zoologica Anton Dohrn (SZN), 80122 Naples, Italy; e.fanelli@univpm.it (E.F.); jaguzzi@icm.csic.es (J.A.)
[5] Department of Life and Environmental Sciences, Polytechnic University of Marche, Via Brecce Bianche, 60131 Ancona, Italy
[6] Institute of Marine Research, PO Box 1870, 5817 Bergen, Norway; espen.johnsen@hi.no
[7] Instituto de Ciencias del Mar (ICM) of the Consejo Superior de Investigaciones Científicas (CSIC), 08003 Barcelona, Spain
*   Correspondence: vlopez@deustosistemas.net; Tel.: +34-618-042-913

check for updates

**Abstract:** An understanding of marine ecosystems and their biodiversity is relevant to sustainable use of the goods and services they offer. Since marine areas host complex ecosystems, it is important to develop spatially widespread monitoring networks capable of providing large amounts of multiparametric information, encompassing both biotic and abiotic variables, and describing the ecological dynamics of the observed species. In this context, imaging devices are valuable tools that complement other biological and oceanographic monitoring devices. Nevertheless, large amounts of images or movies cannot all be manually processed, and autonomous routines for recognizing the relevant content, classification, and tagging are urgently needed. In this work, we propose a pipeline for the analysis of visual data that integrates video/image annotation tools for defining, training, and validation of datasets with video/image enhancement and machine and deep learning approaches. Such a pipeline is required to achieve good performance in the recognition and classification tasks of mobile and sessile megafauna, in order to obtain integrated information on spatial distribution and temporal dynamics. A prototype implementation of the analysis pipeline is provided in the context of deep-sea videos taken by one of the fixed cameras at the LoVe Ocean Observatory network of Lofoten Islands (Norway) at 260 m depth, in the Barents Sea, which has shown good classification results on an independent test dataset with an accuracy value of 76.18% and an area under the curve (AUC) value of 87.59%.

**Keywords:** cabled observatories; artificial intelligence; deep learning; machine learning; deep-sea fauna

## 1. Introduction

### 1.1. The Development of Marine Imaging

Over the past couple of decades, imaging of ocean biodiversity has experienced a spectacular increase [1], revolutionizing the monitoring of marine communities at all depths of the continental margins and the deep sea [2]. At the same time, a relevant development has taken place in robotic platforms bearing different types of imaging devices in association with a diversified set of environmental sensors [3]. Among these assets there are cabled observatories as video-functioning multiparametric platforms connected to shore by fiber optic cables [4]. These fixed platforms are being used to acquire image material from which animals of different species can be identified and then counted in a remote fashion, at a high frequency and over consecutive years [5–9]. Then, extracted biological time series are used to estimate how populations and species respond to the changes in environmental conditions (also concomitantly measured) [10–14].

### 1.2. The Human Bottleneck in Image Manual Processing

Although a large amount of scientific literature has been produced in recent years about underwater content-based image analysis [15–19], the processing of video data within ecological application contexts is still mostly manual and cabled observatory platforms and their networks are not yet equipped with permanent software tools for the automated recognition and classification of biological relevant image content [3,20]. In this context, cabled observatory networks such as Ocean Network Canada (ONC, www.oceannetworks.ca), European Multidisciplinary Seafloor and water-column Observatories (EMSO, http://emso.eu/), and Lofoten-Vesterålen (LoVe, https://love.statoil.com/) among others, are missing the opportunity to increase their societal impact by enforcing service-oriented image acquisition and automatically processing target species of commercial relevance.

At the same time, the development of artificial intelligence (AI) oriented to image treatment for animal counting and classification requires the development of analysis tools for ecological annotation, as well as semantic infrastructures for combining datasets and recognition and classification algorithms [21]. To do this, it is necessary to create relevant and accessible ecological repositories (e.g., Fish4Knowledge, http://groups.inf.ed.ac.uk/f4k/ and SeaCLEF, https://www.imageclef.org/) and, in them, to define effective ground-truth datasets for an effective classification [6].

### 1.3. Objectives and Findings

The cabled Lofoten-Vesterålen (LoVe) observatory network is located in the Norwegian deep sea, in the Norwegian continental slope of Lofoten Islands, in the Barents sea at 260 m depth, an area hosting one of the highest abundance cold-water coral (CWC) reefs in the world [22–24]. CWCs host a rich associated fauna [25], especially fish, with several species of high commercial value for the local fishery, such as the rockfish *Sebastes* spp. [26]. Presently, the monitoring of its local population and other species in the surrounding community has not yet been undertaken, which is relevant to the production of some ancillary data for fishery-independent and ecosystem-based management models (e.g., how fish respond to other species or oceanographic variations and how this is reflected in commercial availability).

Within the envisaged development of the LoVe observatory network, aiming to establish a science-based infrastructure for continuous online monitoring of the ocean interior including benthic, pelagic, and the demersal habitats, we propose a user-friendly integrated library of tools (specifically developed for that cabled observatory network), aimed at the following: (i) The generation of ground-truth datasets through semi-automatic image annotation, (ii) the training of supervised underwater image classifiers based on ground-truth datasets, and (iii) the automated classification of underwater images acquired by cameras installed on fixed and mobile platforms. These tools support the video/image analysis of the LoVe still imaging outputs dedicated to the tracking and classification of different species of the local deep-sea community.

In order to explore and maintain the wide biodiversity and life of underwater ecosystems, monitoring and subsequent analysis of the information collected is necessary. Due to the numerous underwater images, as well as videos collected at sea, manual analysis becomes a long and tedious task, therefore, this study proposes a pipeline to solve this task automatically.

The objective of this study is to introduce a pipeline for underwater animal detection and classification, which includes image enhancing, image segmentation, and manual annotation (to define training and validation datasets), and automated content recognition and classification steps. This pipeline has demonstrated good results in the classification of animals of the Norwegian deep sea, reaching an accuracy value of 76.18% and an area under the curve (AUC) value of 87.59%.

The paper is organized as follows: Section 2 presents the dataset used in this work, and describes the processing pipeline and the experimental setup, within the chosen evaluation metrics; Section 3 shows the obtained results; while Section 4 introduces the discussion about the preliminary results; and finally, Section 5 presents our conclusions.

## 2. Materials and Methods

### 2.1. The Cabled Observatory Network Area

The LoVe observatory is located 20 km of the Lofoten Islands (Norway) in the Hola trough (Figure 1) and was deployed in 2013. This glacially deepened trough is 180 to 260 m deep and incises the continental shelf in a northwest to southeast direction from the continental slope to the coast. The location of the observatory (~260 m deep) is enclosed by two 100 m deep banks, Vesterålsgrunnen in the northeast and Eggagrunnen in the southwest. The trough has a diverse topography with sand wave fields of up to 7 m high, 10 to 35 m high ridges, and approximately 20 m high CWC mounds [27]. The CWC mounds are predominantly found in the southeastern part of the trough at a depth of ~260 m just south of the Vesterålsgrunnen bank, being mostly constituted by CWC *Desmophyllum pertusum* [28].
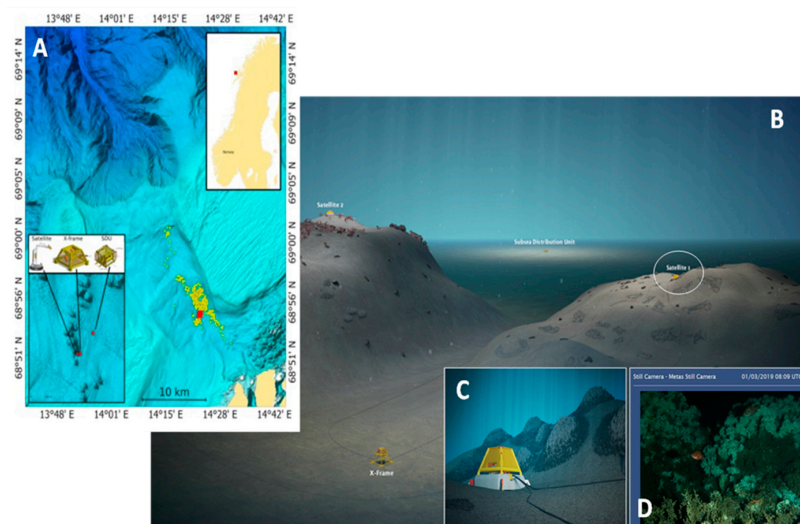


**Figure 1.** Overview of the study area where the Lofoten-Vesterålen (LoVe) observatory is located: (**A**) Bathymetric map of the canyon area showing (in red) the observatory area and (in yellow) relevant *Desmophyllum pertusum* reef mounds around it (adapted from [24]), (**B**) three-dimensional (3D) detailed representation of the area showing (encircled in white) the video node providing the footage used to train AI procedures, (**C**) enlarged view of the areas surrounding the node where *D. pertusum* reefs are schematized, and finally (**D**) the field of view as it appears in the analyzed footages (B, C, and D) taken from the observatory site at https://love.statoil.com/.

The following three platforms compose the data collection system of this area: The X-Frame, which measures water current and biomass in water (with an echosounder); Satellite 1, which collects multiple

types of data, such as photos, sound, chlorophyll, turbidity, pressure, temperature, conductivity, etc.; and Satellite 2, which only collects photos. The images used in this paper were acquired with Satellite 1 (see also Section 2.3).

## 2.2. The Target Group of Species

The area around the LoVe observatory is rich in biodiversity, and the following species were identified according to the local fauna guides [29–31] (Figure 1 and Table 1): *Sebastes* sp., *Lithodes maja*, *Sepiolidae*, *Bolocera tuediae*, *Pandalus* sp., *Echinus esculentus*, *Brosme brosme*, *Cancer pagurusa*, and *Desmophyllum pertusum*. Some other species were also targeted by our automated protocol but could not be classified and are generically categorized as "hermit" crab and "starfish".

**Table 1.** An example of video-detected species used for building the training dataset for reference at automated classification.

| Class (alias) | Species Name | # Specimens per Species in Dataset | Image in Figure 2 |
| --- | --- | --- | --- |
| Rockfish | *Sebastes* sp. | 205 | (A) |
| King crab | *Lithodes maja* | 170 | (B) |
| Squid | *Sepiolidae* | 96 | (C) |
| Starfish | *Unidentified* | 169 | (D) |
| Hermit crab | *Unidentified* | 184 | (E) |
| Anemone | *Bolocera tuediae* | 98 | (F) |
| Shrimp | *Pandalus* sp. | 154 | (G) |
| Sea urchin | *Echinus esculentus* | 138 | (H) |
| Eel like fish | *Brosme brosme* | 199 | (I) |
| Crab | *Cancer pagurus* | 102 | (J) |
| Coral | *Desmophyllum pertusum* | 142 | (K) |
| Turbidity | - | 176 | (L) |
| Shadow | - | 101 | (M) |
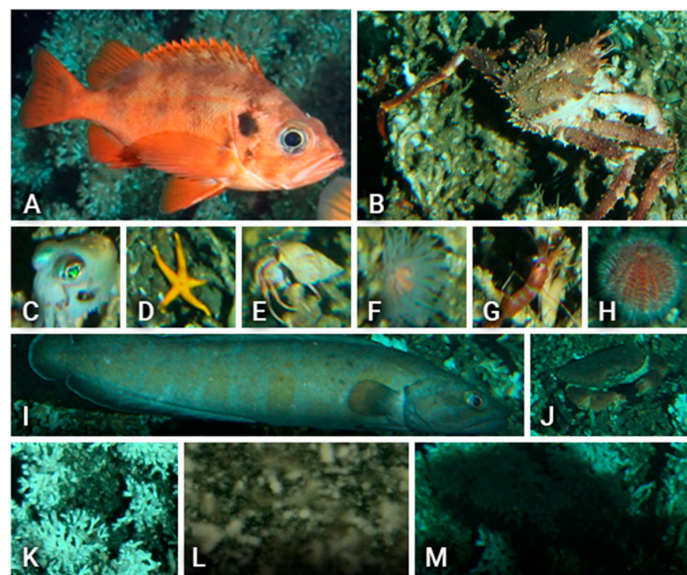


**Figure 2.** An example of video-detected species used for building the training dataset for reference at automated classification: (**A**) Rockfish (*Sebastes* sp.), (**B**) king crab (*Lithodes maja*), (**C**) squid (*Sepiolidae*), (**D**) starfish, (**E**) hermit crab, (**F**) anemone (*Bolocera tuediae*), (**G**) shrimp (Pandalus sp.), (**H**) sea urchin (*Echinus esculentus*), (**I**) eel-like fish (*Brosme brosme*), (**J**) crab (*Cancer pagurus*), (**K**) coral (*Desmophyllum pertusum*), and finally (**L**) turbidity, and (**M**) shadow.

