Toward Visual Communication Methods for Underwater Human-Robot Interaction

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Chelsey Edge

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

Advisor: Junaed Sattar

April, 2024

Acknowledgements

First I would like to express my sincere gratitude to my advisor, Dr. Junaed Sattar, who believed I could succeed in this program without a traditional CS background. Thank you for your encouragement and guidance throughout my time in the Ph.D. program. Your interest in and excitement about your work has been truly motivational, I hope I can provide the same to my own future students.

I would also like to extend my gratitude to my committee members: Dr. Anand Tripathi, Dr. Victoria Interrante, and Dr. Daniel Kersten. Thank you for providing insights and expertise throughout my preliminary exams, proposal, and final dissertation. Your feedback and encouragement throughout the process has been greatly appreciated.

Thank you to Dr. Shana Watters and Kate Jensen for the many times you welcomed me when I dropped by to chat about teaching, research and life in general. Your advice and support has been invaluable.

Many thanks to all my fellow lab members, past and present. My fellow Ph.D. students: Dr. Jahidul Islam, Dr. Jiawei Mo, Dr. Michael Fulton, Dr. Jungseok Hong, Dr. Sadman Sakib Enan, Corey Knutson, Demetri Kutzke, Sakshi Singh, David Widhalm, Maryam Kameli, and Karin de Langis. I thank you for not only your enthusiasm for academic collaborations, but also for your friendship and willingness to drop what you are doing and grab some lunch. Masters students: Cory Ohnstead for all his help diving during field trials and Andrea Walker for helping with the pointing gesture project even after her graduation. Undergraduate students who chose to spend their time helping with research: Hannah DuBois, Kevin Orpen, Kimberly Barthelemy, Christopher Morse, Preeti Pidatala, Sohan Addagudi, Megdalia Bromhal, and many others whose help at pool trials was essential.

My deepest gratitude to my family and friends for their love and support through the

entire Ph.D. program. Thank you for helping me maintain balance and while supporting my academic goals through everything. And finally, thank you, Robert, for convincing me to apply to the Computer Science program and for your encouragement along the way. These years have been amazing and I am looking forward to the future.

Dedication

To my family for their unwavering support.

Abstract

Trained divers take on the complex and often dangerous underwater environment to perform essential tasks. These tasks include inspection and repair of underwater infrastructure and monitoring the health of water systems through tasks such as observations of coral reefs and tracking of invasive species. Autonomous Underwater Vehicles (AUVs) able to assist with these tasks have become more widely deployed as their capabilities improve, however, when deployed as solo agents they lack the intuition and ability to adapt to unexpected situations as a human diver would. The objective of collaboration between a diver and an AUV brings together the ability of an AUV to perform tasks that are dangerous to the human diver, while maintaining the ability of the diver to monitor the situation and update task information as necessary. For this collaboration to be successful meaningful communication is essential, especially when the goal of the collaboration is to complete a task.

This dissertation presents our work towards improving diver-AUV collaboration, focusing on utilizing visual perception onboard the AUV. In the following chapters, we discuss two novel communication algorithms that allow divers to communicate information about the location of an object required by an AUV to perform a task. These methods have been designed to take into account challenges such as limitations of onboard computation as well as challenges inherent to working in the underwater domain, such as non-traditional human body poses and limitations of traditional, terrestrial, computer vision. Evaluations of these methods are performed onboard AUVs. We then incorporate these algorithms into a communication system which allows a diver to assign the AUV a task based on the object detected. This system also provides feedback from the AUV to the diver about the task which will be performed, forming a closed loop communication system between diver and AUV. Validation of this system was performed fully onboard an AUV in the Caribbean Sea. In addition, as AUV visual perception can be hampered by the visual degradation of the underwater environment, we therefore present an investigation into a task-based method to improve AUV vision. We also discuss our contributions to the design and creation of the research platforms necessary for this research to move forward.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

For thousands of years, humans have taken on the complex and often dangerous underwater environment for exploration, economic, industrial, and recreational purposes. There now exists vast underwater infrastructure for commercial domains such as communication and oil industries in addition to other surface constructs such as bridges and ships that connect the world. The study and conservation of underwater environments such as coral reefs and even submerged sites of ancient human civilizations has become more feasible as the ability to explore deeper and for longer time periods has improved. The utilization of Remotely Operated and Autonomous Underwater Vehicles (ROVs and AUVs) for completing tasks has been seen as a way to make activities such as inspection and maintenance of oil pipeline inspection [5], biological monitoring [6], and archaeology [7] safer and more efficient. ROVs are controlled, usually through a tether, to gather information or perform tasks. However, there is a general disconnect between the vehicle and the operator as the operator can view the environment through limited onboard sensors and then react based on this information. Environmental aspects unseen by the ROV as well as tangling of the tether are additional shortcomings. AUVs used without a collaborative diver are limited to a specific task designation, often programmed before the AUV is put in the water. A change in environmental circumstances or anything unexpected that requires *in situ* mission re-calibration is very challenging as communication is limited to acoustic and very low-bandwidth electromagnetic (EM) channels, which limits the amount and type of information that can be exchanged. A

Figure 1.1: A diver is providing instruction to an AUV through the use of a pointing gesture during an open-water field trial in the Caribbean Sea.

third paradigm (beyond 'pure' ROV and AUV use) of underwater robotics shows potential to bridge the gap between ROVs and fully autonomous AUVs. In this situation, an experienced, and likely non-roboticist, diver can work in collaboration with an AUV to provide technical expertise and direction while taking advantage of the AUV for safety, performing routine tasks or time-consuming measures. However, for these collaborations to exist, natural and easy-to-decipher communication between the diver and AUV must occur. Underwater human-robot interaction (U-HRI) as well as collaboration between divers and AUVs is a very open research problem, partially because of the challenges perpetuated in the underwater environment.

In this dissertation, we present our work toward improving visual communication for divers and Autonomous Underwater Vehicles for underwater task-based collaboration. Our methods focus on natural human communication, primarily through pointing (See Fig 1.1), to facilitate ease of use in an environment where attention needs to be focused on other, often critically essential, tasks, and a diver should not need to expend extra mental or physical energy to work with a collaborative AUV. We focus on communication designed to allow a diver and AUV to complete a task and use scene information to guide the AUV. In addition, we investigate task-based image enhancement to improve perceptual information in situations in which visibility is degraded and vision-based

algorithms onboard AUVs have difficulty computing valid results. The AUV itself must also be compatible with use as a diver companion, and so we provide a description of the platforms used to perform and evaluate this research and include detailed information about HydroEye and LoCO AUV, platforms we developed.

1.1 Research Contributions

Our research thus far has focused on improving visual communication for underwater human-robot interaction. In particular, we focus on natural communication methods and the use of scene information to direct AUVs to task completion and enhance AUV vision for specific task-based purposes. We also investigate the use of task-based image enhancement to improve visual AUV perception and provide details of our involvement in the creation of an AUV platform. A brief description of our contributions that will be expanded upon in the dissertation follows.

