Underwater Vision-Based Gesture Recognition

By Arturo Gomez Chavez, Andrea Ranieri, Davide Chiarella, and Andreas Birk

Underwater robotics requires very reliable and safe operations. This holds especially true for missions in cooperation with divers who are—despite the significant advancements of marine robotics in recent years—still essential for many underwater operations. Possible application cases of underwater human–robot collaboration include marine science, archeology, oil and gas production, handling of unexploded ordnance (e.g., from World War II ammunition dumped in the seas), or the inspection and maintenance of marine infrastructure like pipelines, harbors, or renewable energy installations, to name just a few examples.

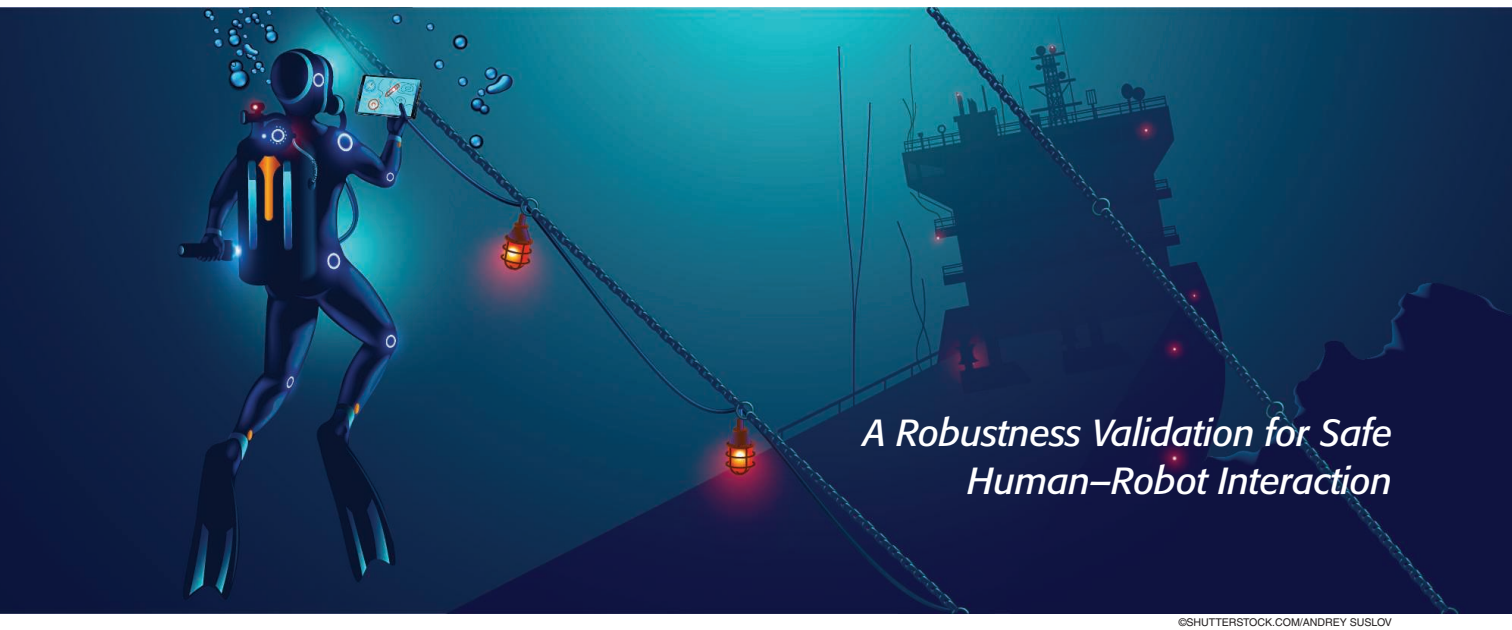
We present a fully integrated approach to underwater human–robot interaction (U-HRI) in the form of a front end for gesture recognition combined with a back end with a full language interpreter. The gesture-based language is derived from the existing standard gestures for communication between human divers. It enables a diver to issue single commands as well as complex mission specifications to an autonomous underwater vehicle (AUV) as demonstrated in several field trials.

Gesture recognition is an essential component of the overall approach. It requires high reliability under the challenging conditions of the underwater domain. There is an especially high amount of variation in the visual data due to various effects in underwater image formation. Hence, in this article we investigate different machine learning (ML) methods for robust diver gesture recognition. This includes a classical ML approach and four state-of-the-art deep learning (DL) methods. Furthermore, we introduce a physically realistic way to use range information for adding underwater haze to produce meaningful additional data from existing real-world data. This can be of interest for creating evaluation data for underwater perception in general or for producing additional training data for ML-based approaches.

Related Work

Given the importance of cameras for underwater systems, especially for near-field perception, computer vision is predominantly used for U-HRIs. Alternatives are acoustic approaches with pingers or sonars as well as the use of dedicated devices like underwater tablets [1], [2]. The first step

Digital Object Identifier 10.1109/MRA.2021.3075560
Date of current version: 20 May 2021



©SHUTTERSTOCK.COM/ANDREY SUSLOV

1070-9932/21©2021IEEE

toward U-HRI is the detection and tracking of one or multiple divers [3]–[13]. Given the relative localization, different protocols for interaction can be studied and trained in a computer simulation [14].

Relative motions between divers and robots can already be used for a basic nonverbal form of communication [15], but more capable forms of communication—in terms of expressiveness and reliability—are needed to enable real U-HRI for collaborative missions. Work in that direction is described in [16], where artificial fiducial markers are used that are then interpreted by the robot using grammatical rules. While cards with artificial markers ease the challenges of underwater vision, there are disadvantages like the number of cards that the diver must carry and the effort required to handle them.

Gestures are a more natural basis for underwater communication because 1) they are already extensively used by divers and 2) they operate despite the limitations of water as a medium, e.g., it is impossible to use voice recognition. Early research on the use of gestures for U-HRI is described in [17], where waving gestures are recognized by differential imaging with a spectral registration method in the form of the improved Fourier Mellin invariant. Based on that, trajectories of hand motions are recognized with a finite state machine (FSM). The experiments in [17] are done in a pool.

An imaging sonar, also known as an *acoustic camera*, is used in [18] for gesture recognition. Preprocessing stages with cascade classifiers and shape processing are combined with three different classification approaches, namely, a convex hull method, support vector machines, and the fusion of both. Experiments are conducted in a pool as well as during field trials with divers in the context of the EU project Cognitive Autonomous Diving Buddy (CADDY) (Figure 1). The selection of device parameters within a mission is a known challenge for this type of sensor, which is also reported in [18].

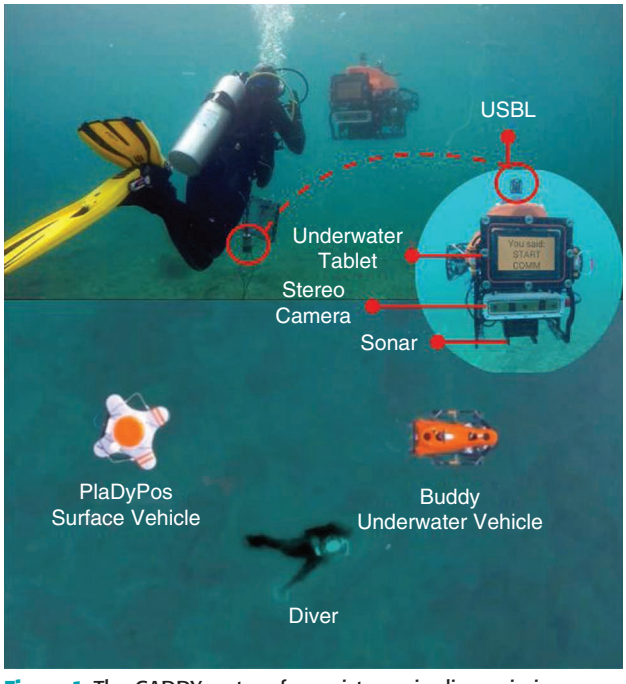


Figure 1. The CADDY system for assistance in diver missions. (Inset) The Buddy AUV is equipped with a Blueprint Subsea X150 ultra-short baseline (USBL), an underwater tablet, a BumbleBeeXB3 stereo camera, and an ARIS 3000 imaging sonar for diver tracking, monitoring, and communication. In the top portion of the image, a diver gestures a command, and the lower portion of the image shows an aerial view of the system with a PlaDyPos surface vehicle for global positioning.