Among these species, only *Sebastes* (Figure 2) has commercial importance. This is a genus of fish in the family Sebastidae, usually called Rockfish, encompassing 108 species, two of them (*Sebastes norvegicus* and *Sebastes mentella*) inhabiting Norwegian deep waters and presenting very

similar morphological characteristics including coloring [32]. *Sebastes norvegicus* has been reported in LoVe *Desmophyllum* areas up to six times with higher density as compared with the surrounding seabed [25,29]. Accordingly, we referred to *Sebastes* sp. for all rockfish recorded at the LoVe observatory.

Another two elements were selected due to their abundance in the footage, turbidity, and shadows. The so-called "turbidity" class refers to the cloudiness sometimes seen in water containing sediments or phytoplankton, while the "shadow" class corresponds to the shadows cast by some of the fish.

### 2.3. Data Collection

The images used for testing the proposed tools were extracted from time-lapse footages (image acquisition period of 60 min) generated with photos obtained by the camera from Satellite 1 (see previous Section 2.1 and Figure 1) in two time windows, the first from 4 October 2017 to 27 June 2018 and the second from 10 December 2018 to 29 June 2019. Accordingly, a total of 8818 images were available continuously over the 24 h period during 372 consecutive days. Some images were missing due to the observatory structure maintenance.

### 2.4. Image Processing Pipeline for Underwater Animal Detection and Annotation

The images provided by LoVe observatory were acquired in an uncontrolled environment, characterized by a heterogeneous background of coral bushes, where turbidity and artificial lighting changes make it difficult to detect elements with heterogeneous shapes, colors, and sizes.

An image processing pipeline (Figure 3) was designed and developed based on computer vision tools for enhancing the image contrast and for segmenting relevant image subregions [19,33]. To speed up this process, the images were resized from $3456 \times 5184$ pixels to a quarter of their size, i.e., $964 \times 1296$ pixels.
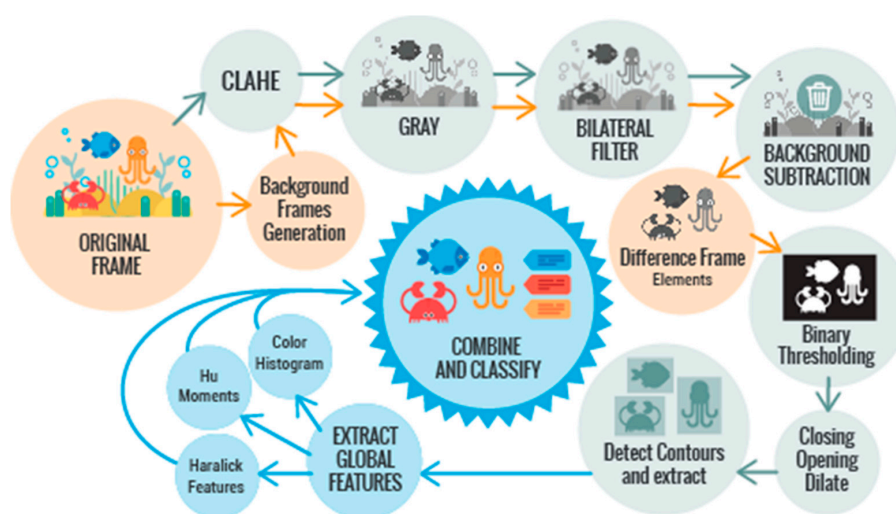


**Figure 3.** Image processing pipeline.

First, a background image was generated for each day, that is, obtaining the average of the 24 images for each 24 h. These images were used to perform the background subtraction after applying different techniques to the images.

The contrast limited adaptative histogram equalization (CLAHE) technique [34] was applied to enhancing the image background/foreground contrast. While the traditional adaptive histogram equalization [35] is likely to amplify noise in constant or homogeneous regions, the CLAHE approach reduces this problem by limiting the contrast amplification using a filtering technique [36–38]. After this equalization, a bilateral filtering [39] was applied in order to discard irrelevant image information while preserving the edges of the objects that are to be extracted.

The background subtraction took place at this time. In this way, a frame was obtained with only the elements detected in the original image.

A binary thresholding value, which was chosen by testing different values, was performed to obtain the mask of the elements in the image [19,33,40–43] and different morphological transformations such as closing, opening, and dilation were applied to remove noise.

Global features were extracted for subsequent classification, which is explained later.

Finally, the contours of the threshold image were detected in order to identify the relevant elements in the input image. The whole process was carried out with Python, OpenCV [44], and Mahotas [45].

As the size of the collected set was only a total of 1934 elements, we decided, first, to apply data augmentation techniques to 80% of the images (a total of 1547 images), which are the ones that made up the training set.

Data augmentation involves different techniques in order to generate multiple images from an original one to increase the size of the training set. Within this work, several image transformations were used such as image flipping, rotation, brightness changes, and finally zoom. After applying data augmentation techniques, the training set increased from 1547 to 39,072 images.

Because global features such as texture or color features have obtained good results in the classification task in the literature [46–48], we extracted and combined several global features from all images, which are summarized in Table 2.

**Table 2.** Extracted global features from each image.

| Type | Description | Obtained Features |
| --- | --- | --- |
| Hu invariant moments [49] | They are used for shape matching, as they are invariant to image transformations such as scale, translation, rotation, and reflection. | An array containing the image moments |
| Haralick texture features [50] | They describe an image based on texture, quantifying the gray tone intensity of pixels that are next to each other in space. | An array containing the Haralick features of the image |
| Color histogram [35,51] | The representation of the distribution of colors contained in an image. | An array (a flattened matrix to one dimension) containing the histogram of the image |

For the classification part, several algorithms were compared with each other to clarify which one obtained a more accurate classification result. Traditional classifiers such as support vector machine (SVM), k-nearest neighbors (K-NN), or random forests (RF) have been widely used for underwater animal classification. For example, in [52], the authors made a comparison between many classical algorithms obtaining an accuracy value higher than 0.7. Another study reached 82% of correct classification rate (CCR) with a SVM [53]. In recent years, deep learning (DL) approaches [54] have gained popularity due to their advantages, as they do not need the input data to be processed and often they get better results for problems related to image quality, language, etc. [23]. Accordingly, we decided to make a comparison between both types of methods; evaluating the results and performance of four classical algorithms and two different neural networks.

SVM is a supervised learning approach that can perform both linear or nonlinear classification or regression tasks [55–57] and has shown good results in the classification of underwater image features [58,59].

K-NN is a fast algorithm that classifies an object by a majority vote of its $k$ (a positive integer) nearest neighbors [60], being a recurrent classifier used in this domain [40,53].

Decision trees (DTs) are algorithms that perform both classification and regression tasks, in addition, they use a tree structure to make decisions [61,62]; each middle leaf (called node) of the tree represents an attribute, the branches are the decisions to be made (by rules), and each leaf of the tree

that is a final node, corresponds to a result. This kind of classifier is also popular in underwater animal classification, thus, the obtained results are quite good [63,64].

RF is an ensemble of DTs [65,66]. It normally applies the technique of bootstrap (also called bagging) at training. It uses averaging of the DT results to improve the predictive accuracy and to avoid over-fitting. Although RF have not been used as much as other algorithms, they have shown their performance and results [67].

Convolutional neural networks (CNNs or ConvNets) have shown good accuracy results solving underwater classification problems [68–70]. Deep neural networks (DNNs) have also been used successfully in this field [71].

Different structures, training parameters, and optimizers were chosen in order to make a comparison between them and determine which of the combinations obtained the best results. This is described in the next section.

## 2.5. Experimental Setup

Two versions of SVM were selected. The first one is a linear SVM (LSVM) with C = 1, which is a parameter used to determine the margin that separates the hyperplane for classification and influences the objective function. The second one is also a LSVM but with stochastic gradient descent (SGD) [72] training, which is an iterative method commonly used for optimizing, with hinge as the loss function and elasticnet [73] as the regularization term.

Regarding the K-NN algorithm, for the selection of the *k*-value, three criteria were considered as follows:

The selected number is preferable to be odd in order to avoid confusion between two classes;

The *k* value should be accurately selected, since small values could lead to overfitting and large values would make the algorithm computationally expensive [74,75];

The approximation used to set the *k*-value was the result of calculating the square root of the total number of samples in the training set [74], following Equation (1)

$$k = \sqrt{\# \, samples \, training \, set}. \tag{1}$$

Using the last criteria, two K-NN classifiers were tested, one with $k = 39$ and the other $k = 99$.

As was explained in the previous section, DTs have gained popularity and two DTs were chosen. For the proposed analysis, the selected number of nodes between the root and the leaves, was 3000 and 100,000.

Regarding RFs, two different RFs were selected, each with different parameters. The first one with 75 trees, 300 nodes, and 10 features to consider when performing the splitting; the second one with 50 trees, 1000 nodes, and 50 features.

The implementation of all the classical algorithms used are within the Scikit-learn library [76] (https://scikit-learn.org).

In the case of the DL approach, we selected four CNNs and four DNNs.

Two different structures were selected for the four CNNs. The first structure (CNN-1 and CNN-3) was composed of two blocks of convolution, activation, and pooling layers, while the second one (CNN-2 and CNN-4) contained three blocks. The activation function selected was rectified linear unit (ReLU), which is a commonly used function with CNNs. The four models have fully connected layers at the end, with an activation layer bearing a softmax function, which is a categorical classifier widely used in DL architectures [68]. For training, two different optimizers were selected. For the CNN-1 and CNN-2, Adadelta [77] was used and for the second group, CNN-3 and CNN-4, RMSProp was used [78]. The training parameters, such as epochs and batch size, were established on the basis of initial tests in which it was observed that Networks 1 and 2 (which have the optimizer in common) reached high accuracy values in the early epochs, while CNN-3 and CNN-4 took longer to improve their accuracy. In this way, for CNN-1 and CNN-2 the number of epochs was 50 and the batch size was

356. For the other two networks, CNN-3 and CNN-4, the number of the epochs was 150 and the batch size was decreased to 128.

The DNNs models have a similar layer structure. Similar to the previous network groups, the first structure (corresponding to DNN-1 and DNN-3) contains an input layer followed by three dense layers, each one followed by one activation layer. The first activation layer contains a ReLU function, whereas the others have a hyperbolic tangent function (tanh). Even this function is not as common as ReLU because it can cause training difficulties, it has obtained good results with some optimizers such as SGD [79]. These layers are followed by a dropout layer to prevent overfitting [80]. The second structure (for DNN-2 and DNN-4) is basically the same as the previous one but has one layer more and the activation function for each layer is the ReLU function. This time, RMSPprop and SGD were selected as the optimizers. As DNNs can be trained faster than the CNNs, the number of epochs selected was 500 for all DNNs, while the batch size was 518 for DNN-1 and DNN-2 and 356 for DNN-3 and DNN-4. A summary of the experimental setup of the DL models is shown in Table 3.