- **Two-Dimensional Human-Directed Object Inspection for AUVs:** In this work [8], we contribute the Diver Interest via Pointing (DIP) algorithm, a highly modular method for conveying a diver's area of interest to an AUV using pointing gestures for underwater human-robot collaborative tasks. DIP uses a single monocular camera and exploits human body pose, even with complete dive gear, to extract underwater human pointing gesture poses and their directions. By extracting 2D scene geometry based on the human body pose and density of salient feature points along the direction of pointing, using a low-level feature detector, the DIP algorithm can locate objects of interest as indicated by the diver. DIP makes it possible for scuba divers and swimmers to use directional cues, through pointing, to an AUV for inspection, surveillance, manipulation, and navigation. We examine the elements that make up our method, provide quantitative and qualitative evaluation, and demonstrate AUV actuation based on a diver's pointing gesture in closed-water human-robot collaborative experiments. Our evaluations demonstrate the high efficacy of the DIP algorithm in correctly identifying the direction of a pointing gesture and locating an object within that region of interest. We also show that the findings of the algorithm qualitatively conform with human assessment of pointing gestures, directions, and targets.

- **Three-Dimensional Human-Directed Object Inspection for AUVs:** This work [9] extends Diver Interest via Pointing to three dimensions (DIP-3D), creating a method to relay an object of interest from a diver to an AUV by pointing that includes three-dimensional distance information to discriminate between multiple objects in the AUV's camera image. Traditional dense stereo vision for distance estimation underwater is challenging because of the relative lack of saliency of scene features and degraded lighting conditions. Yet, including distance information is necessary for robotic perception of diver pointing when multiple objects appear within the robot's image plane. We subvert the challenges of underwater distance estimation by using sparse reconstruction of keypoints to perform pose estimation on both the left and right images from the robot's stereo camera. Triangulated pose keypoints, along with a classical object detection method, enable DIP-3D to infer the location of an object of interest when multiple objects are in the AUV's field of view. By allowing the scuba diver to point at an arbitrary object of interest and enabling the AUV to autonomously decide which object the diver is pointing to, this method will permit more natural interaction between AUVs and human scuba divers in underwater-human robot collaborative tasks.

- **Pointing Gesture Based Communication System for Divers and AUVs:** This work provides a crucial component towards closing the loop on dynamic pointing-gesture based diver to AUV communication. The methods described in the previous two works allow an AUV to locate an object that is of interest to the diver. However, as stand-alone communication methods they lack flexibility as the action the AUV takes must be pre-determined before beginning the task and there is no easy method for task-switching if the parameters of the mission change during the dive. Semantic Pointing for Communication (SPOC) bridges this gap by providing a library of pointing-related terms that the diver is able to utilize along with a method such as DIP or DIP-3D to provide for flexible mission planning or multiple objectives during a single mission. SPOC adds a hand gesture recognition module which allows the diver to easily instruct the AUV on the appropriate action to take relating to the specific object of interest. We create an annotated dataset of five pointing related hand gestures to denote tasks an AUV would commonly be instructed to perform, train an instance segmentation model

to recognize the hand gestures, and integrate the trained model with DIP onboard an AUV to provide demonstrations of the full SPOC communication vector in both pool and sea environments.

- **Investigations in Task-Driven Generative Image Enhancement**: In this work [10], we investigate a novel approach to improve diver detection performance for AUVs using a task-driven generative image enhancement model. In particular, we present a model that integrates generative adversarial network (GAN)-based image enhancement with the diver detection task. Our proposed approach restructures the GAN objective function to include information gained from a trained diver detector, to generate images that would enhance detection accuracy in adverse visual conditions. By incorporating the detector output into both the generator and discriminator loss functions while training, our model can focus on enhancing images beyond aesthetic qualities and specifically to improve the AUV's detection of scuba divers. We train our network on a paired dataset of scuba divers, using a diver detector, and demonstrate its potential utility on images collected from oceanic explorations of human-robot teams. Experimental evaluations demonstrate that while currently inconclusive, our approach has merit to potentially improve diver detection performance over several state-of-the-art underwater image enhancement algorithms while retaining aesthetic similarity to the original raw images. Finally, we demonstrate the inference performance of our network on embedded devices to highlight the feasibility of operating onboard mobile robotic platforms.

- **Platforms for U-HRI Research**: In this chapter, we describe the platforms used in our research as well as our contributions towards the hardware needed for AUVs to assist divers with underwater tasks. The three platforms are: 1. HydroEye, a data collection rig containing a stereo camera. 2. Aqua [11],an amphibious AUV utilized for its stereo vision. 3. The Low Cost Open-source Autonomous Underwater Vehicle (LoCO AUV) [12], designed and made open source by members of the University of Minnesota's Interactive Robotics and Vision Lab (IRVLab). We highlight our significant contributions to the creation and validation of the HydroEye and LoCO platforms in this manuscript.

(a) (b)

Figure 1.2: Two examples of common visibility challenges in the underwater environment. Images in (a), taken 5 seconds apart, present color variation due to sudden movement of cloud cover. Image (b) presents an example of turbidity and light reflection in shallow water.

In addition to these contributions, collaborative efforts have allowed us to extend our work into the domains of human-to-robot communication and underwater image segmentation. As they are not central to the dissertation, brief discussions of these research contributions can be found in Appendix A.

1.2 Domain Challenges

Work in the underwater domain provides many unique and varying challenges in terms of furthering robotics research. Many sensor and control channels used in terrestrial and aerial robotics are unavailable or underperform for necessary tasks. Signal attenuation and degradation affect electromagnetic waves and render modalities such as Wi-Fi and Bluetooth unusable except at very short distances, if at all. Acoustic signals can be used, however, sound also suffers from attenuation and degradation in addition to potentially being masked by other sounds such as the diver's breathing. Acoustics also have the potential to disturb the natural environment, which may be counterintuitive to some task objectives. Light absorption, refraction, and attenuation can hamper visual sensors in addition to water turbidity as seen in Fig. 1.2, however methods of underwater image enhancement have been shown to improve vision capabilities for AUVs (See Section 2.4). For these reasons, our work can use cameras, monocular and stereo, as the main sensor for the AUV.

1.3 Document Overview

Overall, this dissertation presents multiple methods to improve visual communication between divers and AUVs. The rest of the document is organized as follows. Chapter 2 presents background and other works related to the threads of research presented in this dissertation. Then, Chapter 3 introduces a novel communication method using pointing to inform an AUV of an object of interest to a diver. Following which, Chapter 4 greatly enhances the ability to notify an AUV of an object of interest through the inclusion of distance when given a pointing gesture. Chapter 5 improves the flexibility of pointing gesture-based communication as a diver is able to provide explicit instruction demanded by the situation or object located for downstream AUV tasks. Chapter 6 presents our investigation into task-based underwater image enhancement. A brief description of robotic platforms used to complete this research and our contributions to the design and building of them are presented in Chapter 7 when notifying an AUV. Chapter 8 summarizes our key findings and outlines potential future directions for this research.