The main type of sensor for U-HRI in CADDY is, therefore, a (stereo)camera. To process the visual data, a modification of nearest class mean forests (NCMF) in the form of a multidescriptor extension (MD-NCMF) is introduced. MD-NCMF is used for both diver detection and tracking [8] and the classification of diver gestures [19]. As the name suggests, MD-NCMF is designed to exploit different types of descriptors to achieve high robustness under the challenging conditions of underwater visibility. To this end, MD-NCMF builds on NCMF, which partitions the sample space by comparing the distances between class means instead of comparing values at each feature dimension, as in more traditional random forests approaches. Therefore, MD-NCMF can treat each feature–object pair as a new class, e.g., speeded up robust features (SURF) object1, scale-invariant feature transform (SIFT) object2, SURF object2, SIFT background, and so on, and MD-NCMF can examine which one provides the best partition of the sample set.

Based on MD-NCMF gesture recognition [19], a machine interpreter [20] with a phrase parser, syntax checker, and command dispatcher linked to the mission control allows the use of a very expressive language for U-HRI [21]. This Caddian language is based on a context-free grammar that allows the diver to specify missions with a sequence of tasks. The syntax checker is implemented as an FSM that gives constant feedback to the diver and allows for in situ corrections. The gesture recognition front end and the machine interpreter back end are reported in field tests to be not only robust but also useful in complex missions with professional divers [22], [23].

A full language for U-HRI is also presented in [24]. It is syntactically a bit simpler than Caddian, as the FSM in its interpreter is restricted to only one possible transition from state to state, i.e., gesture to gesture, to avoid ambiguities. The gesture recognition front end in [24] is based on DL models. More precisely, the single shot detector (SSD) [25] and faster region-based convolutional neural networks (faster R-CNNs) [26] are investigated, which achieve more than 90% accuracy when being trained with a data set of 50,000 points.

It is assumed in [24] that the diver wears no gloves; this enables the use of skin detection and image contour estimation. In practice, professional divers tend to always wear gloves, both for protection and to avoid heat loss. For the MD-NCMF gesture recognition [19] mentioned previously, regular diving gloves are augmented with colored stripes to provide some detectable contrast. The first results toward a classification under a wide range of conditions, including divers with and without gloves, are presented in [27]. Building upon a DL-based approach dubbed SCUBANet to recognize diver body parts [28], MobileNetV2 [29] is trained to

recognize 25 image classes using finger count and palm direction, although the authors also state that a significant portion of these classes is unused in most gestures [27].

U-HRI With Gesture-Based Communication

Our gesture-based communication for U-HRI consists of a gesture recognition front end and an interpreter back end. In this article, different options for the front end are investigated, as described in the "Gesture Detection and Classification" section. A short overview of the actual language and the interpreter back end is then given in the "The Caddian Language and Its Interpretation" section. An example from a field trial in the "Example Use Case and Challenges" section illustrates the use of the complete system and the challenges that occur in practice and motivates the investigation of different DL methods.

ML in general and DL in particular typically require high amounts of data for training and evaluation. A physically realistic way to use the range information for adding underwater haze is hence introduced in the "Physically Realistic Underwater Image Degradation" section. This is used to add artificial degradations to existing real-world images from field trials to produce additional data, which are useful for covering the high amount of variability in the underwater domain without the need for many costly field campaigns.

Gesture Detection and Classification

MD-NCMF as a Classical ML Approach

MD-NCMF is a multidescriptor extension of NCMF that is used for both diver detection/tracking and the classification of diver gestures [8], [19] (Figure 2). This variant of random forests aggregates multiple descriptors (SIFT, SURF, ORB, HoG, and so on) that encode different representations of the objects of interest as we observed that each of these descriptors is robust to different types of underwater image degradations. MD-NCMF can be considered to be a classical ML approach, which forms a comparison basis for the different DL methods described in the next section.

For the first step of hand detection, both 2D monocular images and 2.5D stereo disparity are used. The 2.5D disparity maps are segmented based on distance and density. This provides reliable hand detection in many cases. However, it fails for texture-rich interferences close to the stereo camera, e.g., due to air bubbles. Therefore, 2D cascade classifiers are used in a second process running in parallel to filter out the false positive regions. The resulting region proposals, i.e., object candidates, serve as the input to the actual classifier. MD-NCMF then filters out further false positives that may still exist, and it maps the hand regions to the gestures of the Caddian language described in the following sections.

DL Approaches

State-of-the-art deep models for visual object detection and classification often follow three meta-architectures: SSD [25], faster R-CNN [26], and region-based fully convolutional neural network (R-FCN). SSD models offer fast computation speeds since they perform object detection and classification in one single pass of the network. These are, hence, often preferred for embedded systems. Faster R-CNN has two stages that are conceptually similar to the described classical ML approach (see the "MD-NCMF as a Classical ML Approach" section): a region proposal network generates candidates for object regions, and a classifier then verifies and refines the proposals. The R-FCN architecture is a mixture of the previous two meta-architectures. It shares features learned in the initial layers between the region proposal and the actual classifier network.

To this end, MD-NCMF builds on NCMF, which partitions the sample space by comparing the distances between class means instead of comparing values at each feature dimension, as in more traditional random forests approaches.

The DL models are used with pretrained feature extractors (Table 1). A fully connected network like ResNet [30] can be considered the most straightforward approach since it requires only one label per image and no region candidate, which ultimately satisfies our system's requirements. The SSD [25] and faster R-CNN [26] differ mostly in their architectures among the considered DL methods; the former is tailored toward fast computation when using the MobileNet feature extractor [31]. A deformable ConvNet [32] allows region proposals with nonuniform boundaries by using a flexible sampling grid on the image. Thus, it is no longer assumed that the object geometry is fixed, which can be beneficial for detecting 6-degrees of freedom hands of a free-floating diver.

The Caddian Language and Its Interpretation

The gestures form a language for U-HRI called Caddian [20], which is derived from the routine communication of divers. The Caddian syntax defines boundaries to understand complex commands, i.e., sequences of gestures, which can also be aggregated to form missions composed of several tasks. Two gestures to start a command and to end a communication, denoted as A and $\forall$, are used for this purpose. Commands are sequences of individual gestures delimited (A, A) that represent a single task. A practical example from field trials is the command "Take a photo at 3 meters altitude." Missions consist of aggregated commands that are delimited by $(A, \forall)$. An example for a mission used in practice is "Take a photo, go to the boat, and carry the equipment back."