**Table 3.** Summary of the experimental setup of the different neural networks.

| CNN-1 | CNN-2 | CNN-3 | CNN-4 |
| --- | --- | --- | --- |
| Structure 1 | Structure 2 | Structure 1 | Structure 2 |
| Optimizer 1 | Optimizer 1 | Optimizer 2 | Optimizer 2 |
| Parameters 1 | Parameters 1 | Parameters 2 | Parameters 2 |
| **DNN-1** | **DNN-2** | **DNN-3** | **DNN-4** |
| Structure 1 | Structure 2 | Structure 1 | Structure 2 |
| Optimizer 1 | Optimizer 1 | Optimizer 2 | Optimizer 2 |
| Parameters 1 | Parameters 1 | Parameters 2 | Parameters 2 |

Each one of the networks was fed with the extracted global features from each element of the training dataset. These features were stacked together in a one-dimensional (1D) array. The output of each of the networks is one of the 13 classes defined in Table 1.

The environment used for training the selected algorithms and the defined models was Google Colaboratory (also known as Colab). Colab operates currently under Ubuntu 18.04 (64 bits) and it is provided by an Intel Xeon processor and 13 GB RAM. It is also provided with a NVIDIA Tesla K80 GPU. Traditional algorithms were trained on CPU, while deep learning models were trained on GPU.

*2.6. Metrics*

On the basis of the choices made by some studies in the literature of similar scope [47,76], every classifier was validated by 10-fold cross-validation by considering that the elements of each class were distributed evenly in each one of the folds. The performance of the models was evaluated by the accuracy, loss, and area under the curve (AUC) average scores [81].

The accuracy is given by Equation (2):

$$Accuracy = \frac{TP + TN}{P + N} = \frac{TP + TN}{TP + FP + TN + FN} \tag{2}$$

where *TP* is true positive, *TN* is true negative, *FP* is false positive, *FN* is false negative, *P* is real positives, and *N* is real negatives.

The AUC measures the area underneath the receiver operating characteristic (ROC) curve, as shown in Figure 4:
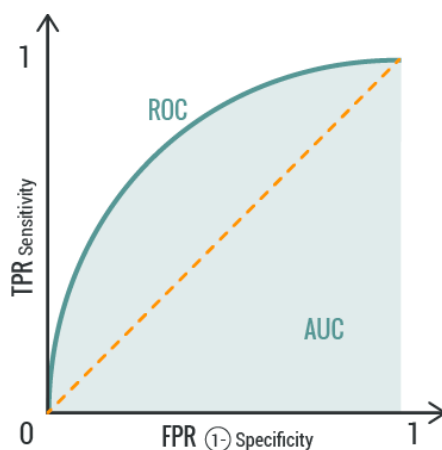
**Figure 4.** Receiver operating characteristic (ROC) curve.

The true rate positive (*TPR*) or sensitivity is given by Equation (3), while the false rate positive (*FPR*) or specificity is defined by Equation (4):

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN} \tag{3}$$

$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN} \tag{4}$$

The accuracy and AUC values were calculated by the macro called averaging technique, which calculates metrics for each label, without considering the label imbalance.

The loss function measures the difference between the prediction value and the real class. It is a positive value that increases as the robustness of the model decreases.

## 3. Results

Accuracy and AUC average values obtained for all classes and for each classifier were obtained performing cross-validation. The average training time is also shown in Table 4. The obtained confusion matrices of RF-2 and DNN-1 are summarized in Figures 5 and 6 respectively, while the remaining detailed results are found in the supplementary material (Appendix A).

**Table 4.** Accuracy and area under the curve (AUC) values with test dataset and training time obtained by different models.

| Type of Approach | Classifier | Accuracy | AUC | Training Time (h:mm:ss) |
|---|---|---|---|---|
| **Traditional classifiers** | Linear SVM | 0.5137 | 0.7392 | 0:01:11 |
| | LSVM + SGD | 0.4196 | 0.6887 | 0:00:28 |
| | K-NN ($k = 39$) | 0.4463 | 0.7140 | 0:00:02 |
| | K-NN ($k = 99$) | 0.3111 | 0.6390 | 0:00:02 |
| | DT-1 | 0.4310 | 0.6975 | 0:00:08 |
| | DT-2 | 0.4331 | 0.6985 | 0:00:08 |
| | RF-1 | 0.4326 | 0.6987 | 0:00:08 |
| | RF-2 | 0.6527 | 0.8210 | 0:00:08 |
| **DL** | CNN-1 | 0.6191 | 0.7983 | 0:01:26 |
| | CNN-2 | 0.6563 | 0.8180 | 0:01:53 |
| | CNN-3 | 0.6346 | 0.8067 | 0:07:23 |
| | CNN-4 | 0.6421 | 0.8107 | 0:08:18 |
| | DNN-1 | 0.7618 | 0.8759 | 0:07:56 |
| | DNN-2 | 0.7576 | 0.8730 | 0:08:27 |
| | DNN-3 | 0.6904 | 0.8361 | 0:06:50 |
| | DNN-4 | 0.7140 | 0.8503 | 0:07:16 |
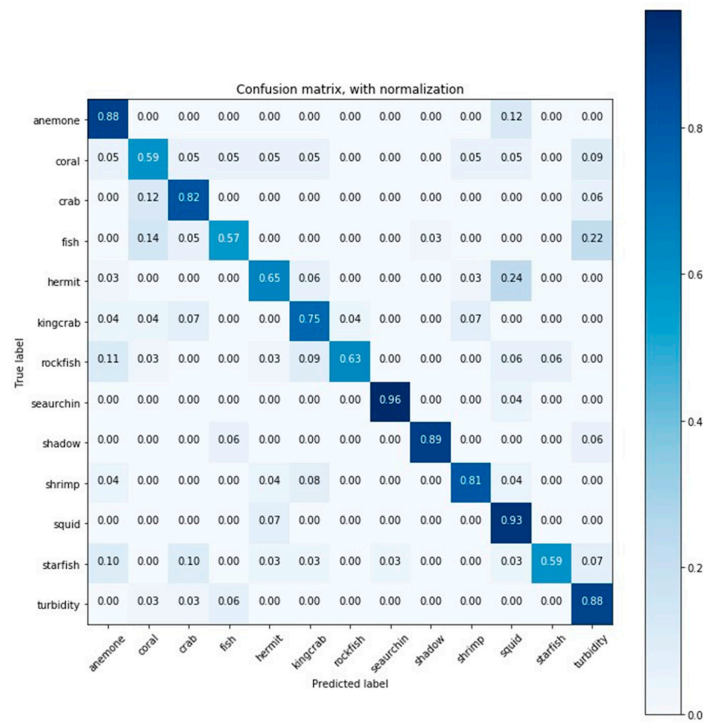
**Figure 5.** Confusion matrix for the classification results (accuracy) obtained by random forest (RF) RF-2.
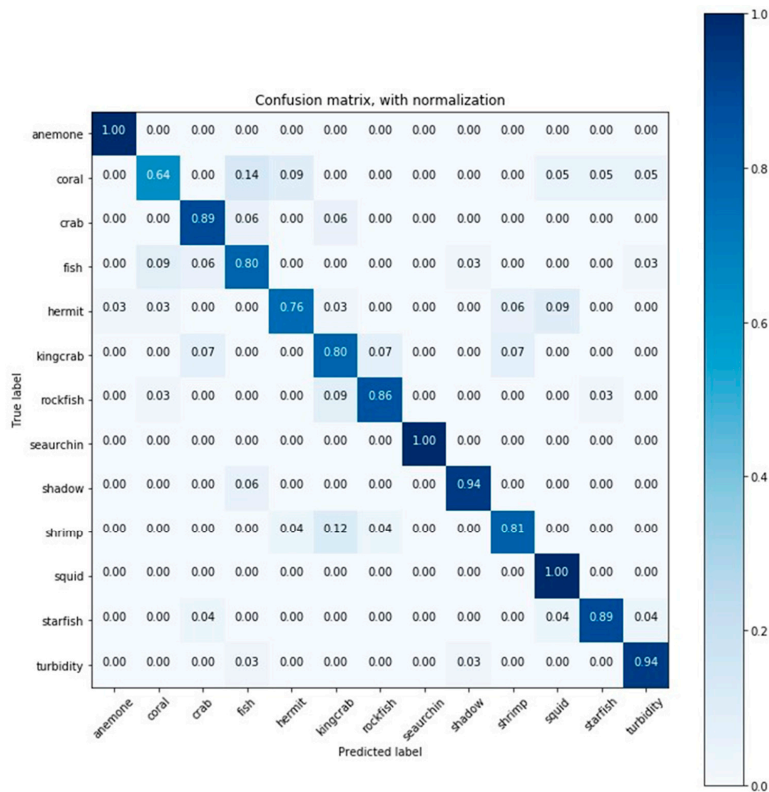


**Figure 6.** Confusion matrix for the classification results (accuracy) obtained by deep neural network (DNN) DNN-1.

Referring to traditional classifiers, the worst result was reached by K-NN with k = 99, as it barely reached an AUC value of 0.6390. However, the other K-NN ($k$ = 39) achieved better results, as it reached an AUC value of 0.7140. The two DTs and the RF-1 performed quite well, as they almost

achieved an AUC of 70%. The linear SVM reached an AUC of 0.7392 but also had the longest training time, which was 1 minute and 11 seconds. The SVM with the SGD optimization function did not work as well as the linear SVM, as it barely reached an AUC value of 0.6887. The RF-2 gained the highest AUC value, 0.8210, using a short training time of 8 s. The accuracy values are much lower for every classifier as compared with the AUC values.

The DL approaches obtained better results than almost every other traditional classifier. The eight networks obtained AUC values from 80% to 88%. CNNs achieved an AUC values between 0.7983 and 0.8180. The four DNNs obtained results between 0.8361 and 0.8759, respectively. Similar to the case of the accuracy values obtained by traditional classifiers, the accuracy achieved by DL approaches was also lower than AUC values. However, despite being lower values, all neural networks exceeded values of more than 60%; and most of the DNNs exceeded accuracy values of 70%.

The confusion matrix of Figure 5 corresponds to the results obtained by RF-2, were the X axis shows the predicted label and the Y axis shows the true label. For some classes, such as anemone, crab, sea urchin, shadow, shrimp, squid, and turbidity worked well, as it predicted values correctly between 70% to 93%. Coral, fish and starfish classes were misclassified by 59%, 57%, and 59%. Other classes such as hermit crab, king crab, and rockfish were also misclassified, but at least 60% of the elements were correctly classified.

Figure 6 shows the confusion matrix for the classification results obtained by DNN-1, which achieved good results for almost every class. In this case, three classes (anemone, sea urchin, and squid) were classified correctly at 100%, and the worst ranked class (coral) had 64% correctly labeled.