Chapter 2

Background and Related Work

Communication between divers and AUVs requires natural communication that can be easily utilized by the diver and provide as much information to the AUV as possible. In this chapter, we provide background and previous work relating to our topic of computational visual communication. As the use of diver pointing for control with AUVs is a novel idea, we focus our attention on the areas of general underwater human-robot interaction (U-HRI), as well as the use of pointing gestures within the scope of robotics in general. We use RGB cameras as our main sensor to perform the U-HRI methods. Due to environmental challenges that occur with visual perception underwater, we also investigated the use of task-based underwater image enhancement in relation to U-HRI. We therefore introduce background information on underwater image enhancement as well.

2.1 Diver and AUV Interaction

In this section, we focus on two aspects of Diver AUV interaction, first through the *passive* interaction of diver detection and tracking and second, *active* communication and interface with AUVs.

Visual detection and following of human divers by autonomous robots is well-studied for its usefulness in human-robot cooperative underwater missions [13, 14]. Due to operational simplicity and computational efficiency, the simple feature-based detectors followed by standard model-free trackers have been traditionally used in autonomous

diver-following applications [15, 16]. Particle filters and optical flow-based methods are also utilized for tracking divers in spatio-temporal domain [16]. Since color distortions and low visibility issues are common underwater, the frequency-domain cues [17, 18] of divers' motion are used for reliable detection as well. Also, several feature-based learning approaches [16, 19] such as support vector machines (SVMs) and ensemble methods have been investigated for diver tracking and underwater object tracking. However, these methods lack generalizability and often fail in noisy visual conditions. Enhancing underwater images for diver detection may help to improve current detection methods

Active communication with AUVs requires specialized communication methods [20] often including the use of robotic vision or audio to receive directives from divers through the use of tools such as fiducial markers [21] or tablets [22], or special dive gloves [23]. Using gestures to interface with AUVs, while relatively under-explored in comparison with aerial and terrestrial vehicles, is beneficial for divers as it requires no extra equipment. Currently, deployed methods ([24, 25, 14, 26]) have been designed with ease of use of divers in mind, including some with directional signals (*e.g.*, [26]). Expanding this interface to convey locational cues can be seen as the next step. While Walker [27] provides an analysis of the ability to recognize pointing gestures underwater on state-of-the-art deep-learned object detectors (*e.g.*, Single Shot Multibox Detector (SSD) [28], Faster R-CNN [29]), there is no current literature on directing an AUV to a location or object of interest through pointing gestures. We contribute the first look at using pointing to provide specific locational cues of a diver's area of interest to an AUV and exploit these cues for AUV-assisted task completion in that location.

2.2 Underwater Applications of Human Pose Estimation

Human pose estimators are often used to provide landmarks for pointing gestures ([30, 31, 32, 33]). The pose estimator locates landmarks, typically joints, on a human body. While human pose estimation in both 2D and 3D is widely studied (see [34, 35, 36]), there does not exist a vision-based human pose estimator dedicated for underwater use. Chavez *et al.* [37] tracks a diver using point clouds of diver pose; however, specific pose landmarks are not found. Terrestrial-based, out-of-the-box monocular 2D pose estimators have been applied with some success; *e.g.*, Islam *et al.* [38] use estimated

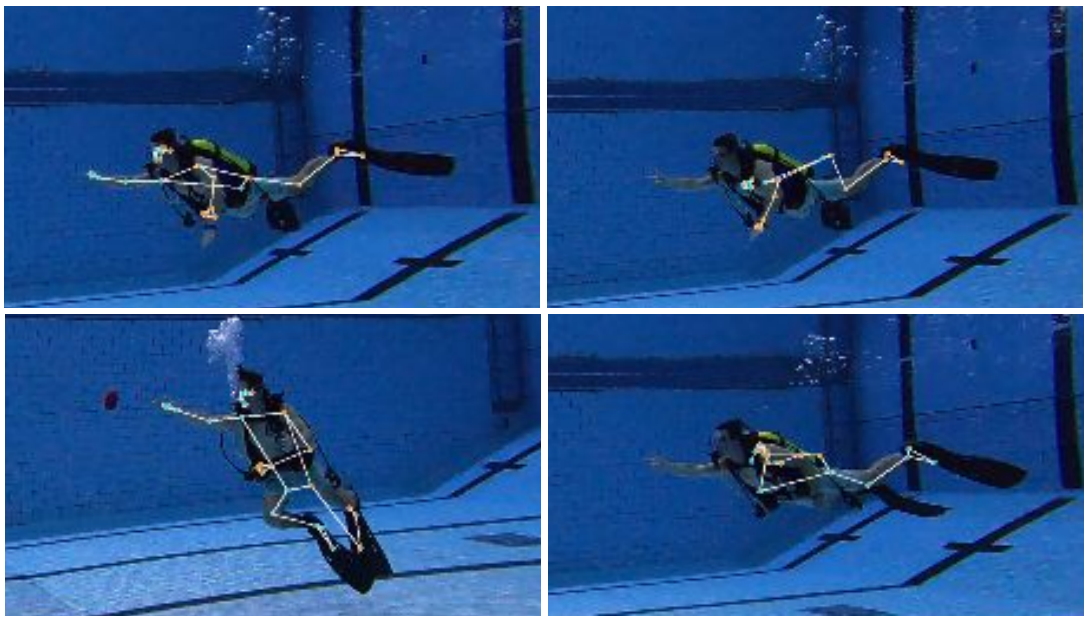


Figure 2.1: Example of some failure cases in underwater human pose estimation. Even in closed-water conditions, a combination of environmental factors, non-standard body pose, and SCUBA attire can affect robust 'terrestrial' vision algorithms. Image quality is due to natural underwater artifacts.

human body pose (using OpenPose [39]) to find relative positions of two robots, and Fulton *et al.* [40] create an autonomous diver approach method using TRT pose [41]. The success of human pose estimators is severely impacted not only by underwater vision degradation, but also by the positions of the human body and additional gear worn by scuba divers. Fig. 2.1 shows some of the common yet challenging results of pose estimation on a diver in a closed water environment using Mediapipe [42] Pose [43].