To handle very frequent tasks or emergencies, there is a special slang group of gestures. They have higher priority and a simpler syntax. Examples include a gesture to instruct the AUV to take a photo at the current location, i.e., without

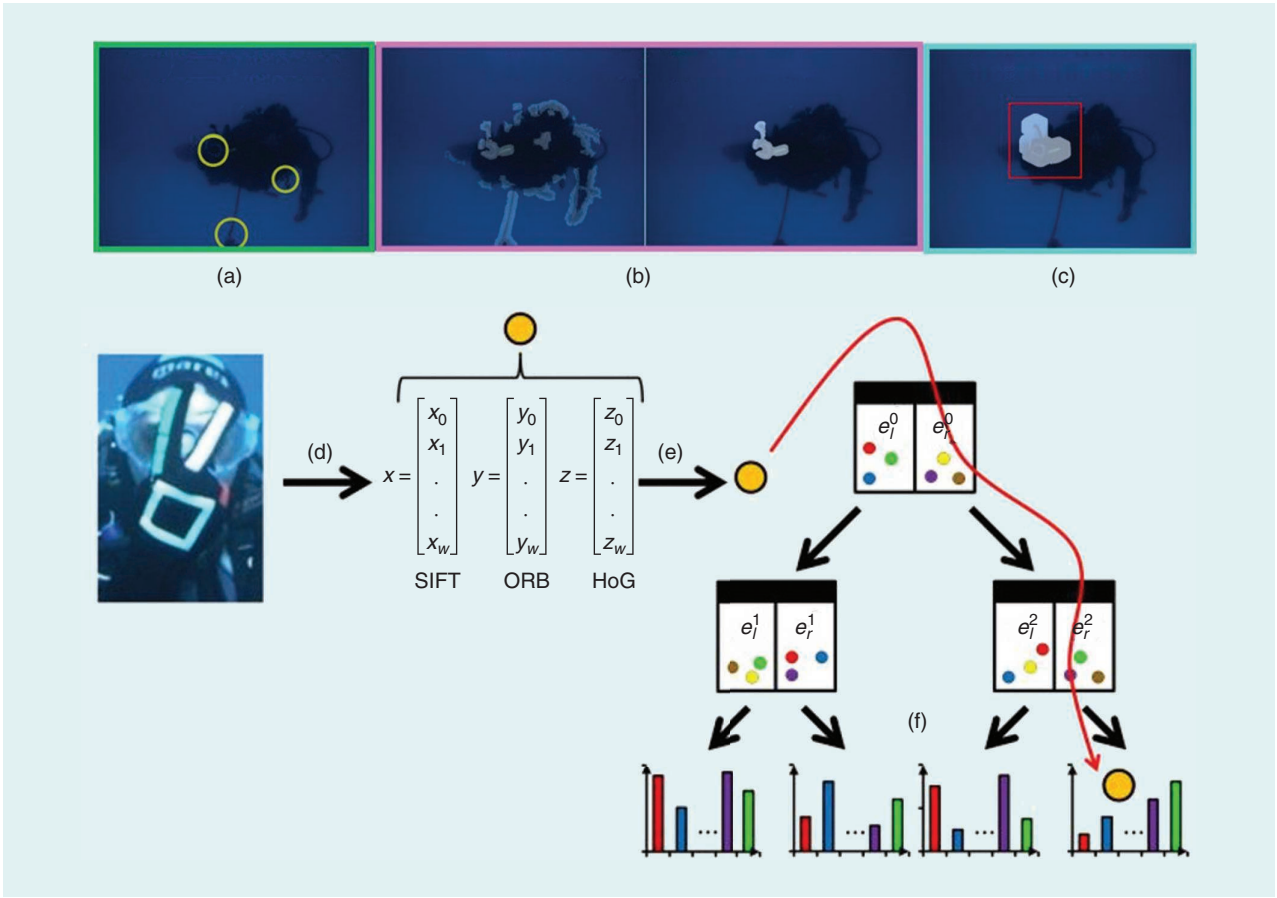


Figure 2. Hand detection, where the possible regions are detected by two processes running in parallel: (a) a Haar cascade model and (b) a disparity map that is thresholded by distance and morphologically transformed to reduce noise. (c) A cross check between the two methods generates the final hand image candidates. (d)–(f) Gesture classification using an MD-NCM tree. (d) Each class centroid (marked by a colored dot) traverses a path through the decision tree. (e) The image is encoded into different types of feature vectors $\vec{x}\,\vec{y}\,\vec{z}$. (f) The sample passes down the tree following the closest centroid as an aggregated similarity measure. ORB: oriented FAST and rotated BRIEF; HoG: histogram of oriented gradients.

specifying any parameters, or a gesture to signal that the diver is out of air, which triggers the emergency response protocols on the AUV and the surface vehicle to which it is connected.

As illustrated in Figure 3(a), the phrase parser constantly saves the recognized gestures until it detects one of the delimiter pairs (A, A), $(A, \forall)$. It then sends the gesture set to the syntax checker for validation. If the command is syntactically correct, it is passed to the command dispatcher, where it is saved until a complete mission is received. After the diver confirms, the commands are passed to the mission controller for execution.

Despite the syntax validation, gestures can be misclassified; i.e., a message can have the correct structure but represents an infeasible or undesired action. Therefore, the system integrates the diver in a human-in-the-loop approach to identify and correct possible errors as quickly as possible. Four types of feedback are provided to the diver at different times during the communication process through an underwater tablet on the AUV (see Figure 3).

1) *Single gesture*: Every time a gesture is recognized, the tablet displays the classification label given to that gesture.
2) *Syntax error*: Whenever a phrase/command is detected and analyzed by the syntax checker, an error is displayed if the Caddian grammatical rules are not followed. The received message is shown to the diver for his/her analysis, and the communication is reset.
3) *Mission confirmation request*: When the diver ends communication, the system displays the complete mission, and it waits for a confirmation gesture or a gesture to abort.

Table 1. An overview of the four DL models and the pretrained feature extractors.

Visual Model	Feature Extractor	Software Library	References
FC-CNN	ResNet-50	Fast.ai/Pytorch	[30]
SSD	MobileNets	Tensorflow	[25], [31]
Faster R-CNN	ResNet-101	Tensorflow	[26], [30]
Deformable faster R-CNN	[32]	MXNet	[26], [32]

4) *Mission status*: When the AUV is executing a mission, the current status of the mission is displayed, e.g., carrying equipment and mission completed.

Example Use Case and Challenges

The following example illustrates a typical use case and the challenges. The related field trial emulates an archaeological mission. It was designed in cooperation with the Diver Alert Network Europe to take safety and ergonomic aspects in the evaluation into account. The test was done in Biograd na Moru, Croatia, in 2016. Figure 4 depicts the overall mission. The field trial tested many system functionalities, not only gesture-based communication. Therefore, only the photo and boat commands happened to be used, along with the gestures for the start and the end of the communication. More commands and also full missions were tested in other trials in 2017 at the Brodarski Institute in Zagreb, Croatia (Figure 5). But the Biograd field trial nicely illustrates the challenges: there is, for example, reduced visibility compared to experiments in a pool, and the sea was rough on several days, causing motion blur as neither the divers nor the AUV could keep still on the spot.

Figure 4(c) shows an example where the photo command is incorrectly classified by the gesture recognition front end. Note that, due to the interpreter back end, the error was captured and remedied, indicating the importance of a complete HRI and collaboration framework to ensure correct operation and diver safety. Also, accuracy is boosted in the real system by fusing the classification results of multiple consecutive frames. Nonetheless, the higher the accuracy in the gesture recognition, the more convenient the system is to use. Even more importantly, it is of high interest to check that the recognition does not fail in previously untested environment conditions, i.e., that the accuracy does not suddenly drop to unexpectedly low rates. This evaluation across a wide range of environmental conditions is a nontrivial chore for underwater vision in general.