The performance of the four DNNs had different accuracy and loss values during the training, as shown in Figure 7a,b. The first two, which are the ones that obtained the best results, in the first 50 epochs, had already reached an accuracy value close to the final value (just over 0.60) and at the same time, the loss value also decreased to the final minimum value reached.
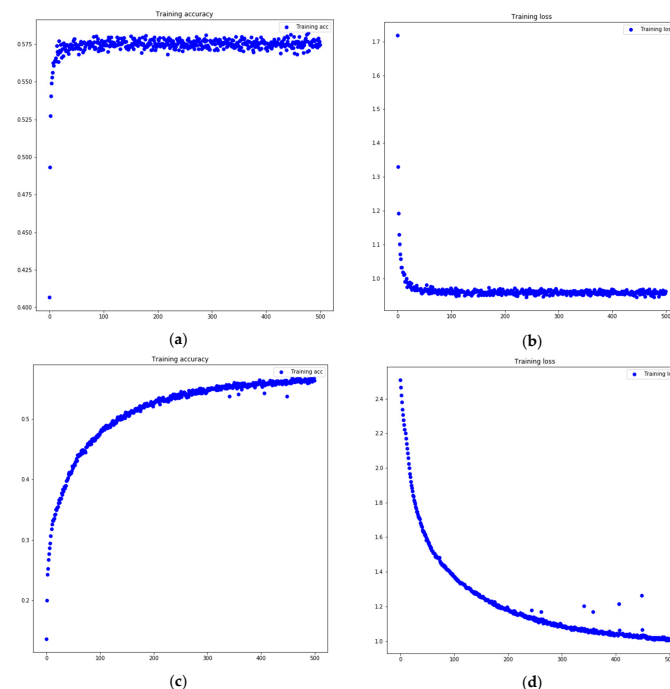


**Figure 7.** Training accuracy and loss plots of the DNNs with different structures. The X axis of all of plots shows the number of epochs, while the Y axis show the loss or accuracy value that reached the trained model. Training accuracy and loss plots of DNN-1: (**a**) Accuracy values obtained in every epoch at training time and (**b**) loss values obtained in every epoch at training time. Training accuracy and loss plots of DNN-4: (**c**) Accuracy values obtained in each epoch at training time and (**d**) loss values obtained in each epoch at training time.

However, both DNN-3 and DNN-4 took a longer amount of epochs to reach the highest accuracy value, as well as the lowest loss value, as shown in Figure 7c,d. As it progressively reached more optimal values, it did not reach the best values until at least 450 epochs.

DNN-1 was used to extract the time series of organism abundance, that is, it was used to detect, classify, and count animals in a short period of time in order to compare that result with the ground truth. This was performed on images not used during the training and test phase, corresponding to the period from 17 November 2017 to 22 November 2017.

Figure 8 shows three different time series for the rockfish, shrimp, and starfish during that period of six days, which covers 80 images. The classifier detected rockfish in 27 images, whereas with the manual detection, animals were detected in 24 images, which means that there are at least three false positives. In the other time series, the difference is much higher.

(**a**)

(**b**)

(**c**)
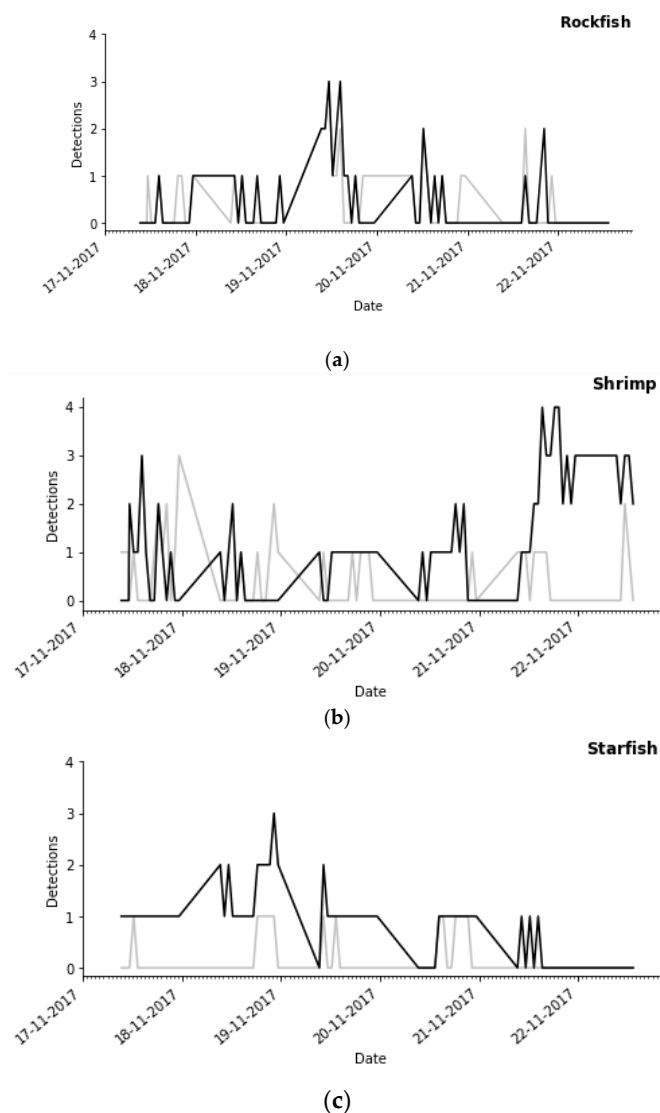
**Figure 8.** Time series of detections per day of (**a**) rockfish, (**b**) starfish, and (**c**) shrimp taxa. In the three plots, the X axis shows consecutive dates, while the Y axis shows the number of detections. The black lines correspond to the manual detection and the grey lines correspond to the estimated counts by the automatic process.

## 4. Discussion

In this study, we have presented a novel pipeline that can be used in an automatic pipeline for analysis of video image with the goal of identification and classification of organisms belonging to multiple taxa. The environment is difficult due to the turbidity that can sometimes be seen in the water, which makes it hard to appreciate the species; the small size of the dataset, which limits the appearance of some of the animals; the colors of the species detected, as well as the size of some of them, which sometimes blend in with the environment. All this can sometimes lead to incorrect classifications. Despite all this, we obtained successful classification results over the thirteen different taxa that we identified.

The image preprocessing pipeline automatically extracted 28,140 elements. Among them, between 90 and 200 specimens were manually selected from the 13 different classes of organisms (Table 1).

Two different types of methods were used in this study, i.e., classical algorithms and DL techniques. In general, the training phase for a DL approach needs hundreds of thousands of examples [82–84] or as an alternative, it can benefit from transfer learning approaches [85,86]. On the contrary, the proposed work uses only images acquired by the LoVe observatory with the aim of using the proposed image processing tools for incrementing the training set during the operational activities of the observatory.

Data augmentation was applied to the training dataset to obtain a richer one. The final training dataset consists of 39,072 images as follows: 2886 specimens of anemone, 3034 of coral, 3034 of crab, 3034 of fish, 3034 of hermit crab, 3034 of king crab, 3034 of rockfish, 3034 of sea urchin, 2997 of shadow, 3034 of shrimp, 2849 of squid, 3034 of starfish, and 3034 of turbidity. Similar studies also detected the advantages of DL over ML methods in marine environments [87–90].

With respect to the structures and training parameters chosen for all the networks, it can be seen that, for CNNs, the ones that obtained the best results were the CNN-2 and CNN-4, which had the same structure (the one with more layers) but different optimizers and parameters. However, in the case of DNNs, the DNN-1 and DNN-2 which share optimizer and parameters but not the same structure, obtained better results. Since the difference in results was not very large, it is necessary to perform more exhaustive experiments in order to conclude which element has the greatest influence on the results. In order to improve the pipeline and, consequently, the result, more work and in-depth study is needed.

As future work in this research line, the pipeline for the automated recognition and classification of image content introduced in this study should be permanently installed on the LoVe observatory augmented with the mobile platforms developed within the ARIM (Autonomous Robotic Sea-Floor Infrastructure for benthopelagic Monitoring) European project. The introduced pipeline could be used to notably increase the ground-truth training and validation dataset and obtain more accurate image classifiers. Within this application context, the development of neural networks could be further extended, creating models with different structures (adding and removing layers, modifying the number of units for each layer) and applying distinct parameter configuration (such as increasing or decreasing the number of epochs, batch size, and varying the chosen optimizer for training, or combining different activation functions). Other types of methods that have been proven to be successful should be considered, such as transfer learning approaches. Many studies have shown that the use of pretrained neural networks overcome results from non-pretrained neural networks [91,92]. This method is commonly used to improve the extraction of features and the classification when the training set is small [93,94], although this was not the case in this study, and it is less and less, because LoVe collects more images.

Changing the dataset would be challenging, as we could select images or videos with other characteristics, such as a moving background similar to [95], where they collected underwater sea videos using an ROV camera. Other possibilities include modifying the dataset, cropping images, or dividing fish into pieces to compare results, similar to [96].

Considering all the above, based on this work, we could make use of the transfer learning technique on a new network, and test it in other datasets.

## 5. Conclusions

The aim of this study was to design an automatic pipeline for underwater animal detection and classification, performing filtering and enhancing techniques, and using machine learning techniques. We obtained results with accuracy values of 76.18% and AUC of 87.59%, so the objective was achieved.

As can be seen in this study, our results reaffirm that unexplored underwater environments can be analyzed with the help of classic approaches and DL techniques. Moreover, DL approaches such as complex neural networks have shown that it is quite appropriate to identify and classify different elements, even if the images quality is sometimes low [74].

The improvement and enhancement of underwater images also plays an important role in detecting elements. It would be interesting to deepen in these methods, since a clear improvement of the images could reduce the later work of detection of features and obtain better classification rates.

The use of traditional classifiers and DL techniques aimed at the detection of marine species and, consequently, their assessment, both qualitative and quantitative, of the environment corresponding to each one, demonstrates that it can be an important advance in this field.

If we contemplate the advances in the acquisition of images and other parameters in different underwater ecosystems, it is easily deduced that the amount of information provided by the different acquisition centers would be impossible to analyze if it was not through this type of automatic technique.

## Appendix A

This appendix contains the rest of the confusion matrices of the results obtained on the test dataset from Table 4.
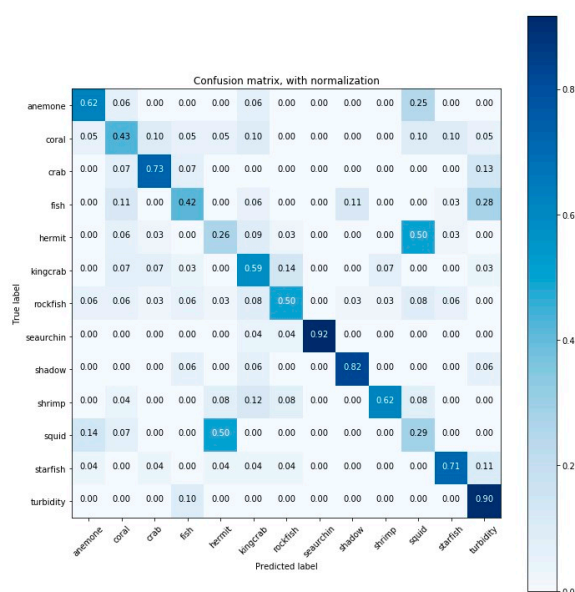