2.3 Pointing for Robotic Direction

In terrestrial and aerial robotics, the use of pointing gestures to relay information has been widely studied. For example, terrestrial and aerial vehicles can be given directions to park in a certain location determined by a pointing pose estimate [30, 31, 32]. For these applications, the user can point at a location on the ground plane and direct

the robot there. This is an impossibility in many underwater robotics applications as touching the ground plane, if it exists in view, goes against the task that is being accomplished (*e.g.*, coral reef inspection, or invasive species detection). Aerial and terrestrial robotics have also seen applications of finding an object of interest through pointing gestures. Pick-and-place tasks with robotic arms have seen success through hand gesture pointing [44, 45]. Großmann *et al.* [46] use a pointing gesture to indicate an object on a shelf that should be of interest to a mobile manipulator. Medeiros *et al.* [33] can direct a drone towards an object based on the intersection of a pointing direction of the user's arm and an object's bounding box. Delmerico *et al.* [47] provides a survey of many more uses for pointing gestures, with an emphasis on the uses with rescue robotics.

In aerial and terrestrial based robotics, many gesture based systems utilize light-wave based Time-of-Flight RGB-D cameras ([46, 44, 30, 48, 32, 49]) to gain access to depth information. The reliability of these sensors is significantly impacted underwater by environmental factors. Indeed, underwater depth estimation overall is challenging, as water causes significant distortions of incident visual light. The combined effects of optical attenuation and low-light conditions negatively impact traditional depth estimation methods. Stereo vision, our choice of sensor for depth estimation in [9], while challenging in the underwater domain (see Fig. 2.2 for an example dense disparity map from an underwater diver scene), has been used in a variety of functions such as detection of fish length [50]. The Caddian dataset [51] also includes stereo images along with data from IMU sensors attached to the diver for use in diver pose and tracking. In our work, we mitigate some challenges of dense stereo reconstruction by obtaining distance information for only points relevant to the pointing task through the use of *sparse* stereo triangulation. However, no previous work exists for the use of pose-based or locational pointing gestures for AUV control or communicating user interest in the underwater environment.

2.4 Enhancing Underwater Imagery

To address the inherent difficulties of underwater vision, several methods of image enhancement have been proposed. These methods can be used onboard the AUV before

(a) Left camera image. (b) Right camera image. (c) Disparity map of images.

Figure 2.2: Best if viewed at 175% zoom. Demonstration of the challenges of traditional dense stereo reconstruction underwater using the Semi-Global Block Matching algorithm [1]. Notice the disparity map contains inconsistencies within the diver's silhouette, as shown by the red arrow.

passing image information to some vision-based task, such as communication. The underwater image enhancement problem deals with correcting non-linear image distortions caused by the particularities of light propagation underwater [52, 53]. Some of these aspects can be modeled and well estimated by physics-based solutions by exploiting prior knowledge (*e.g.*, haze-lines and dark channel prior [52, 54]) or making statistical assumptions (*e.g.*, adopting an atmospheric dehazing model [55, 56]). However, these approaches require scene depth and water-quality measures for accurate modeling, which are not always available in practice.

As a practical alternative, learning-based approaches have been widely explored and they have demonstrated inspiring success in recent years. Several models based on deep convolutional neural networks (CNNs) and generative adversarial networks (GANs) report state-of-the-art (SOTA) performance [57, 4, 58] on benchmark datasets. Driven by large-scale supervised training, these approaches learn sequences of non-linear filters to approximate the underlying pixel-to-pixel mapping [59] between the *distorted* and *target* image domains. The contemporary CNN-based generative and residual networks (*e.g.*, Deep SESR [57], WaterNet [58]) are shown to be very effective in learning such mapping. Moreover, the GAN-based models (*e.g.*, FUnIE-GAN [4], UGAN [60], FusionGAN [61]) attempt to improve generalization performance by employing a two-player min-max game, where an adversarial *discriminator* evaluates the *generator*-enhanced images compared to ground truth samples. This forces the generator to learn realistic

enhancement while evolving with the discriminator toward equilibrium. While the existing models provide good generic solutions for perceptual enhancement, learning to recover image distortions for specific tasks such as diver detection or classification has not been explored in-depth in the literature.

While not extensively explored for underwater applications, integrating object detection components within the GAN architecture has been well-studied in the terrestrial domain. For instance, the auxiliary classifier GANs [62] are proposed to synthetically generate realistic images, while several variants of the Multi-task GAN (MT-GAN) [63] and Perceptual GAN [64] are used to improve small-object detection performance by using image super-resolution techniques. In such models, the components of bounding box regression and classification are determined through discriminator branches using additional fully connected layers. Liu *et al.* [65] integrate a RetinaNet [66] detector in DetectorGAN; however, this is done for generating data to train an object detector rather than improving its performance through image enhancement.

Chapter 3

Diver Interest via Pointing

The use of autonomous underwater vehicles (AUVs) to perform tasks underwater has become increasingly relevant in recent years. From inspection and maintenance of underwater infrastructure [5], biological monitoring [6], and archaeology [7], the utility of these AUVs heavily relies on working and cooperating with human divers. For each of the above-mentioned scenarios, it is highly likely that a robot will be directed to specifically navigate along a particular direction or inspect an object; this is referred to as the site acquisition and scene re-inspection (SASR) task. Such tasks could include inspecting a specified region of a pipeline, taking pictures of particular coral for a conservation biologist, or recovering an artifact. This work [67] presents a novel method that uses a natural human gesture to provide object location information to an AUV.

While Remotely Operated Vehicles (ROVs) are able to receive instruction through a tether, AUVs require different communication methods [20]. These methods include using robotic vision or audio to receive directives from divers through the use of tools such as fiducial markers [68], or special dive gloves [51, 23] as well as with gesture-based languages of hand signals with ([14, 26]) and without ([24, 25]) gloves. As many divers working alongside AUVs have their own areas of expertise and may not be robotics experts, a communication vector that is easily understood and capable of being performed naturally is essential. For this reason, we introduce a system for divers that uses pose-based pointing gestures which do not require additional equipment to instigate a response from an AUV towards an object or location of interest (Fig. 3.1).

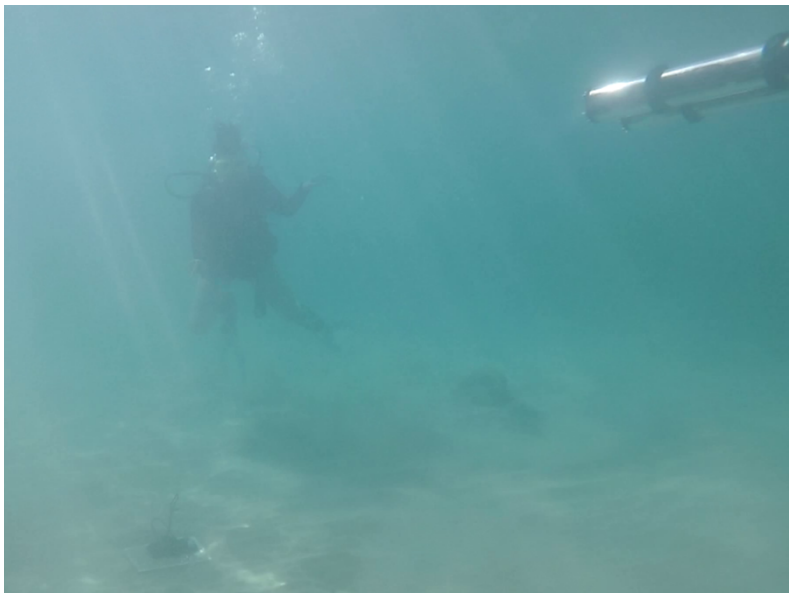


Figure 3.1: The LoCO AUV running DIP to navigate towards an object we point at on the seabed in response to diver pointing gesture. Image taken during field test in the Caribbean Sea.