Physically Realistic Underwater Image Degradation

Underwater image formation is influenced by many factors [33]. In addition to the complexity of underwater field trials, it

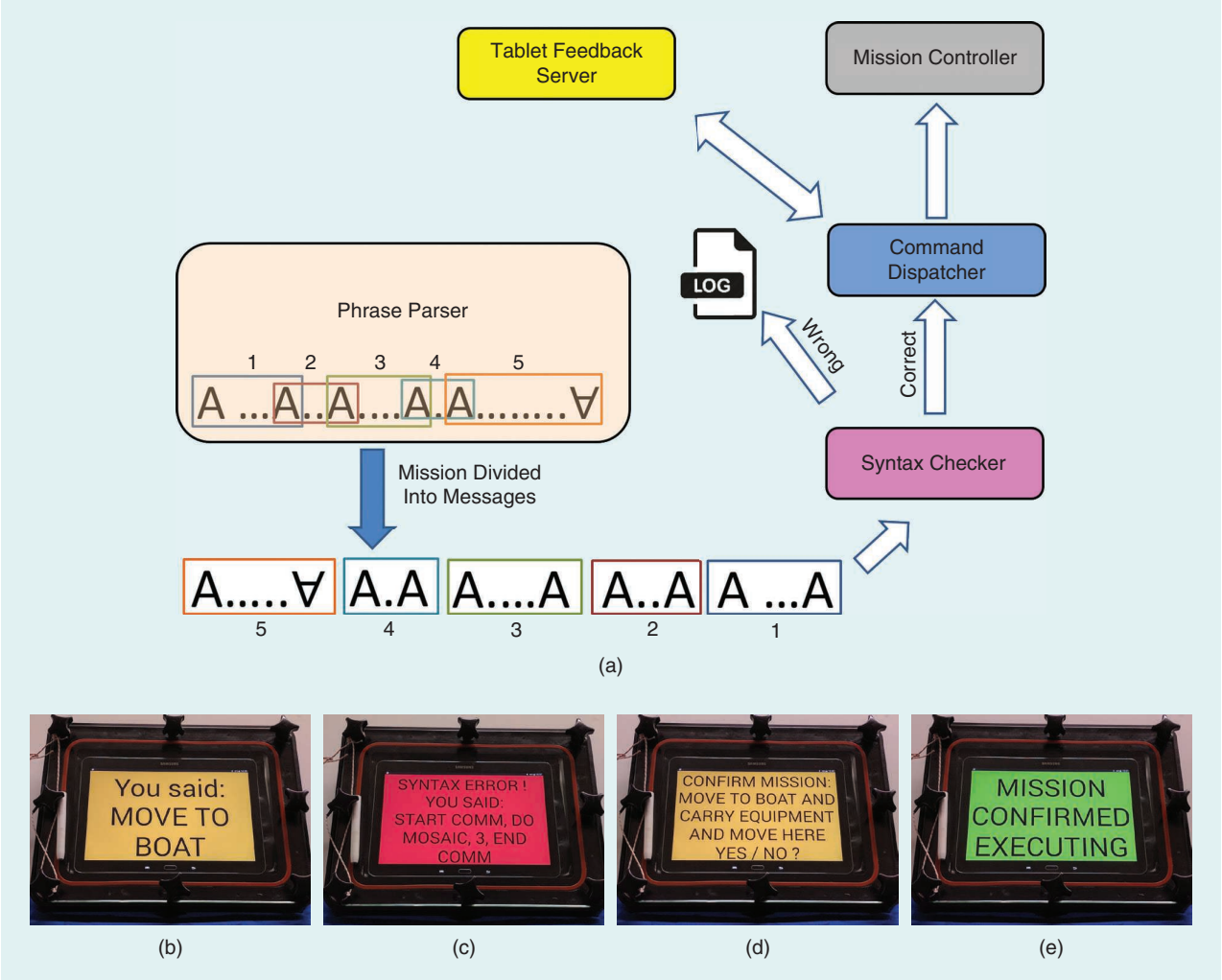


Figure 3. (a) A mission aggregation from single gestures/commands. Examples of feedback given to the diver through an underwater tablet on the AUV: (b) single gesture, (b) syntax error, (c) mission confirmation request, and (d) mission status.

can be challenging to provide sufficient data for the evaluation of underwater vision methods. This especially holds when ML and, in particular DL are employed, where clear and immutable design assumptions cannot be assumed. In addition, there is a need for large amounts of training data for ML and especially DL algorithms. One option is to produce synthetic images in simulations [34], [35]. Another is the use of generative adversarial networks, which have been successfully demonstrated in the context of underwater image enhancement [36]–[38].

Here, we use insights from underwater image formation for artificial image degradation to produce additional data from real-world data. The existing real-world data cover the relevant scenarios, i.e., underwater scenes with divers carrying out realistic tasks, including situations with the use of hand gestures. The artificial degeneration allows us to evaluate the different options for gesture recognition under a very wide range of possible environmental conditions that are unfeasible to cover so broadly with real field trials. In addition, the artificial degeneration allows for evaluation under controllable conditions. Among others, we introduce in the "Geometry-Contextual Perturbations" section a method based on depth information that allows a physically very realistic reduction of visibility conditions. The image degradation

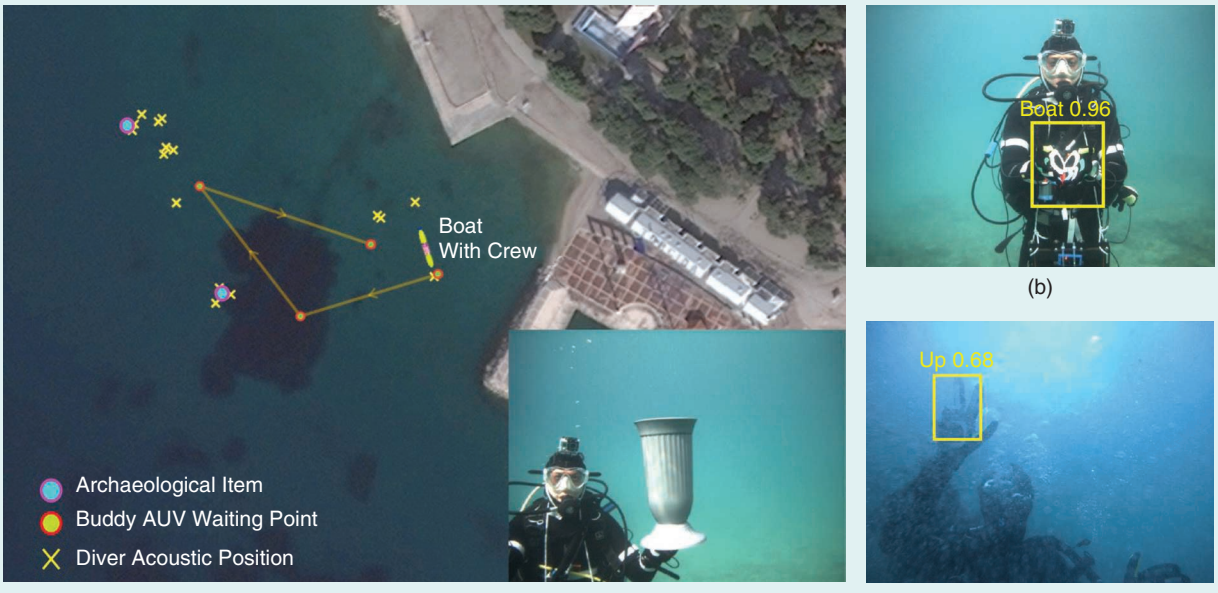


Figure 4. A field trial emulating an archaeological underwater mission in Biograd na Moru, Croatia. One task includes the transport of an object found by the diver, here a mock-up amphora, by the AUV to a boat. (a) The mission layout and archaeological item to be retrieved, (b) the gesture "boat" recognized, and (c) the gesture "photo" not recognized.

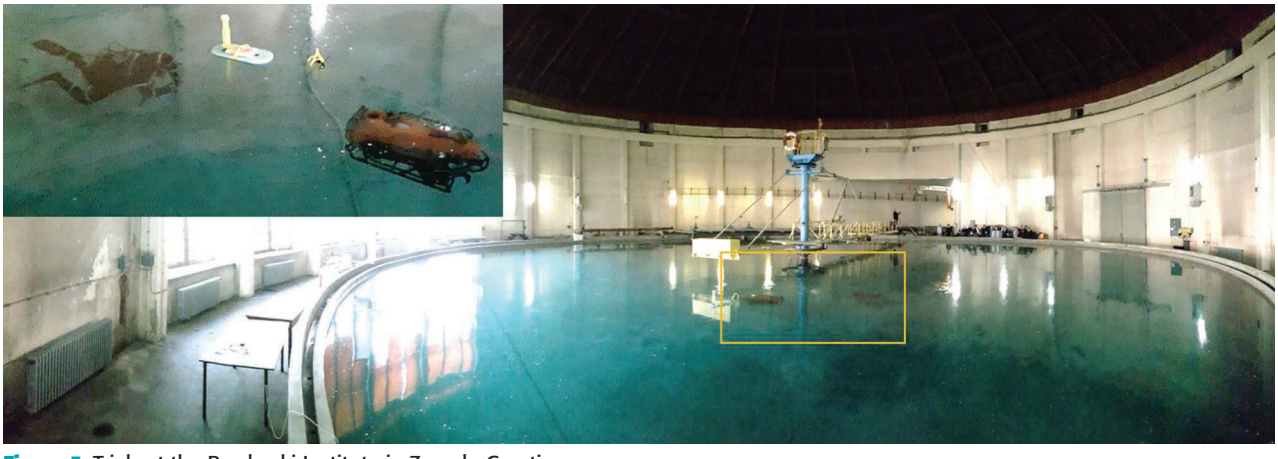


Figure 5. Trials at the Brodarski Institute in Zagreb, Croatia.

can also be used in other applications of underwater vision. This includes the production of additional training data for ML methods, including DL methods.