**Figure A1.** Confusion matrix for the classification results (accuracy) obtained by linear support vector machine (SVM).
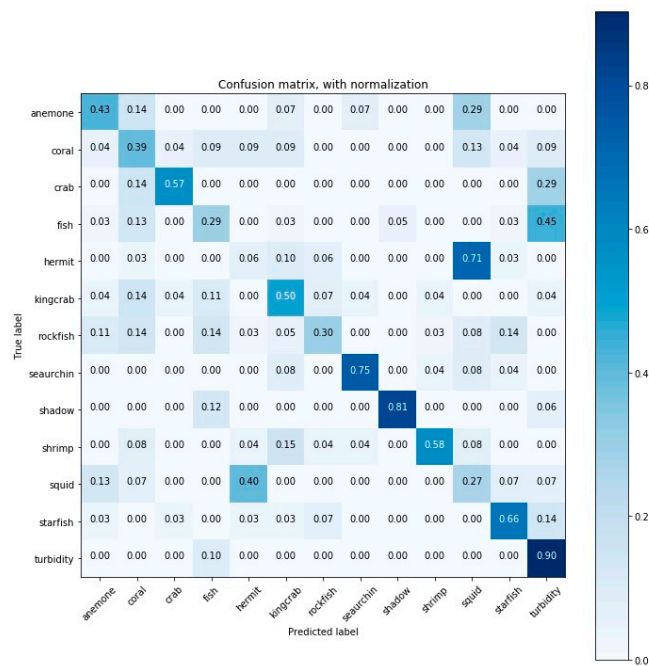
**Figure A2.** Confusion matrix for the classification results (accuracy) obtained by linear support vector machine and stochastic gradient descent (LSVM + SGD).
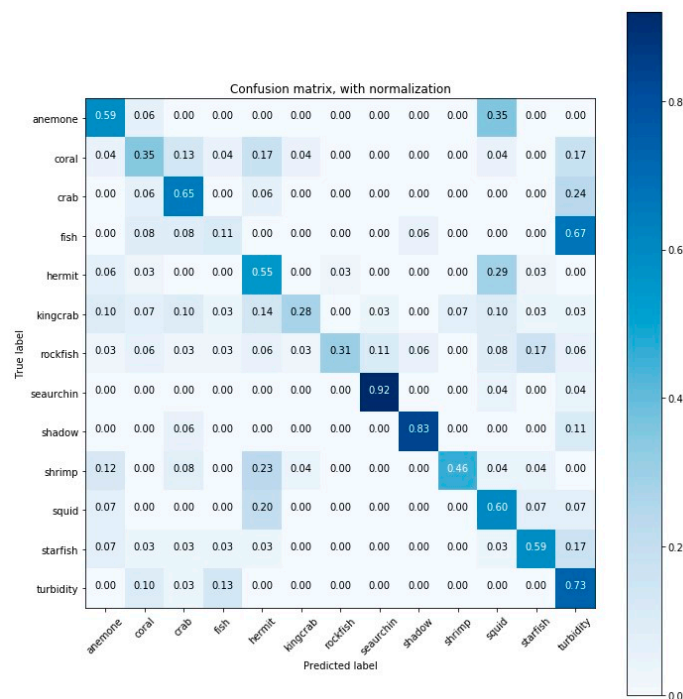


**Figure A3.** Confusion matrix for the classification results (accuracy) obtained by K-nearest neighbors (K-NN) (k = 39).
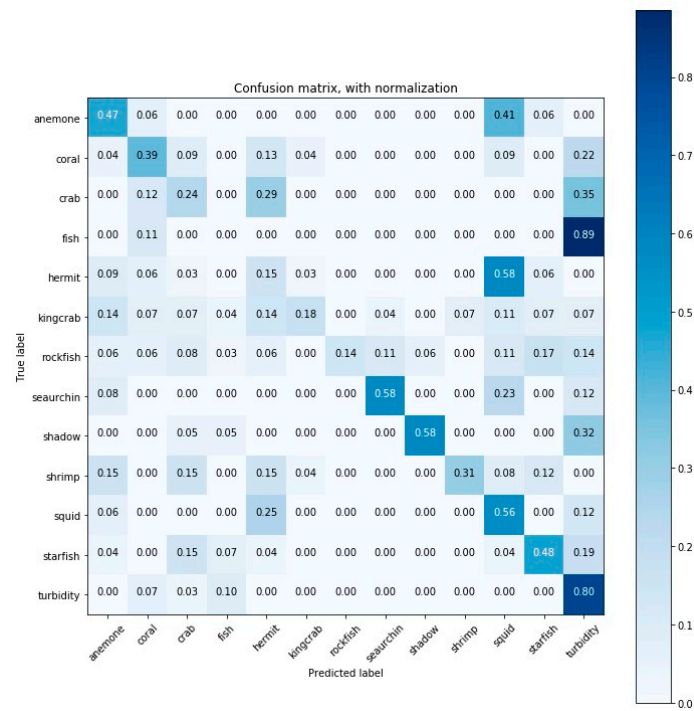
**Figure A4.** Confusion matrix for the classification results (accuracy) obtained by K-NN (k = 99).
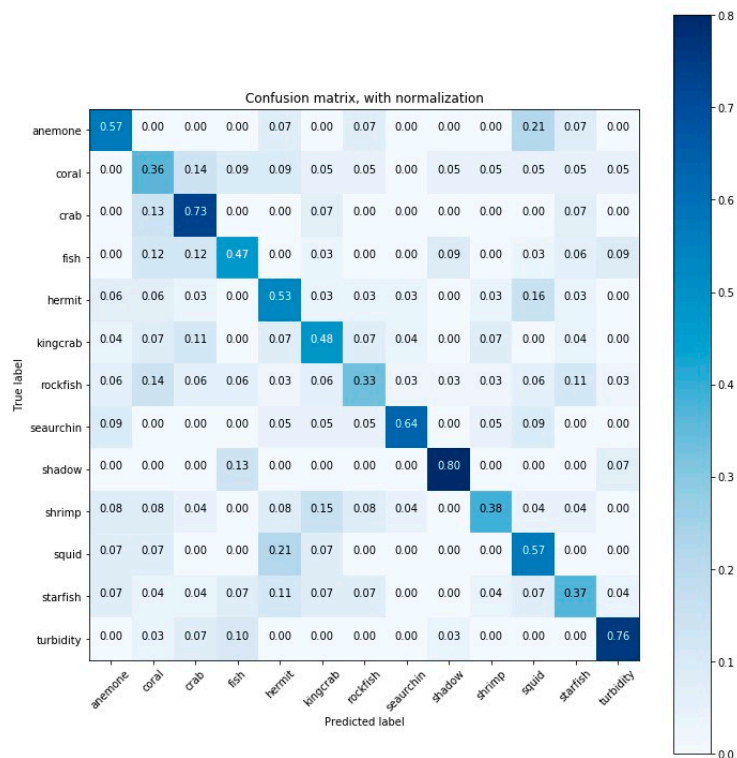


**Figure A5.** Confusion matrix for the classification results (accuracy) obtained by decision tree (DT) DT-1.
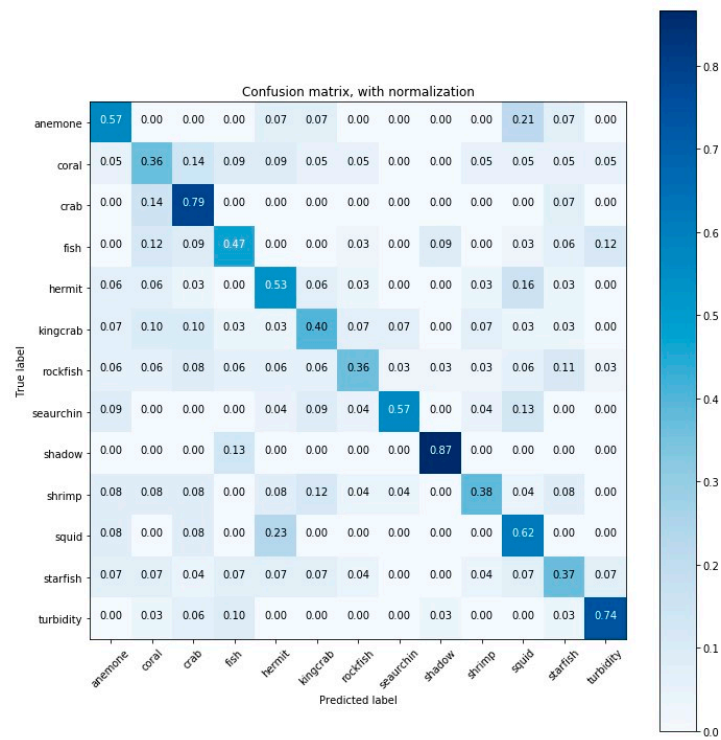
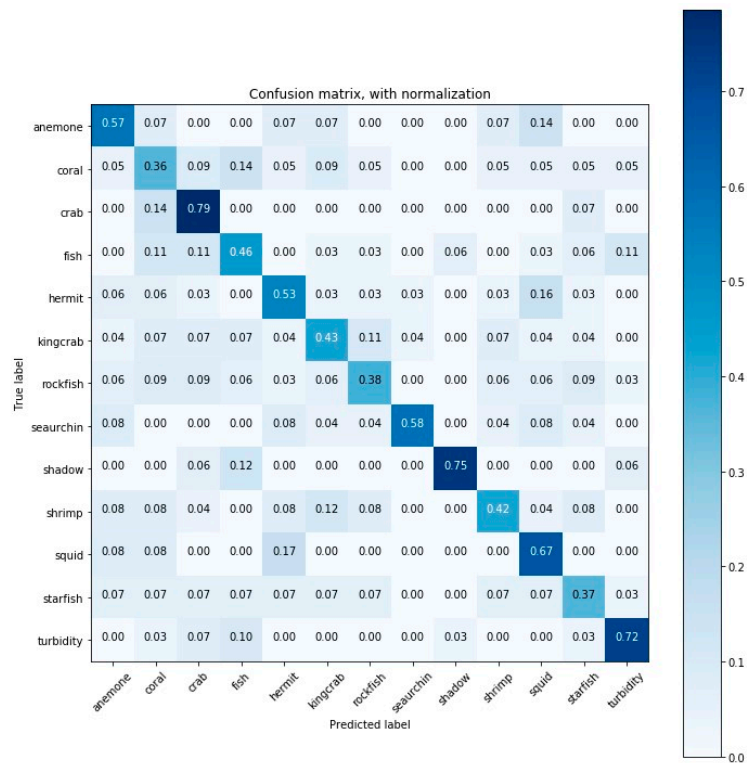**Figure A6.** Confusion matrix for the classification results (accuracy) obtained by DT-2.



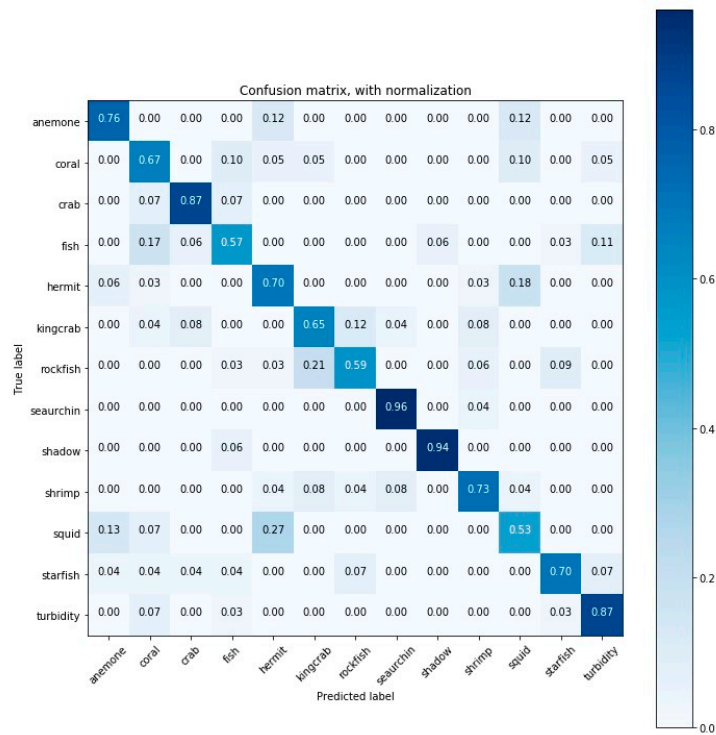**Figure A7.** Confusion matrix for the classification results (accuracy) obtained by RF-1.

**Figure A8.** Confusion matrix for the classification results (accuracy) obtained by convolutional neural network (CNN) CNN-1.