Pointing is used as a natural form of human communication to share location and interest. Expanding this communication vector to robotics is not itself an unexplored area of interest (see Sec. 2). In the subtopic of marine robotics, however, the use and even exploration of this topic is almost non-existent (other than very few exceptions, *e.g.*, [27] [26]). Different methods of indicating interest are often used by scuba divers, *e.g.*, shining lights, carrying physical objects such as sticks or poles, and dropping strobes or beacons. However, these methods come with significant limitations. The underwater domain is known for its high degree of sensory signal attenuation and dispersion, and thus methods such as shining lights on objects of interest may not be precise enough for many needs. Without necessitating extra tools, pointing gesture communication relies on some form of visual perception. Underwater vision is challenging due to natural environmental factors (*e.g.*, scattering, absorption, and refraction of light among others). Image enhancement techniques (*e.g.*, [57, 69]) have been shown to mitigate some of these factors and improve the ability to use RGB cameras. These factors still significantly impact the *reliability* of commonly used terrestrial sensors, such as light-wave

based Time-of-Flight RGB-D cameras, used in many pointing gesture-based systems ([46, 44, 30, 48, 32]). As vision itself is challenging even in the best of water conditions, adaptation of ideas and implementation of terrestrially designed algorithms can be difficult.

Also unique to the underwater environment is the high probability that the diver, robot, and even object of interest may have a free-floating ability. For example, during terrestrial and pick-and-place applications, it can generally be assumed that anything static on a ground or table environment will remain there until moved deliberately. Even with relation to drones, the human is potentially able to remain on the ground plane and calculations for direction can take advantage of this. In open water, however, it is highly unlikely that the diver will remain on the floor of the body of water; in fact, they may even be moving along with water currents. Our approach takes this into account by leveraging only the pose of the diver and robot view space to find the area of interest.

We introduce **DIP** (*Diver Interest via Pointing*), a method to inform an AUV of the location of an unknown object via a pointing gesture. We show that the use of a human pose estimator in conjunction with a single camera is sufficient to create an area of interest from which an AUV can detect an object of interest. In addition to locating the area of interest as indicated by a diver in still images, we provide a working example of how our method can be integrated into an object inspection system, utilizing the LoCO AUV [12].

3.1 Methodology

3.1.1 Diver Interest via Pointing

The DIP approach is composed of a human pose estimator, a method for the AUV to predict a diver's area of interest, and an object detection algorithm applied to the predicted area of interest (Fig. 3.2). In the following section we describe the method to detect a diver's pointing pose, and using that, detail the creation of the area of interest. In addition, we enumerate the design choices that allow the implementation of the proposed algorithm on-board a physical robot.

For the purposes of this section, we make the following assumptions:

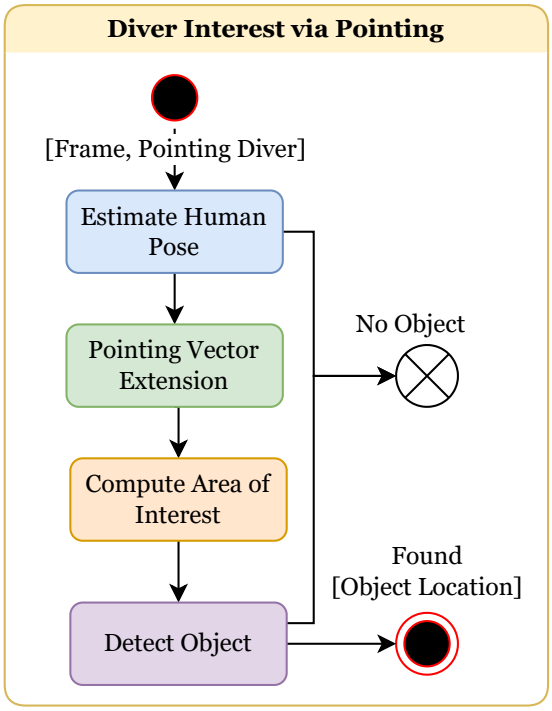


Figure 3.2: Schematic diagram of the DIP algorithm. Human pose estimation of the elbow and wrist joints are used to extend a vector in the direction the diver is pointing. Then, a triangular area of interest is created around the pointing vector. Finally, an object can be located within the region of interest. Colors denoting each step are consistent with those in Figs. 3.7 and 3.8.

- The diver is situated in an upright, pointing pose.
- The diver is pointing with their right arm.

3.1.2 Diver Pose Estimation

Vision-based AUV communication using pointing gestures from divers requires robust estimation of human pose. Human pose estimation underwater is challenging for a variety of reasons, including naturally degraded vision, non-standard body positions, and the wearing of SCUBA gear by divers (see Fig. 2.1). To minimize the need for correctly positioned body landmarks, we use only two, the elbow and wrist. As the

connection between these two points must be linear (*i.e.*, anatomically, there are no joints between them in human physiology), this creates the ability to generalize potential locations of interest in a rather straightforward manner. By simply extending the line segment connecting the elbow to the wrist, the general pointing direction can be found. The shoulder is excluded as unless the diver's body is turned towards the camera, landmark detection can be difficult (Fig. 2.1). We explicitly avoid the use of the hands or fingers for finding the pointing direction, as the poor visibility underwater, particularly at the usual interaction distance between an AUV and a diver, can make it difficult for pose estimators designed for terrestrial use to identify those body parts. For the purpose of our project, we use the Mediapipe Pose [42], estimator, although any pose estimator that can provide wrist $w_{(x,y)}$ and elbow $e_{(x,y)}$ landmarks in the two-dimensional image space can be used. Once the pose with the wrist and elbow landmarks have been located, we are able to determine a direction and local area of interest.

The use of 2D pose for determining pointing direction for SASR type tasks is sufficient, as objects of interest will often be close to the diver in distance, thus we can approximate the 3D directional vector in 2D image space. As marine environments are also generally uncluttered and without many structures, the need for depth information about the scene is not necessary for many tasks. Particularly when coupled with a robust object detector, 2D pointing vectors will often suffice for such purposes. Future work looks at expanding DIP to include 3D information for use in specialized situations.