Several different image degradation methods are considered. The first group of transformations, named *pixel-based perturbations*, requires information from only a single monocular image and transforms only pixel values; i.e., all operations are constrained to the image domain. The second group of geometry-contextual perturbations uses the 3D scene geometry information obtained from stereo imagery [39] to compute the depth relative to the camera and, in turn, to render a more detailed simulation of underwater light backscattering effects. Figure 6 depicts examples of each type of distortion. The code for the degradations is available at https://github.com/arturokkboss33/caddy-underwater-diver-classification.

Pixel-Based Perturbations

Gaussian Blur

The image is blurred to approximate the effects caused by moving objects, sediment clouds, material on the lens/housing, misalignment of the camera with respect to the housing window, or incorrect focus caused by light forward scattering. This is done using a Gaussian kernel with standard deviation σ and size k_s pairs: $\{(1.5, 9)(3, 17)\}$.

Brightness Shift

For shallow water operations (depth <15 m), the ambient light can drastically change the brightness of the image depending on the weather conditions and time of the day. To simulate this, a scaling factor b is applied to each image channel with respective saturation values of $b = \{0.5, 2\}$.

White Balance

White balance is considered because it can lead to unexpected image artifacts. White balancing methods are typically based on the assumption that there is a minimum range of colors in the scene, including neutral (white) colors. But, when there are large regions with uniform color in the scene (water medium), this can shift the color correction to more blueish or reddish colors. Thus, if the white balance is not properly configured, e.g., if a standard in-air method is used, it can degrade the quality of the image.

To reflect this, a gray-world (GW) white balance is applied that assumes the average of all channels should result in a gray image. It requires a saturation threshold $t_{\text{GW}} = 0.7$. Not all normalized pixels above this value are used during the color correction process. Another method, denoted as simple white balance (SWB), just stretches each input channel to generate similar ranges for each channel. It uses a threshold $t_{\text{SWB}} = 5\%$ to ignore the top and bottom 5% of pixels.

Underwater Alpha Blend

The image I is blended with a background image of uniform color H that represents simple underwater haze effects. This image operation, known as *alpha blend*, is defined as $A = H \times \alpha + I \times (1 - \alpha)$. The blending coefficients used here are $\alpha = \{0.25, 0.5\}$. Typically, a gray color for H is used based on the color of fog on land. However, to emulate underwater haze with higher fidelity, the Jerlov water types [33] are used with their associated light downwelling and backscattering attenuation factors to tune H to a more realistic color.

For our experiments, an ambient light at depth d of 10 m is assumed. Jerlov water types $w = II$, $1C$ are considered, i.e., murky oceanic water and coastal water with low amounts of sediment. Based on our experience, these offer challenging visibility conditions but are within the operational range for

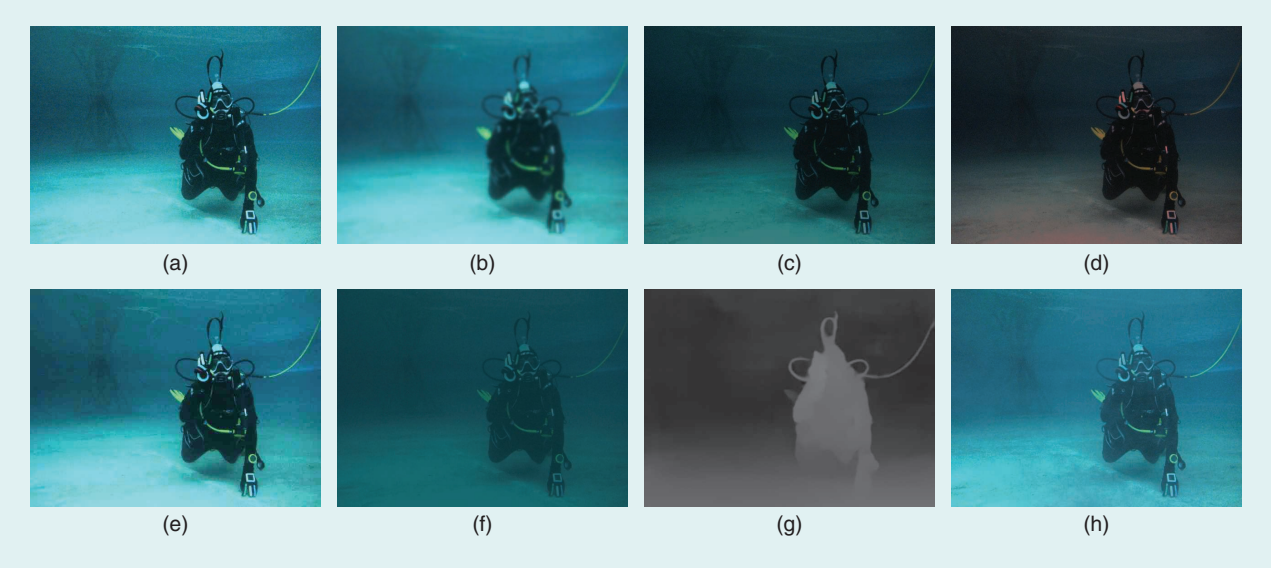


Figure 6. Examples of underwater image perturbations: (a) the original image, with (b)–(f) pixel-based and (g)–(h) geometry-contextual perturbations. (b) Gaussian blur, $\sigma = 3$; (c) brightness shift, $b = 0.5$; (d) gray-world white balance, $t = 0.7$; (e) JPEG compression, $q = 20\%$; (f) alpha blend, $w = 1C$, $\alpha = 0.25$; (g) depth map from stereo; and (h) underwater haze.

divers. With these values of α, w, and d, values for H are computed based on the Jerlov classification.

Image/Video Compression

Underwater robots are employed in practice in a wide range of applications for a wide range of different tasks, even within a particular application. Hence, different bandwidth values or CPU resources are typically allocated to each system component depending on the mission and the task within the mission. It is, hence, a common practice that compression algorithms are applied to the images (specifically, video frames) during real missions to free resources for other processes as well as for data storage, e.g., for data transmission in the case of remotely operated vehicles. Often, motion JPEG is used to optimize for coding speed and frame-by-frame quality over bitrate. To study the image degradation effects, compression quality values of $q = \{60, 20\}$ are used here.

Data were recorded in the open sea as well as in indoor and outdoor pools at three different locations, namely, in Biograd na Moru (Croatia), at the Brodarski Institute in Zagreb (Croatia), and in Genova (Italy).

Geometry-Contextual Perturbations

In underwater environments, ambient light attenuates exponentially with depth d and even further with the distance z between the target object (here, the diver) and the observer (here, the camera on the AUV). However, the attenuation factor $K(d)$ due to the depth can typically be ignored because the attenuation factor $\beta(\lambda, z)$ due to distance z and wavelength λ is 2–5 times greater [33]. Note that we can typically assume an observer–object viewing direction of approximately $\theta = 90^\circ$ (Figure 1).