**Figure A9.** Confusion matrix for the classification results (accuracy) obtained by CNN-2.

**Figure A10.** Confusion matrix for the classification results (accuracy) obtained by CNN-3.



**Figure A11.** Confusion matrix for the classification results (accuracy) obtained by CNN-4.

**Figure A12.** Confusion matrix for the classification results (accuracy) obtained by DNN-2.
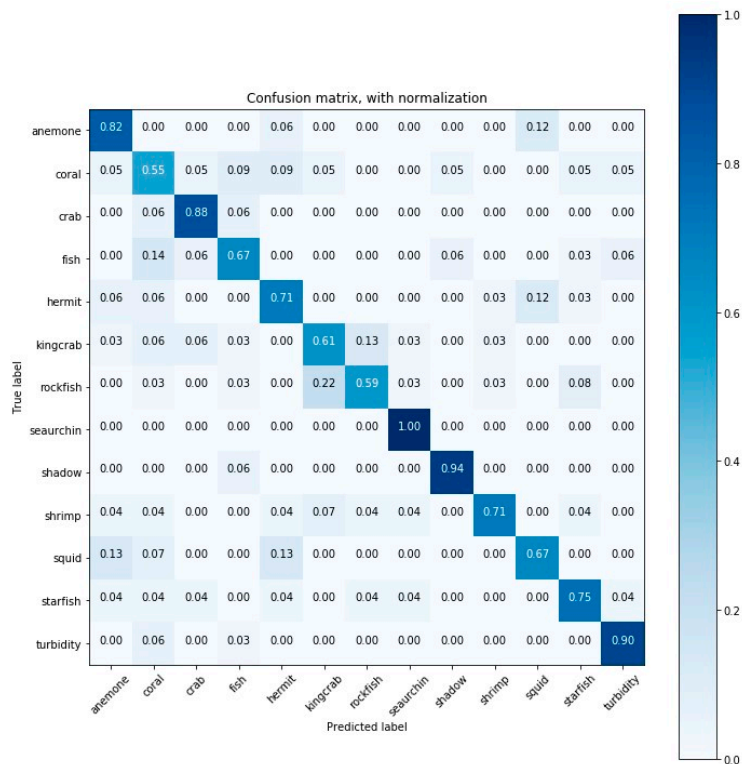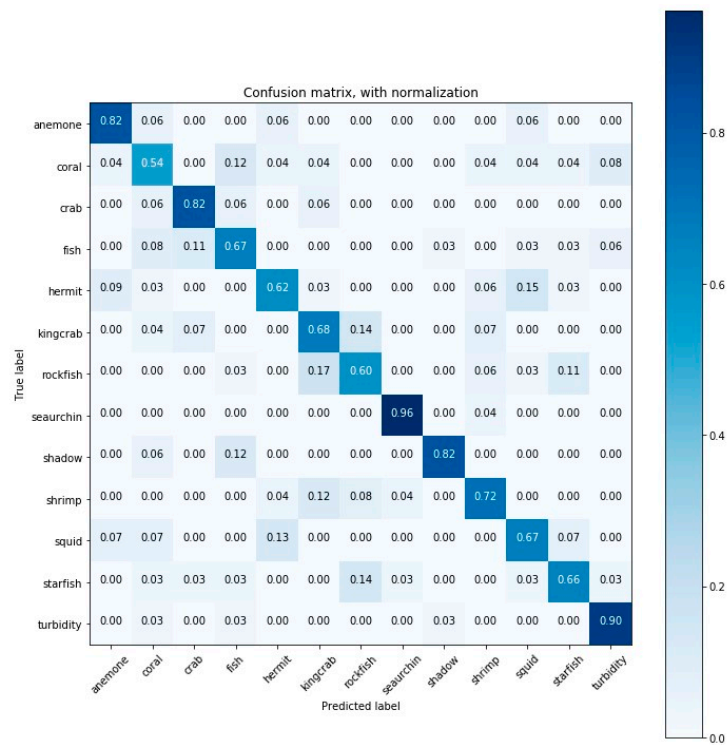


**Figure A13.** Confusion matrix for the classification results (accuracy) obtained by DNN-3.
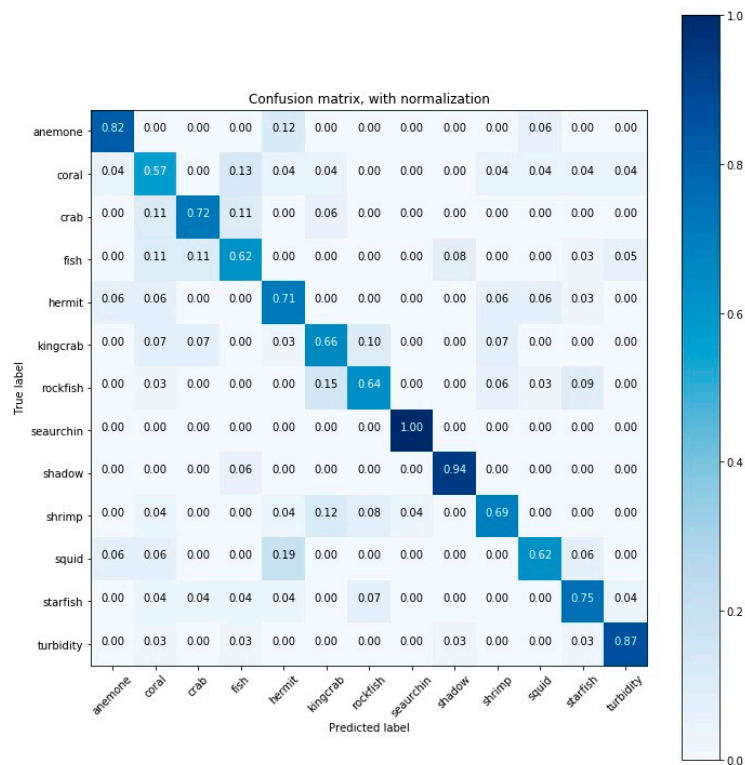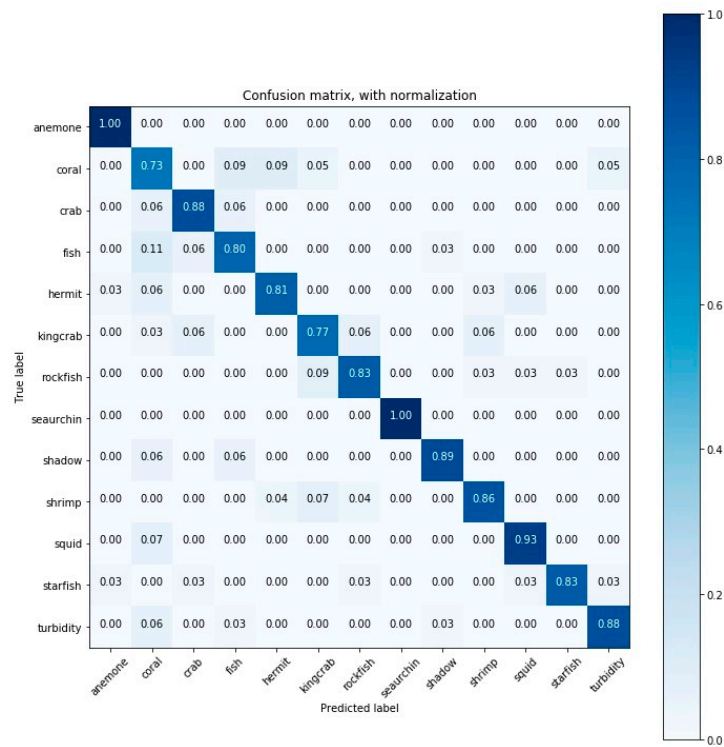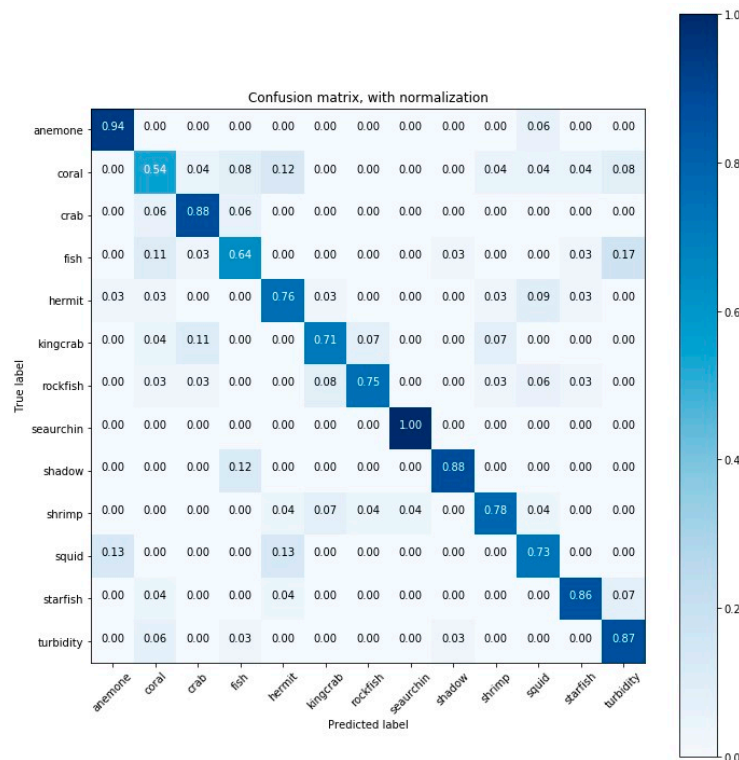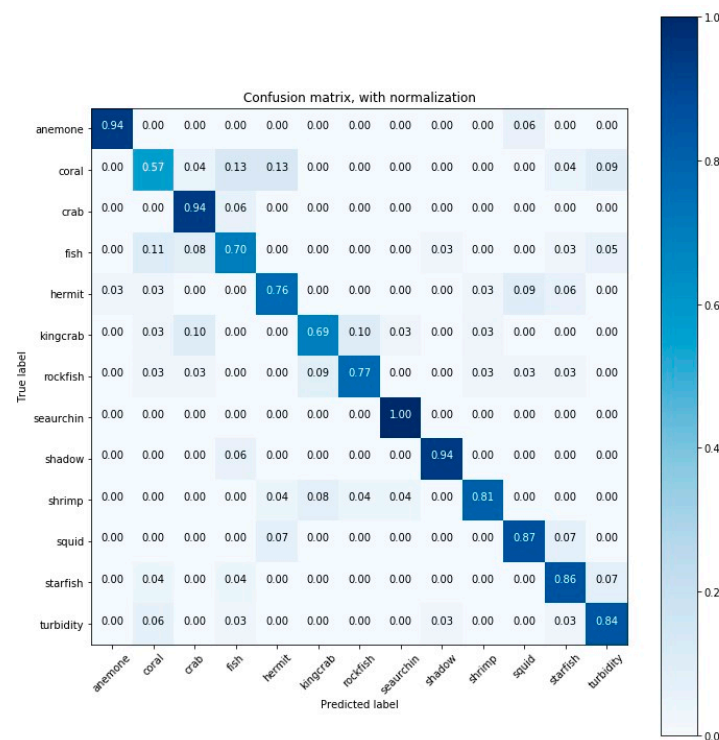
**Figure A14.** Confusion matrix for the classification results (accuracy) obtained by DNN-4.

## References

1. Bicknell, A.W.; Godley, B.J.; Sheehan, E.V.; Votier, S.C.; Witt, M.J. Camera technology for monitoring marine biodiversity and human impact. *Front. Ecol. Environ.* **2016**, *14*, 424–432. [CrossRef]

2. Danovaro, R.; Aguzzi, J.; Fanelli, E.; Billett, D.; Gjerde, K.; Jamieson, A.; Ramirez-Llodra, E.; Smith, C.; Snelgrove, P.; Thomsen, L.; et al. An ecosystem-based deep-ocean strategy. *Science* **2017**, *355*, 452–454. [CrossRef]

3. Aguzzi, J.; Chatzievangelou, D.; Marini, S.; Fanelli, E.; Danovaro, R.; Flögel, S.; Lebris, N.; Juanes, F.; Leo, F.C.D.; Rio, J.D.; et al. New High-Tech Flexible Networks for the Monitoring of Deep-Sea Ecosystems. *Environ. Sci. Tech.* **2019**, *53*, 6616–6631. [CrossRef] [PubMed]

4. Favali, P.; Beranzoli, L.; De Santis, A. *SEAFLOOR OBSERVATORIES: A New Vision of the Earth from the Abyss*; Springer Science & Business Media: Heidelberg, Germany, 2015.