3.1.3 Determining the Area of Interest

The area of interest to the robot is defined to be a triangular region extending from the diver's wrist, shown in Fig. 3.3 as the white triangle. A triangular shape is chosen, as the farther away a diver is from an object, the less accurate the actual gesture may be (*i.e.*, the diver's own gesture pointing at the object of interest will be less accurate from farther away). This error can potentially be magnified in relation to above-water scenarios, as the object, diver, and robot itself may not be able to remain stationary for a variety of reasons, such as being suspended in the water, moving with a current, or constantly maneuvering to hold position. Choosing a triangular region to search makes it possible to account for such inherent inaccuracies in pointing and have a more accurate detection of the object of interest.

First, the line segment connecting the elbow and wrist landmarks (as defined by the human pose estimator) is extended $ext_{(x,y)}$ by a scaling factor sf which can be modified as needed (Eq. 3.1):

$$\begin{aligned} ext_{(x)} &= w_{(x)} + sf * (w_{(x)} - e_{(x)}) \\ ext_{(y)} &= w_{(y)} + sf * (w_{(y)} - e_{(y)}), \end{aligned} \tag{3.1}$$

A scale factor of 10 has empirically been found to be sufficient in our investigations.

Once the segment is extended, the resulting point $ext_{(x,y)}$ is defined as the end of the pointing vector. To create an area of interest in the shape of a triangle in image space, two vertices are defined by adding and subtracting a *vertical constant* c, to the y value of the point extension (*e.g.*, $ext_{(x,y\pm c)}$) (see Fig. 3.3 for a visualization).

The third vertex of the triangle is defined by the wrist landmark of the human pose detector. This vertex may also be offset by a small amount ϵ_p to reduce false positives of hand detection to compensate for detector errors. With the use of a task-specific object detector, the offset may be omitted. The resulting vertices of the triangle are therefore defined as: $(w_{(x-\epsilon_p,y+\epsilon_p)}), (ext_{(x,y\pm c)})$.

Once the area of interest is computed, the search for the object of interest via object detection is confined to this area.

3.1.4 Detecting the Object of Interest

As the main goal is to identify the diver's intent to highlight an area of interest, the consequent object detection step is highly dependent on the actual task given to the AUV and thus should be changed accordingly, *i.e.*, a trained trash detector should be used if the AUV's task is to locate trash in an area of interest. However, even without knowing the object type ahead of time, it is possible to exploit low-level image features to identify objects pointed at by divers within the triangular area of interest. For example, point features (*e.g.*, SIFT [70], ORB [71]) and edge detectors (*e.g.*, Canny [72]) can be used to identify regions with a high probability of being objects of interest. The motivation to use such low-level feature extractors stems from the fact that the underwater environment often has little background variation, and only salient features will be on or in close proximity to the object. Due to the challenges of detection in an underwater environment, we demonstrate detection through two methods: keypoint

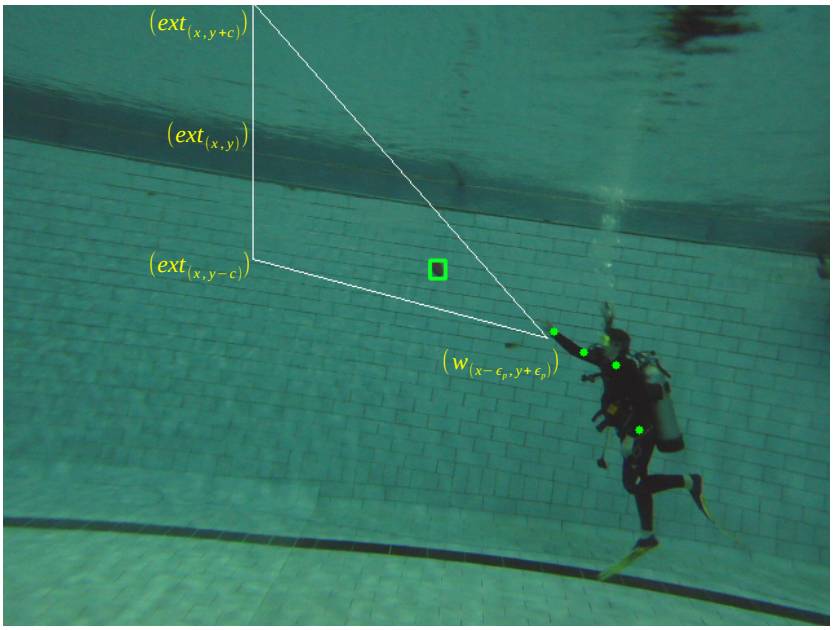


Figure 3.3: Visualization of the DIP algorithm: Human pose landmarks are located, an area of interest is created based on pose pointing vector, and an object is located within the area of interest.

and contour detection. Via keypoint detection, we choose the *keypoint with the greatest strength* to represent the object, even if this has the potential for occasional false positive detections. The Canny edge detector is used to extract object contours. If the centroid of the contoured region is within the triangle of interest, the target is chosen as the object of interest. Fig. 3.3 shows an example of DIP successfully finding the object pointed to by the diver.

3.2 Evaluation

Obtaining ground truth is difficult in underwater environments as there is constant motion between the diver, robot, and objects. In addition, the accuracy of DIP as a whole rather than the accuracy of the pose and object point location is a necessity; therefore, for the following evaluations, visual cues rather than straightforward landmark matching will be used in the assessment.

Figure 3.4: Two of the eight images the human correlation study participants were asked to annotate. The locations of the target objects are blackened out to not bias the participants' assessment of the pointing direction.

All evaluations are performed using the same parameters with input image sizes of 640×480 pixels. Empirically, we obtain a vertical constant of 100. In other words, the height of the triangular area of interest will be 200 pixels, just under half the image height. We also offset the wrist vertex by 5 pixels (*i.e.*, $\epsilon_p = 5$, see Sec. 3.1.3) to reduce false positives of detecting the hand. The vertices of the triangular area of interest therefore become $(w_{(x-5,y+5)}), (ext_{(x,y\pm100)})$.

3.2.1 DIP and Human-based Pointing Correlation

We present a small human study in which participants annotate a pointing vector on 8 images. Each image consists of a submerged diver pointing towards an object, either suspended in the water or resting on the bottom. The general location of an object of interest was removed (Fig. 3.4) to prevent preconceptions of what the diver is pointing towards. Nine participants annotated a line segment of the assumed pointing vector, beginning at the diver's wrist. Fig. 3.5 shows a compilation of the results: the green line segments represent the human-annotated vectors, the pink segment represents DIP's foundational pointing vector, and the white triangle represents the location of diver interest as produced by DIP. As can be seen from the quantitative data in Table 3.1, where image numbers match the vector images, fitting a 2D vector to a point can be highly variable even by human annotators. Angles are not included for DIP where pose estimation is incorrect.