To obtain z or the depth relative to the image, DispnetC [40] is used. It is a 100% dense disparity estimator with $\approx 4\%$ error in the KITTI Stereo 2015 benchmark. This accuracy is more than enough for our purposes, and the method has proven to perform well in underwater scenarios [41]. Nonetheless, the estimated z is refined through a bilateral filter to keep the image edge consistency [Figure 6(g)]. Based on this value, the geometry-contextual image transformations presented in the following sections can be applied.

Underwater Haze

A popular haze model used in terrestrial robotics is based on the following equation:

$$I(x) = J(x)\,t(x) + B(1 - t(x)), \tag{1}$$

$$t(x) = e^{-\beta(\lambda) z(x)}, \tag{2}$$

where $I(x)$ is the image received by the camera sensor and $J(x)$ is the original image (scene radiance), which is exponentially attenuated by the transmission matrix $t(x)$ at every pixel x as the range (distance to object) increases and depends on the wavelength λ. B is the ambient light. As mentioned previously, $z(x)$ is computed here by DispnetC. It is refined with a bilateral and a Gaussian filter to avoid discontinuity effects. But a physically realistic haze model is more complex in the underwater case [33]. In summary, a different transmission matrix is needed for each $J(x)$ and B,

$$I(x) = J(x)\,t_J(x) + B(1 - t_B(x)), \tag{3}$$

$$t_J(x) = e^{-\beta(\lambda) z(x)}, \tag{4}$$

$$t_B(x) = e^{-\beta(w,d) z(x)}. \tag{5}$$

The underwater haze model from (3) is, hence, combined with the range map $z(x)$. This allows us to apply systematic and controlled image degradations to real-world data in the form of physically realistic underwater haze.

For the values of B and its corresponding $t_B(x)$, the same values as for the underwater alpha blend $\alpha = 0.25$ are used (see the "Pixel-Based Perturbations" section). For $t_J(x)$, attenuation coefficients $\beta(\lambda)$ are chosen to allow for visibility within a distance of approximately 10 m, which can be considered to be a reasonable maximum operational distance in U-HRI. In terms of Jerlov water types, this corresponds to $\beta = [0.5, 0.15, 0.90]$ for the red, green, and blue channels, respectively.

Experiments and Results

Data Set and Setup

The presented approach to U-HRI originated within the EU project CADDY. Major efforts were devoted in this project to the collection of data, including experiments on the use of underwater gestures. Data were recorded in the open sea as well as in indoor and outdoor pools at three different locations, namely, in Biograd na Moru (Croatia), at the Brodarski Institute in Zagreb (Croatia), and in Genova (Italy). The data are divided into eight scenarios representing different diver missions and field experiments. The scenarios, named Biograd-A, Biograd-B, and Genova-A, represent trials that were mainly organized for data collection; they, hence, feature a high number of samples. The other scenarios, Biograd-C and Brodarski-A–D, cover experimental or real diver missions. A detailed discussion of the number of samples and environmental conditions of each scenario is provided in [39]. For the evaluation of the different ML methods here, they are trained according to the partition of the data detailed in Table 2.

Each method described in the "Gesture Detection and Classification" section has four Model X versions. The partition is made to gather samples with similar environmental conditions (e.g., location, light, and so on) and observe how the methods perform against unseen types of data. Samples from Biograd-C and Brodarski-B and -D are used only as test sets.

The complete data set contains 18,478 images (9,239 stereo pairs) that represent 16 gesture classes. A split of 80/20% for training and validation sets is used for each model, except for Model F. This splitting criterion is applied to each gesture class. The test sets comprise all scenarios that are not included in the training. All of the classifiers, with the exception of Model C, have samples of all gestures. Model C, being trained only on the Brodarski-A and -C scenarios, is trained and tested only on nine gestures. Model F (F stands for "full") is trained with samples from all of the scenarios according to a standard split of 70/20/10% following the data distribution per scenario. As mentioned previously, the data and their distribution are described in detail in [39].

Setup of the ML Methods

The following settings are used for the ML methods that are evaluated here as possible approaches for underwater gesture recognition. For the classical ML approach, i.e., MD-NCMF [42], 15 trees are used for the ensemble forest. The tree branches stop splitting when the number of samples is 20 or fewer to avoid overfitting. Each node has a subset of feature centroids of three. The engineered features used are ORB, Harris corners, edge-based regions, difference of Gaussians, Harris affine Laplace, and DAISY.

For the fully connected CNN (FC-CNN) with ResNet-50, the default parameters of [30] are not strictly followed. The reason is to train the network using a cyclic learning rate implemented in the Fast.ai library, which has been found in the literature to yield better results. The other parameters are set as follows: $epochs = 10$, maximum learning rate $m\_lr = 10e^{-2}$, and batch size $bs = 32$.

For SSD with MobileNets, faster R-CNN with ResNet-101 and deformable faster R-CNN, the default parameters of their respective publications and the related source code are used. The training convergence only is monitored to choose the training iteration with the best validation performance. Likewise, a minimum intersection over union of 0.5 is set.

Table 2. Partitioning of the data for training (mean and median samples are provided per class).

	Model A	Model B	Model C	Model F
Training Sets	**Biograd A, B**	**Genova A**	**Brodarski A, C**	**All Scenarios**
Sample mean	338	415	[222]	1,156
Sample median	151	294	[206]	792

Table 3. The accuracy (0 → 1) of the visual models in all scenarios according to Table 2.

	MD-NCMF				FC-CNN With ResNet-50			
	Model A	**Model B**	**Model C**	**Model F**	**Model A**	**Model B**	**Model C**	**Model F**
Biograd-A	0	0.72	0.52	0.81	0	0.42	0.5	0.99
Biograd-B	0	0.71	0.51	0.84	0	0.21	0.51	0.99
Biograd-C	0.74	0.75	0.68	0.85	0.53	0.45	0.51	0.97
Brodarski-A	0.76	0.76	0	0.78	0.52	0.24	0	0.95
Brodarski-B	0.81	0.79	0.71	0.73	0.63	0.12	0.68	0.86
Brodarski-C	0.7	0.65	0	0.77	0.57	0.48	0	0.98
Brodarski-D	0.69	0.61	0.55	0.71	0.71	0.48	0.53	1
Genova-A	0.52	0	0.48	0.69	0.34	0	0.24	0.89
All scenarios	0.56	0.64	0.46	0.77	0.45	0.36	0.43	0.95

	SSD With MobileNets				Faster R-CNN With ResNet-101				Deformable Faster R-CNN			
	Model A	**Model B**	**Model C**	**Model F**	**Model A**	**Model B**	**Model C**	**Model F**	**Model A**	**Model B**	**Model C**	**Model F**
Biograd-A	0	0.35	0.38	0.84	0	0.63	0.65	0.99	0	0.65	0.64	0.99
Biograd-B	0	0.29	0.44	0.88	0	0.51	0.71	0.99	0	0.54	0.7	1
Biograd-C	0.36	0.31	0.4	0.82	0.74	0.58	0.67	0.98	0.74	0.57	0.67	0.98
Brodarski-A	0.38	0.3	0	0.87	0.72	0.48	0	0.97	0.73	0.49	0	0.97
Brodarski-B	0.33	0.29	0.48	0.84	0.72	0.52	0.85	0.96	0.74	0.5	0.87	0.97
Brodarski-C	0.32	0.28	0	0.86	0.79	0.56	0	0.99	0.78	0.6	0	0.99
Brodarski-D	0.29	0.26	0.36	0.79	0.82	0.55	0.68	0.99	0.84	0.54	0.72	0.99
Genova-A	0.25	0	0.23	0.75	0.69	0	0.44	0.94	0.66	0	0.41	0.96
All scenarios	0.28	0.361	0.29	0.85	0.59	0.49	0.52	0.98	0.61	0.5	0.53	0.98

All methods are real-time capable. Their runtimes are so small that the differences among them are completely negligible compared to the computation needs of all of the other processes running on the system during a mission.