5. Schoening, T.; Bergmann, M.; Ontrup, J.; Taylor, J.; Dannheim, J.; Gutt, J.; Purser, A.; Nattkemper, T. Semi-Automated Image Analysis for the Assessment of Megafaunal Densities at the Arctic Deep-Sea Observatory HAUSGARTEN. *PLoS ONE* **2012**, *7*, e38179. [CrossRef] [PubMed]

6. Aguzzi, J.; Doya, C.; Tecchio, S.; Leo, F.D.; Azzurro, E.; Costa, C.; Sbragaglia, V.; Rio, J.; Navarro, J.; Ruhl, H.; et al. Coastal observatories for monitoring of fish behaviour and their responses to environmental changes. *Rev. Fish Biol. Fisher.* **2015**, *25*, 463–483. [CrossRef]

7. Widder, E.; Robison, B.H.; Reisenbichler, K.; Haddock, S. Using red light for in situ observations of deep-sea fishes. *Deep-Sea Res PT I* **2005**, *52*, 2077–2085. [CrossRef]

8. Chauvet, P.; Metaxas, A.; Hay, A.E.; Matabos, M. Annual and seasonal dynamics of deep-sea megafaunal epibenthic communities in Barkley Canyon (British Columbia, Canada): a response to climatology, surface productivity and benthic boundary layer variation. *Prog. Oceanogr.* **2018**, *169*, 89–105. [CrossRef]

9. Leo, F.D.; Ogata, B.; Sastri, A.R.; Heesemann, M.; Mihály, S.; Galbraith, M.; Morley, M. High-frequency observations from a deep-sea cabled observatory reveal seasonal overwintering of Neocalanus spp. in Barkley Canyon, NE Pacific: Insights into particulate organic carbon flux. *Prog. Oceanogr.* **2018**, *169*, 120–137. [CrossRef]

10. Juniper, S.K.; Matabos, M.; Mihaly, S.F.; Ajayamohan, R.S.; Gervais, F.; Bui, A.O.V. A year in Barkley Canyon: A time-series observatory study of mid-slope benthos and habitat dynamics using the NEPTUNE Canada network. *Deep-Sea Res PT II* **2013**, *92*, 114–123. [CrossRef]

11. Doya, C.; Aguzzi, J.; Chatzievangelou, D.; Costa, C.; Company, J.B.; Tunnicliffe, V. The seasonal use of small-scale space by benthic species in a transiently hypoxic area. *J. Marine Syst.* **2015**, *154*, 280–290. [CrossRef]

12. Cuvelier, D.; Legendre, P.; Laes, A.; Sarradin, P.-M.; Sarrazin, J. Rhythms and Community Dynamics of a Hydrothermal Tubeworm Assemblage at Main Endeavour Field—A Multidisciplinary Deep-Sea Observatory Approach. *PLoS ONE* **2014**, *9*, e96924. [CrossRef]

13. Matabos, M.; Bui, A.O.V.; Mihály, S.; Aguzzi, J.; Juniper, S.; Ajayamohan, R. High-frequency study of epibenthic megafaunal community dynamics in Barkley Canyon: A multi-disciplinary approach using the NEPTUNE Canada network. *J. Marine Syst.* **2013**. [CrossRef]

14. Aguzzi, J.; Fanelli, E.; Ciuffardi, T.; Schirone, A.; Leo, F.C.D.; Doya, C.; Kawato, M.; Miyazaki, M.; Furushima, Y.; Costa, C.; et al. Faunal activity rhythms influencing early community succession of an implanted whale carcass offshore Sagami Bay, Japan. *Sci. Rep.* **2018**, *8*, 11163.

15. Mallet, D.; Pelletier, D. Underwater video techniques for observing coastal marine biodiversity: a review of sixty years of publications (1952–2012). *Fish. Res.* **2014**, *154*, 44–62. [CrossRef]

16. Chuang, M.-C.; Hwang, J.-N.; Williams, K. A feature learning and object recognition framework for underwater fish images. *IEEE Trans. Image Proc.* **2016**, *25*, 1862–1872. [CrossRef] [PubMed]

17. 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. [CrossRef]

18. Siddiqui, S.A.; Salman, A.; Malik, M.I.; Shafait, F.; Mian, A.; Shortis, M.R.; Harvey, E.S.; Browman, H. editor: H. Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data. *ICES J. Marine Sci.* **2017**, *75*, 374–389. [CrossRef]

19. Marini, S.; Fanelli, E.; Sbragaglia, V.; Azzurro, E.; Fernandez, J.D.R.; Aguzzi, J. Tracking Fish Abundance by Underwater Image Recognition. *Sci. Rep.* **2018**, *8*, 13748. [CrossRef]

20. Rountree, R.; Aguzzi, J.; Marini, S.; Fanelli, E.; De Leo, F.C.; Del Río, J.; Juanes, F. Towards an optimal design for ecosystem-level ocean observatories. *Front. Mar. Sci.* **2019**. [CrossRef]

21. Nguyen, H.; Maclagan, S.; Nguyen, T.; Nguyen, T.; Flemons, P.; Andrews, K.; Ritchie, E.; Phung, D. Animal Recognition and Identification with Deep Convolutional Neural Networks for Automated Wildlife Monitoring. In Proceedings of the IEEE International Conference on Data Science and Advanced Analytics (DSAA), Tokyo, Japan, 19–21 October 2017; pp. 40–49. [CrossRef]

22. Roberts, J.; Wheeler, A.; Freiwald, A. Reefs of the Deep: The Biology and Geology of Cold-Water Coral Ecosystems. *Sci. (New York, N.Y.)* **2006**, *312*, 543–547. [CrossRef]

23. Godø, O.; Tenningen, E.; Ostrowski, M.; Kubilius, R.; Kutti, T.; Korneliussen, R.; Fosså, J.H. The Hermes lander project - the technology, the data, and an evaluation of concept and results. *Fisken Havet.* **2012**, *3*.

24. Rune, G.O.; Johnsen, S.; Torkelsen, T. The love ocean observatory is in operation. *Mar. Tech. Soc. J.* **2014**, *48*, 24–30.

25. Hovland, M. *Deep-water Coral Reefs: Unique Biodiversity Hot-Spots*; Springer Science & Business Media: Heidelberg, Germany, 2008.

26. Sundby, S.; Fossum, P.A.S.; Vikebø, F.B.; Aglen, A.; Buhl-Mortensen, L.; Folkvord, A.; Bakkeplass, K.; Buhl-Mortensen, P.; Johannessen, M.; Jørgensen, M.S.; et al. *KunnskapsInnhenting Barentshavet–Lofoten–Vesterålen (KILO), Fisken og Havet 3, 1–186. Institute of Marine Research (in Norwegian)*; Fiskeri- og kystdepartementet: Bergen, Norway, 2013.

27. Bøe, R.; Bellec, V.; Dolan, M.; Buhl-Mortensen, P.; Buhl-Mortensen, L.; Slagstad, D.; Rise, L. Giant sandwaves in the Hola glacial trough off Vesterålen, North Norway. *Marine Geology* **2009**, *267*, 36–54. [CrossRef]

28. Engeland, T.V.; Godø, O.R.; Johnsen, E.; Duineveld, G.C.A.; Oevelen, D. Cabled ocean observatory data reveal food supply mechanisms to a cold-water coral reef. *Prog. Oceanogr.* **2019**, *172*, 51–64. [CrossRef]

29. Fosså, J.H.; Buhl-Mortensen, P.; Furevik, D.M. Lophelia-korallrev langs norskekysten forekomst og tilstand. *Fisken og Havet* **2000**, *2*, 1–94.

30. Ekman, S. *Zoogeography of the Sea*; Sidgwood and Jackson: London, UK, 1953; Volume 417.

31. O'Riordan, C.E. Marine Fauna Notes from the National Museum of Ireland–10. *INJ* **1986**, *22*, 34–37.

32. Hureau, J.C.; Litvinenko, N.I. Scorpaenidae. In *Fishes of the North-eastern Atlantic and the Mediterranean (FNAM)*; P.J.P., W., Ed.; UNESCO: Paris, France, 1986; pp. 1211–1229.

33. Marini, S.; Corgnati, L.; Mantovani, C.; Bastianini, M.; Ottaviani, E.; Fanelli, E.; Aguzzi, J.; Griffa, A.; Poulain, P.-M. Automated estimate of fish abundance through the autonomous imaging device GUARD1. *Measurement* **2018**, *126*, 72–75. [CrossRef]

34. Reza, A.M. Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. *J. VLSI Sig. Proc. Syst. Sig. Image Video Tech.* **2004**, *38*, 35–44. [CrossRef]

35. Pizer, S.M.; Amburn, E.P.; Austin, J.D.; Cromartie, R.; Geselowitz, A.; Greer, T.; Romeny, B.H.; Zimmerman, J.B.; Zuiderveld, K. Adaptive histogram equalization and its variations. *Image Vis. Comput.* **1987**, *39*, 355–368. [CrossRef]

36. Ouyang, B.; Dalgleish, F.R.; Caimi, F.M.; Vuorenkoski, A.K.; Giddings, T.E.; Shirron, J.J. Image enhancement for underwater pulsed laser line scan imaging system. In Proceedings of the Ocean Sensing and Monitoring IV; International Society for Optics and Photonics, Baltimore, MD, USA, 24–26 April 2012; Volume 8372.

37. Lu, H.; Li, Y.; Zhang, L.; Yamawaki, A.; Yang, S.; Serikawa, S. Underwater optical image dehazing using guided trigonometric bilateral filtering. In Proceedings of the 2013 IEEE International Symposium on Circuits and Systems (ISCAS2013), Beijing, China, 19–23 May 2013; pp. 2147–2150.

38. Serikawa, S.; Lu, H. Underwater image dehazing using joint trilateral filter. *Comput. Electr. Eng.* **2014**, *40*, 41–50. [CrossRef]

39. Tomasi, C.; Manduchi, R. Bilateral filtering for gray and color images. In Proceedings of the IEEE Sixth International Conference on Computer Vision, Bombay, India, 4–7 January 1998; pp. 839–846. [CrossRef]

40. Aguzzi, J.; Costa, C.; Fujiwara, Y.; Iwase, R.; Ramirez-Llorda, E.; Menesatti, P. A novel morphometry-based protocol of automated video-image analysis for species recognition and activity rhythms monitoring in deep-sea fauna. *Sensors* **2009**, *9*, 8438–8455. [CrossRef] [PubMed]

41. Peters, J. *Foundations of Computer Vision: computational geometry, visual image structures and object shape detection*; Springer International Publishing: Cham, Switzerland, 2017.

42. Aguzzi, J.; Lázaro, A.; Condal, F.; Guillen, J.; Nogueras, M.; Rio, J.; Costa, C.; Menesatti, P.; Puig, P.; Sardà, F.; et al. The New Seafloor Observatory (OBSEA) for Remote and Long-Term Coastal Ecosystem Monitoring. *Sensors* **2011**, *11*, 5850–5872. [CrossRef] [PubMed]

43. Albarakati, H.; Ammar, R.; Alharbi, A.; Alhumyani, H. An application of using embedded underwater computing architectures. In Proceedings of the IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Limassol, Cyprus, 12–14 December 2016; pp. 34–39.