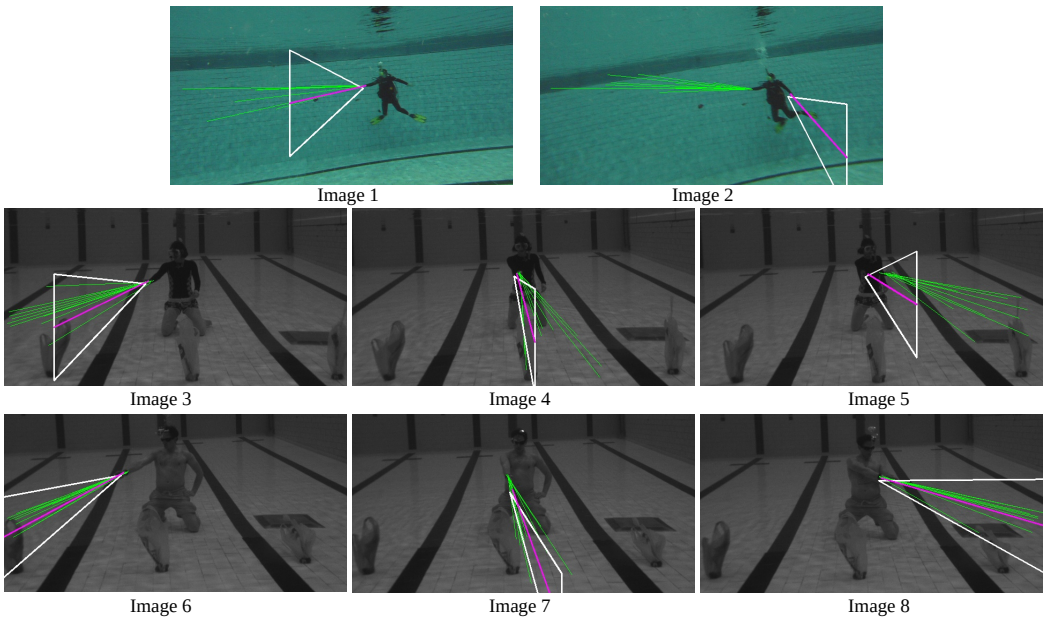
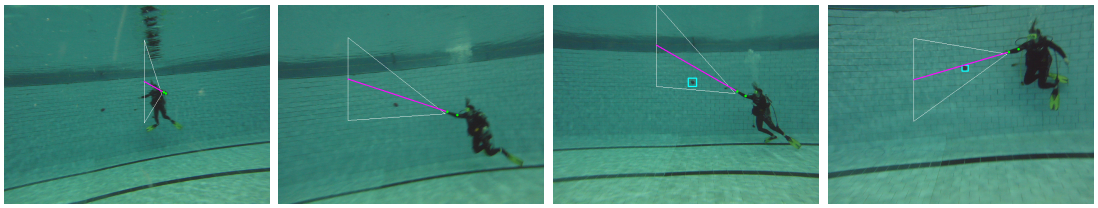


Figure 3.5: Results of human-annotated vectors (green) shown with the pose estimator-based resultant vector (pink), and DIP area of interest (white triangle). DIP's output predominantly aligns with human assessment of pointing directions.

	Mean (Human)	DIP	Difference	Variance (Human)
Image 1	3.059 rad	2.912 rad	0.147 rad	0.008
Image 2	3.198 rad	—	—	0.003
Image 3	2.849 rad	2.677 rad	0.121 rad	0.017
Image 4	1.097 rad	1.309 rad	0.211 rad	0.069
Image 5	0.357 rad	—	—	0.024
Image 6	2.717 rad	2.643 rad	0.074 rad	0.004
Image 7	1.249 rad	—	—	0.028
Image 8	0.286 rad	0.288 rad	0.002 rad	0.008

Table 3.1: Results of human study (Fig. 3.5): Mean human-annotated angle measure, DIP angle measure (if correct pose), Difference in Mean and DIP angles, and variance in human-annotated responses are recorded. All angles are in radians.



(a) (b) (c) (d)

Figure 3.6: Samples from the pose and object detection evaluation image set. All images are sourced from the LoCO AUV and may not be considered good quality to human viewers. In image (a), the pose detection fails and an incorrect region is created. (b) shows correct pose detection and area of interest creation, but an object is not detected. In (c) and (d) objects are detected through the SIFT algorithm and Canny edge detection respectively.

Fig. 3.5 also conveys some of the challenging situations that can occur within a highly specified underwater setting. As our method is dependent on the human pose, if the pose captured is incorrectly such as in Image 2, the area of interest will be so as well. Failure cases were found to occur frequently when the diver's pointing arm intersects with the torso, as seen in diver center and left-pointing evaluation cases (See Images 5, 7 in Fig. 3.5).

3.2.2 DIP-based Pose and Object Detection

We evaluate DIP for the ability to use human pose estimation and low-level feature detectors to first compute a diver's specified region of interest and then locate an unknown object within that region. Due to the challenging environment, we consider the ability to identify an accurate region of interest more beneficial than exact landmark matching. Regarding unknown object detection, we provide results for both the SIFT [70] algorithm and Canny [72] edge detection as defined in Sec. 3.1.4.

A selection of 650 frames were taken from a ROS [73] bag file recorded with the LoCO AUV [12] in a closed-water environment (Fig. 3.6). All images in sequence are included unless the entirety of the diver's body is not in the scene. Fig. 3.7 shows the results of a visual examination of this dataset. While not all images may be considered "good quality" by human standards, we include these images to demonstrate the effectiveness

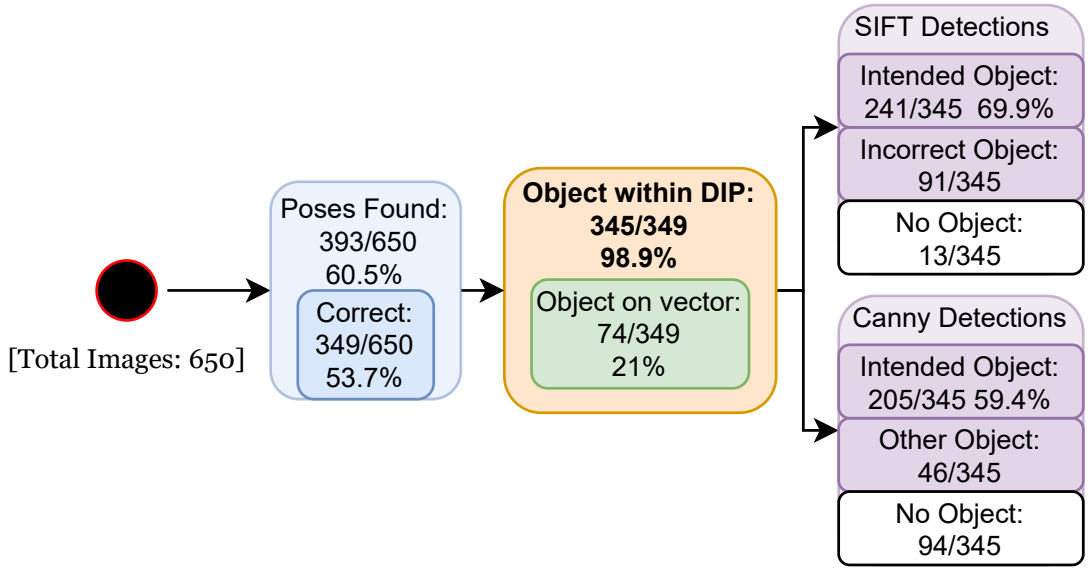


Figure 3.7: Evaluation of DIP pose estimation and object detection. The ratios presented depend on the success of the previous steps. Colors denote the step evaluated as in Fig. 3.2.