Baseline Performance With Only Real-World Data Training

In this experiment, the models trained according to Table 2 are evaluated using the original data without artificial image perturbations. The results with respect to accuracy are detailed in Table 3. Deformable faster R-CNN and faster R-CNN have the lead when the complete data set is used (Model F), followed by FC-CNN, with an accuracy of 95%. This is an indication that, if the amount and variance of the data are high, direct classifiers offer top performance, which can save effort and time dedicated to manually segmenting object regions on the images. SSD MobileNet still has a better performance than the MD-NCMF as a classical ML approach, but it drops below 90%. Note that SSD is mainly known for its superior speed and suitedness for embedded systems. MD-NCMF ranks last, with an accuracy below 80%.

We can conclude that haze effects, which are the most typical natural underwater phenomena, are really important to consider when designing an underwater object detector.

For Models A–C, which are trained with specific scenario data, it can be seen that the performance drastically changes. More precisely, the following observations can be made.

Deformable and standard faster R-CNN still have the lead (except for Model B), but MD-NCMF as a classical method offers competitive results, and it outperforms FC-CNN and SSD MobileNets. Thus, deep visual models suffer a great performance drop, namely, $\approx 40\%$, while MD-NCMF drops only $\approx 20\%$. This strongly indicates that DL techniques are highly dependent on the amount of data and how representative they are of the real-world class distribution.

For Model B versions, MD-NCMF performs better than the rest. A reasonable explanation for this cannot be made without a close examination of the data and a visualization of the learned features by the deep models. It can be assumed that the data used to train Model B, i.e., from Genova-A, are not sufficient for the deep models to learn strong features despite providing more samples per class than Models A and B (see Table 2), and the human-engineered features used for MD-NCMF are simply more representative.

The classical visual model provides a more stable performance across the test sets. The most representative example is when Model C versions are benchmarked against Genova-A samples; accuracy goes down for all of the methods but especially for the DL-based ones. So deep models have strong performance drops for particular tests; this holds

Table 4. The accuracy of each visual model under all image perturbations. In addition to the numerical value of accuracy shown in each cell, the cell color (0 ▬ 1) illustrates the normalized value relative to the baseline accuracy to highlight performance variations and robustness.

	MD-NCMF	FC-CNN With ResNet-50	SSD With MobileNets	Faster R-CNN With ResNet-101	Deformable Faster R-CNN
Baseline	**0.77**	**0.95**	**0.85**	**0.98**	**0.98**
Blur ($\sigma = 1.5$)	0.61	0.8	0.63	0.85	0.95
Blur ($\sigma = 3$)	0.19	0.61	0.28	0.65	0.7
Brightness ($b = 0.5$)	0.63	0.76	0.69	0.93	0.95
Brightness ($b = 2$)	0.49	0.47	0.4	0.77	0.88
White balance (GW, $t = 0.7$)	0.11	0.48	0.46	0.79	0.82
White balance (SWB, $t = 0.5$)	0.73	0.86	0.79	0.94	0.96
JPEG compression ($q = 60$)	0.7	0.92	0.81	0.96	0.98
JPEG compression ($q = 20$)	0.4	0.83	0.73	0.87	0.91
Underwater alpha blend ($w = II$, $d = 10$, $\alpha = 0.5$)	0.29	0.38	0.31	0.91	0.96
Underwater alpha blend ($w = II$, $d = 10$, $\alpha = 0.25$)	0.07	0.3	0.28	0.82	0.89
Underwater alpha blend ($w = 1C$, $d = 10$, $\alpha = 0.5$)	0.24	0.33	0.26	0.85	0.95
Underwater alpha blend ($w = 1C$, $d = 10$, $\alpha = 0.25$)	0.03	0.17	0.17	0.76	0.87
Haze ($w = II$, $\beta_{R,G,B} = [0.5, 0.15, 0.90]$)	0.42	0.49	0.4	0.85	0.95
Haze ($w = 1C$, $\beta_{R,G,B} = [0.5, 0.15, 0.90]$)	0.15	0.33	0.26	0.79	0.89

especially true for FC-CNN. Our hypothesis is that this is the case because FC-CNN is the only method without a region proposal step within its architecture that helps in refining the classification process.

Robustness Under Artificial Image Perturbations

Table 4 displays the performance of all of the visual models tested with samples on which the image perturbations described in the "Physically Realistic Underwater Image Degradation" section are applied. Only Model F versions are evaluated, i.e., visual models trained with the complete data set. Their baseline accuracy from the previous experiment in the "Baseline Performance With Only Real-World Data Training" section is shown for comparison. Based on this, Table 4 gives the numerical values of the absolute accuracies as well as the normalized accuracies with respect to the baseline performance in the form of a color code.

Note that the models are trained with the original sensor images and that none of the image degradations are used to augment the training data. Deformable and standard faster R-CNN show good robustness against the majority of the degradations except for high levels of Gaussian blur, which affects all of the other models as well. They also both exhibit similar performance drops for every image degradation.

The performance of the rest of the models degrades more substantially, especially from haze effects, which are emulated by alpha blend and our proposed method for producing artificial underwater haze using range information. MD-NCMF is completely ineffective at high levels of alpha blend. We can conclude that haze effects, which are the most typical natural underwater phenomena, are really important to consider when designing an underwater object detector.

For the deep visual models, JPEG compression has almost no effect. This holds even at a very low quality level of 10%. GW white balance especially affects FC-CNN and SSD, indicating that the GW assumption is tailored toward terrestrial robotics and that users have to pay attention to camera presets for underwater applications. Increasing the brightness has a bigger effect than lowering it, as saturation levels may be reached more quickly. As mentioned previously, significant blur affects all of the models. MD-NCMF is affected by almost all of the image perturbations, which supports the idea that DL approaches learn important strong features given enough data, which may be hard for a human to mathematically and algorithmically conceptualize.

Conclusions

A fully integrated approach to U-HRI was presented. It features a front end for gesture recognition combined with a back end with an interpreter for a language derived from the existing standard gestures for communication between human divers. The approach enables a diver to communicate commands as well as complex mission specifications to an underwater robot via gestures.

The wide range of environmental conditions, especially with respect to visibility conditions, poses a severe challenge for underwater vision in general and gesture recognition in particular. Hence, different ML methods in the form of four DL approaches and a more classical ML method are investigated with respect to their robustness for gesture recognition. In addition to the exhaustive test with real-world data from different tests in pools and during field trials, artificially degraded image data are used. To this end, we presented, among others, a physically realistic way to use range information for adding underwater haze in controlled ways.

References

[1] B. Verzijlenberg and M. Jenkin, "Swimming with robots: Human robot communication at depth," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 4023–4028.

[2] A. Speers, P. M. Forooshani, M. Dicke, and M. Jenkin, "Lightweight tablet devices for command and control of ROS-enabled robots," in *Proc. 16th Int. Conf. Adv. Robot. (ICAR)*, 2013, pp. 1–6.

[3] J. Sattar and G. Dudek, "Where is your dive buddy: Tracking humans underwater using spatio-temporal features," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS 2007)*, 2007, pp. 3654–3659.

[4] J. Sattar and G. Dudek, "Underwater human–robot interaction via biological motion identification," in *Proc. Robot., Sci. Syst. (RSS)*, 2009, pp. 1–8.

[5] H. Bülow and A. Birk, "Diver detection by motion-segmentation and shape-analysis from a moving vehicle," in *Proc. IEEE Oceans*, 2011, pp. 1–6.

[6] K. J. DeMarco, M. E. West, and A. M. Howard, "Sonar-based detection and tracking of a diver for underwater human–robot interaction scenarios," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, 2013, pp. 2378–2383.

[7] K. J. DeMarco, M. E. West, and A. M. Howard, "Autonomous robot-diver assistance through joint intention theory," in *Proc. IEEE Oceans*, 2014, pp. 1–5.

[8] A. G. Chavez, M. Pfingsthorn, A. Birk, I. Rendulic, and N. Miskovic, "Visual diver detection using multi-descriptor nearest-class-mean random forests in the context of underwater human robot interaction (HRI)," in *Proc. IEEE Oceans*, 2015, pp. 1–6.