44. OpenCV (Open source computer vision). Available online: https://opencv.org/ (accessed on 25 November 2019).

45. Coelho, L.P. Mahotas: Open source software for scriptable computer vision. *arXiv* **2012**, arXiv:1211.4907.

46. Spampinato, C.; Giordano, D.; Salvo, R.D.; Chen-Burger, Y.-H.J.; Fisher, R.B.; Nadarajan, G. Automatic fish classification for underwater species behavior understanding. In Proceedings of the MM '10: ACM Multimedia Conference, Firenze, Italy, 25–29 October 2010; pp. 45–50.

47. Tharwat, A.; Hemedan, A.A.; Hassanien, A.E.; Gabel, T. A biometric-based model for fish species classification. *Fish. Res.* **2018**, *204*, 324–336. [CrossRef]

48. Kitasato, A.; Miyazaki, T.; Sugaya, Y.; Omachi, S. Automatic Discrimination between Scomber japonicus and Scomber australasicus by Geometric and Texture Features. *Fishes* **2018**, *3*, 26. [CrossRef]

49. Wong, R.Y.; Hall, E.L. Scene matching with invariant moments. *Comput. Grap. Image Proc.* **1978**, *8*, 16–24. [CrossRef]

50. Haralick, R.M.; Shanmugam, K. others Textural features for image classification. *IEEE Trans. Syst. Man Cybern.* **1973**, *SMC-3*, 610–621. [CrossRef]

51. Zuiderveld, K. Contrast limited adaptive histogram equalization. In *Graphics gems IV*; Academic Press Professional, Inc: San Diego, CA, USA, 1994; pp. 474–485.

52. Tusa, E.; Reynolds, A.; Lane, D.M.; Robertson, N.M.; Villegas, H.; Bosnjak, A. Implementation of a fast coral detector using a supervised machine learning and gabor wavelet feature descriptors. In Proceedings of the 2014 IEEE Sensor Systems for a Changing Ocean (SSCO), Brest, France, 13–14 October 2014; pp. 1–6.

53. Saberioon, M.; Císař, P.; Labbé, L.; Souček, P.; Pelissier, P.; Kerneis, T. Comparative Performance Analysis of Support Vector Machine, Random Forest, Logistic Regression and k-Nearest Neighbours in Rainbow Trout (Oncorhynchus Mykiss) Classification Using Image-Based Features. *Sensors* **2018**, *18*, 1027. [CrossRef]

54. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]

55. Vapnik, V.N. *The Nature of Statistical Learning Theory*; Springer-Verlag: Berlin/Heidelberg, Germany, 1995.

56. Vapnik, V.N. *Statistical learning theory*; John Wiley: New York, NY, USA, 1998.

57. Scholkopf, B.; Sung, K.-K.; Burges, C.J.C.; Girosi, F.; Niyogi, P.; Poggio, T.; Vapnik, V. Comparing support vector machines with Gaussian kernels to radial basis function classifiers. *IEEE Trans. Signal Proc.* **1997**, *45*, 2758–2765. [CrossRef]

58. Spampinato, C.; Palazzo, S.; Joalland, P.-H.; Paris, S.; Glotin, H.; Blanc, K.; Lingrand, D.; Precioso, F. Fine-grained object recognition in underwater visual data. *Multimed. Tools and Appl.* **2016**, *75*, 1701–1720. [CrossRef]

59. Rova, A.; Mori, G.; Dill, L.M. One fish, two fish, butterfish, trumpeter: Recognizing fish in underwater video. In Proceedings of the MVA, Tokyo, Japan, 16–18 May 2007; pp. 404–407.

60. Fix, E.; Hodges, J.L. *Discriminatory analysis-nonparametric discrimination: consistency properties*; California Univ Berkeley: Berkeley, CA, USA, 1951.

61. Magee, J.F. *Decision Trees for Decision Making*; Harvard Business Review, Harvard Business Publishing: Brighton, MA, USA, 1964.

62. Argentiero, P.; Chin, R.; Beaudet, P. An automated approach to the design of decision tree classifiers. *IEEE T Pattern Anal.* **1982**, *PAMI-4*, 51–57. [CrossRef]

63. Kalochristianakis, M.; Malamos, A.; Vassilakis, K. Color based subject identification for virtual museums, the case of fish. In Proceedings of the 2016 International Conference on Telecommunications and Multimedia (TEMU), Heraklion, Greece, 25–27 July 2016; pp. 1–5.

64. Freitas, U.; Gonçalves, W.N.; Matsubara, E.T.; Sabino, J.; Borth, M.R.; Pistori, H. Using Color for Fish Species Classification. Available online: gibis.unifesp.br/sibgrapi16/eproceedings/wia/1.pdf (accessed on 28 January 2020).

65. Breiman, L. Random forests. *Machine Learning* **2001**, *45*, 5–32. [CrossRef]

66. Ho, T.K. Random decision forests. In Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, Canada, 14–16 August 1995; pp. 278–282.

67. Fang, Z.; Fan, J.; Chen, X.; Chen, Y. Beak identification of four dominant octopus species in the East China Sea based on traditional measurements and geometric morphometrics. *Fish. Sci.* **2018**, *84*, 975–985. [CrossRef]

68. Ali-Gombe, A.; Elyan, E.; Jayne, C. Fish classification in context of noisy images. In *Engineering Applications of Neural Networks, Proceedings of the 18th International Conference on Engineering Applications of Neural Networks, Athens, Greece, August 25–27, 2017*; Boracchi, G., Iliadis, L., Jayne, C., Likas, A., Eds.; Springer: Cham, Switzerland, 2017; pp. 216–226.

69. Rachmatullah, M.N.; Supriana, I. Low Resolution Image Fish Classification Using Convolutional Neural Network. In Proceedings of the 2018 5th International Conference on Advanced Informatics: Concept Theory and Applications (ICAICTA), Krabi, Thailand, 14–17 August 2018; pp. 78–83.

70. Rathi, D.; Jain, S.; Indu, D.S. Underwater Fish Species Classification using Convolutional Neural Network and Deep Learning. *arXiv* **2018**, arXiv:1805.10106.

71. Rimavicius, T.; Gelzinis, A. A Comparison of the Deep Learning Methods for Solving Seafloor Image Classification Task. In *Information and Software Technologies, Proceedings of the 23rd International Conference on Information and Software Technologies, Druskininkai, Lithuania, October 12–14*; Damaševičius, R., Mikašytė, V., Eds.; Springer: Cham, Switzerland, 2017; pp. 442–453.

72. Gardner, W.A. Learning characteristics of stochastic-gradient-descent algorithms: A general study, analysis, and critique. *Sig. Proc.* **1984**, *6*, 113–133. [CrossRef]

73. Zou, H.; Hastie, T. Regularization and variable selection via the elastic net. *J. R Stat. Soc: B* **2005**, *67*, 301–320. [CrossRef]

74. Duda, R.O.; Hart, P.E. *Pattern Classification and Scene Analysis*; A Wiley-Interscience Publication: New York, NY, USA, 1973.

75. Jonsson, P.; Wohlin, C. An evaluation of k-nearest neighbour imputation using likert data. In Proceedings of the 10th International Symposium on Software Metrics, Chicago, IL, USA, 11–17 September 2004; pp. 108–118.

76. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

77. Zeiler, M.D. ADADELTA: an adaptive learning rate method. *arXiv preprint* **2012**, arXiv:1212.5701.

78. Tieleman, T.; Hinton, G. Divide the gradient by a running average of its recent magnitude. COURSERA Neural Netw. *Mach. Learn* **2012**, *6*, 26–31.

79. Gulcehre, C.; Moczulski, M.; Denil, M.; Bengio, Y. Noisy activation functions. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 3059–3068.

80. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.

81. Fawcett, T. An introduction to ROC analysis. *Pattern Recogn. Lett.* **2006**, *27*, 861–874. [CrossRef]

82. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 60. [CrossRef]

83. Pan, Z.; Yu, W.; Yi, X.; Khan, A.; Yuan, F.; Zheng, Y. Recent progress on generative adversarial networks (GANs): A survey. *IEEE Access* **2019**, *7*, 36322–36333. [CrossRef]

84. Cao, Y.-J.; Jia, L.-L.; Chen, Y.-X.; Lin, N.; Yang, C.; Zhang, B.; Liu, Z.; Li, X.-X.; Dai, H.-H. Recent Advances of Generative Adversarial Networks in Computer Vision. *IEEE Access* **2018**, *7*, 14985–15006. [CrossRef]

85. Shao, L.; Zhu, F.; Li, X. Transfer learning for visual categorization: A survey. *IEEE T Neural Networ. Learn. Syst.* **2014**, *26*, 1019–1034. [CrossRef]

86. Konovalov, D.A.; Saleh, A.; Bradley, M.; Sankupellay, M.; Marini, S.; Sheaves, M. Underwater Fish Detection with Weak Multi-Domain Supervision. *arXiv preprint* **2019**, arXiv:1905.10708.

87. Villon, S.; Chaumont, M.; Subsol, G.; Villéger, S.; Claverie, T.; Mouillot, D. Coral reef fish detection and recognition in underwater videos by supervised machine learning: Comparison between Deep Learning and HOG+ SVM methods. In Proceedings of the International Conference on Advanced Concepts for Intelligent Vision Systems, Lecce, Italy, 24–27 October 2016; pp. 160–171.

88. Hu, G.; Wang, K.; Peng, Y.; Qiu, M.; Shi, J.; Liu, L. Deep learning methods for underwater target feature extraction and recognition. *Computational Intell. Neurosci.* **2018**. [CrossRef]

89. Salman, A.; Jalal, A.; Shafait, F.; Mian, A.; Shortis, M.; Seager, J.; Harvey, E. Fish species classification in unconstrained underwater environments based on deep learning. *Limnol. Oceanogr: Meth.* **2016**, *14*, 570–585. [CrossRef]

90. Cao, X.; Zhang, X.; Yu, Y.; Niu, L. Deep learning-based recognition of underwater target. In Proceedings of the 2016 IEEE International Conference on Digital Signal Processing (DSP), Shanghai, China, 19–21 November 2016; pp. 89–93.

91. Pelletier, S.; Montacir, A.; Zakari, H.; Akhloufi, M. Deep Learning for Marine Resources Classification in Non-Structured Scenarios: Training vs. Transfer Learning. In Proceedings of the 2018 31st IEEE Canadian Conference on Electrical & Computer Engineering (CCECE), Quebec City, QC, Canada, 13–16 May 2018; pp. 1–4.

92. Sun, X.; Shi, J.; Liu, L.; Dong, J.; Plant, C.; Wang, X.; Zhou, H. Transferring deep knowledge for object recognition in Low-quality underwater videos. *Neurocomputing* **2018**, *275*, 897–908. [CrossRef]

93. Xu, W.; Matzner, S. Underwater Fish Detection using Deep Learning for Water Power Applications. *arXiv preprint* **2018**, arXiv:1811.01494.

94. Wang, X.; Ouyang, J.; Li, D.; Zhang, G. Underwater Object Recognition Based on Deep Encoding-Decoding Network. *J. Ocean Univ. Chin.* **2018**, 1–7. [CrossRef]

95. Naddaf-Sh, M.; Myler, H.; Zargarzadeh, H. Design and Implementation of an Assistive Real-Time Red Lionfish Detection System for AUV/ROVs. *Complexity* **2018**, *2018*. [CrossRef]

96. Villon, S.; Mouillot, D.; Chaumont, M.; Darling, E.S.; Subsol, G.; Claverie, T.; Villéger, S. A Deep learning method for accurate and fast identification of coral reef fishes in underwater images. *Ecolog. Infor.* **2018**, 238–244. [CrossRef]