[9] A. G. Chavez, C. A. Mueller, A. Birk, A. Babic, and N. Miskovic, "Stereo-vision based diver pose estimation using LSTM recurrent neural networks for AUV navigation guidance," in *Proc. IEEE Oceans*, 2017, pp. 1–6.

[10] Y. Xia and J. Sattar, "Visual diver recognition for underwater human-robot collaboration," in *Proc. Int. Conf. Robot. Automat. (ICRA)*, 2019, pp. 6839–6845.

[11] M. J. Islam, M. Fulton, and J. Sattar, "Toward a generic diver-following algorithm: Balancing robustness and efficiency in deep visual detection," *IEEE Robot. Automat. Lett. (RAL)*, vol. 4, no. 1, pp. 113–120, 2019. doi: 10.1109/LRA.2018.2882856.

[12] W. Remmas, A. Chemori, and M. Kruusmaa, "Diver tracking in open waters: A low-cost approach based on visual and acoustic sensor fusion," *J. Field Robot.*, vol. 38, no. 3, 2020. doi: 10.1002/rob.21999.

[13] K. d. Langis and J. Sattar, "Realtime multi-diver tracking and re-identification for underwater human-robot collaboration," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2020, pp. 11,140–11,146.

[14] K. J. DeMarco, M. E. West, and A. M. Howard, "A simulator for underwater human-robot interaction scenarios," in *Proc. IEEE Oceans*, 2013, pp. 1–10.

[15] M. Fulton, C. Edge, and J. Sattar, "Robot communication via motion: Closing the underwater human-robot interaction loop," in *Proc. Int. Conf. Robot. Automat. (ICRA)*, 2019, pp. 4660–4666.

[16] G. Dudek, J. Sattar, and A. Xu, "A visual language for robot control and programming: A human-interface study," in *Proc. IEEE Int. Conf. Robot. Automat.*, Apr. 2007, pp. 2507–2513. doi: 10.1109/ROBOT.2007.363842.

[17] H. Bülow and A. Birk, "Gesture-recognition as basis for a human robot interface (HRI) on a AUV," in *Proc. IEEE Oceans*, 2011, pp. 1–6.

[18] F. Gustin, I. Rendulic, N. Miskovic, and Z. Vukic, "Hand gesture recognition from multibeam sonar imagery," in *Proc. 10th IFAC Conf. Control Appl. Marine Syst. (CAMS)*, 2016, vol. 49, pp. 470–475.

[19] A. G. Chavez and A. Birk, "Underwater gesture recognition based on multi-descriptor random forests (MD-NCMF)," Progress Report: EU-FP7 Cognitive autonomous diving buddy (CADDY), 2015. [Online]. Available: http://robotics.jacobs-university.de/node/443

[20] D. Chiarella et al., "A novel gesture-based language for underwater human–robot interaction," *J. Marine Sci. Eng.*, vol. 6, no. 3, p. 91, 2018. doi: 10.3390/jmse6030091.

[21] D. Chiarella et al., "Gesture-based language for diver-robot underwater interaction," in *Proc. IEEE Oceans 2015 – Genova*, May 2015, pp. 1–9. doi: 10.1109/OCEANS-Genova.2015.7271710.

[22] N. Miskovic et al., "Caddy project, year 3: The final validation trials," in *Proc. IEEE Oceans*, 2017, pp. 1–5.

[23] N. Miskovic et al., "Caddy project, year 2: The first validation trials," in *Proc. 10th IFAC Conf. Control Appl. Marine Syst. (CAMS)*, 2016, pp. 1–6.

[24] M. J. Islam, M. Ho, and J. Sattar, "Understanding human motion and gestures for underwater human–robot collaboration," *J. Field Robot.*, vol. 36, no. 5, pp. 851–873, 2019. doi: 10.1002/rob.21837.

[25] W. Liu et al., "SSD: Single shot multibox detector," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 21–37.

[26] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, June 2017. doi: 10.1109/TPAMI.2016.2577031.

[27] R. Codd-Downey and M. Jenkin, "Human robot interaction using diver hand signals," in *Proc. 14th ACM/IEEE Int. Conf. Human–Robot Interaction (HRI)*, 2019, pp. 550–551.

[28] R. Codd-Downey and M. Jenkin, "Finding divers with scubanet," in *Proc. Int. Conf. Robot. Automat. (ICRA)*, 2019, pp. 5746–5751.

[29] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recogn (CVPR)*, 2018, pp. 4510–4520.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recogn. (CVPR)*, June 2016. doi: 10.1109/cvpr.2016.90.

[31] A. G. Howard et al., "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017. [Online]. Available: http://arxiv.org/abs/1704.04861

[32] J. Dai et al., "Deformable convolutional networks," 2017. [Online]. Available: http://arxiv.org/abs/1703.06211

[33] D. Akkaynak and T. Treibitz, "A revised underwater image formation model," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recogn.*, June 2018, pp. 6723–6732.

[34] M. O'Byrne, V. Pakrashi, F. Schoefs, and B. Ghosh, "Semantic segmentation of underwater imagery using deep networks trained on synthetic imagery," *J. Marine Sci. Eng.*, vol. 6, no. 3, p. 93, 2018. doi: 10.3390/jmse6030093.

[35] Y. Hu, K. Wang, X. Zhao, H. Wang, and Y. Li, "Underwater image restoration based on convolutional neural network," in *Proc. 10th Asian Conf. Mach. Learn.*, 2018, vol. 95, pp. 296–311.

[36] J. Li, K. A. Skinner, R. M. Eustice, and M. Johnson-Roberson, "Watergan: Unsupervised generative network to enable real-time color correction of monocular underwater images," *IEEE Robot. Automat. Lett.*, vol. 3, no. 1, pp. 387–394, 2018.

[37] C. Fabbri, M. J. Islam, and J. Sattar, "Enhancing underwater imagery using generative adversarial networks," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2018, pp. 7159–7165. doi: 10.1109/ICRA.2018.8460552.

[38] D. G. Kim and S. M. Kim, "Single image-based enhancement techniques for underwater optical imaging," *J. Ocean Eng. Technol.*, vol. 34, no. 6, pp. 442–453, 2020. doi: 10.26748/KSOE.2020.030.

[39] A. Gomez Chavez, A. Ranieri, D. Chiarella, E. Zereik, A. Babić, and A. Birk, "Caddy underwater stereo-vision dataset for human–robot interaction HRI in the context of diver activities," *J. Marine Sci. Eng.*, vol. 7, no. 1, pp. 1–14, 2019. doi: 10.3390/jmse7010016.

[40] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," 2016, arXiv: 1512.02134.

[41] A. Gomez Chavez, Q. Xu, C. A. Mueller, S. Schwertfeger, and A. Birk, "Adaptive navigation scheme for optimal deep-sea localization using multimodal perception cues," 2019, arXiv: 1906.04888.

[42] A. G. Chavez, M. Pfingsthorn, A. Birk, I. Rendulić, and N. Misković, "Visual diver detection using multi-descriptor nearest-class-mean random forests in the context of underwater human robot interaction (HRI)," in *Proc. IEEE Oceans 2015 – Genova*, May 2015, pp. 1–7. doi: 10.1109/OCEANS-Genova.2015.7271556.

Arturo Gomez Chavez, Jacobs University Bremen, Bremen, 28759, Germany. Email: a.gomezchavez@jacobs-university.de.

Andrea Ranieri, Institute of Applied Mathematics and Information Technology, National Research Council of Italy, Pavia, 27100, Italy. Email: andrea.ranieri@cnr.it.

Davide Chiarella, Institute for Computational Linguistics A. Zampolli, National Research Council of Italy, Pisa, 56124, Italy. Email: davide.chiarella@cnr.it.

Andreas Birk, Jacobs University Bremen, Bremen, 28759, Germany. Email: a.birk@jacobs-university.de.