of DIP from the AUV viewpoint. The evaluation dataset can be considered to represent fairly optimal conditions in terms of water quality and distance from the diver.

Taking into account only those images that produce a "correct" pose (349/650 images), we see that the object of interest is included in the region of interest 345 out of 349 times or 98.85%. On the other hand, the pointing vector itself lands on the object only 74 times, or 21.2%. As supported by the results of Sec. 3.2.1, due to variations in pointing, locating an area of interest produces better detection results than choosing a single vector. As to the evaluation of object detection, the inclusion of a specified object detector is essential for each task. However, due to the nature of the underwater environment, objects of interest are located far better than random chance with low-level features, with the SIFT algorithm failing to locate a potential object of interest in only 13 out of 345 images.

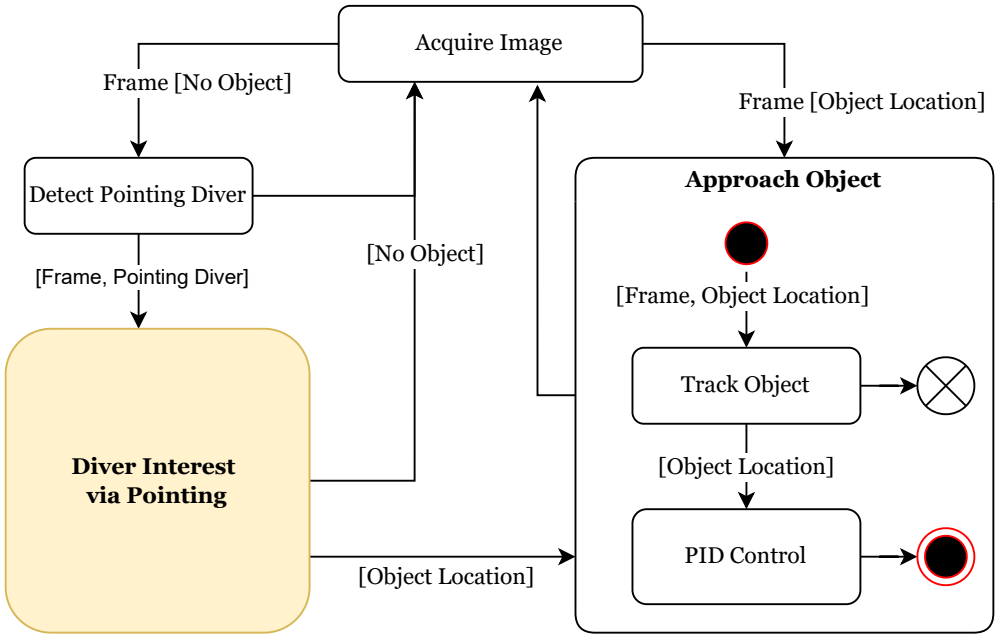


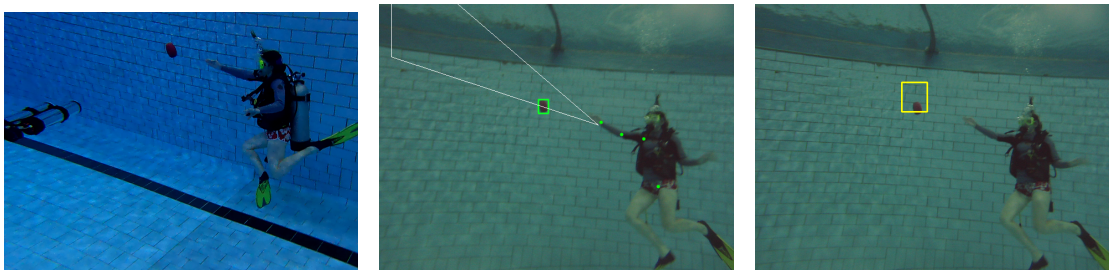
Figure 3.8: Diagram showing the full system of Unknown Object Investigation guided by the DIP algorithm, as implemented on-board the LoCO AUV.

3.2.3 Runtime

With Mediapipe Pose as a backbone, the DIP algorithm (pose estimation and SASR creation) runs at 4.714 fps on an Intel^{TM} i7-6700 CPU, which is acceptable for AUV operations. The object detection portion of algorithm runtime will be dependent on the use case and chosen object detector.

3.3 Experimental Implementation onboard the LoCO AUV

DIP works as the guiding force for an unknown object investigation task (Fig. 3.8). We deploy DIP, along with the rest of the system, entirely onboard the LoCO AUV, use a monocular RGB camera, and perform computations on an NVIDIA Jetson TX2 embedded system. In an enclosed pool environment, LoCO was required to detect that a diver is pointing, employ DIP to determine where an object may be located and locate an object, and then finally actuate towards the object for closer inspection.



(a) An object of interest is indicated to the LoCO AUV.

(b) An object is identified through the DIP algorithm.

(c) The object is tracked in the following frame.

Figure 3.9: Demonstration of the task: Unknown Object Investigation. Images (b) and (c) are from the LoCO AUV point of view.

Within this system, a diver is determined to be pointing through a pre-trained SSD [28] detector with a VGG-16 [74] backbone [27]. Once the pointing diver is detected, DIP proceeds to check incoming frames until the pose is detected. In the same frame the pose is detected, object detection is attempted. This cycle proceeds on successive frames until an object is located. The diver's interest and the object within the location are then considered confirmed, and the algorithm moves to initiate robot locomotion. The AUV begins moving toward the object through a Proportional-Integral-Derivative (PID) approach controller defined through a bounding box and image size ratio. We use the CAMShift tracker [75] to return a new location for the PID controller to continue AUV movement toward the object, as it might be subjected to unintended motion underwater. Fig. 3.9 shows snapshots from LoCO's viewpoint during a successful trial. Fig. 3.8 diagrams the system as a whole as it runs onboard the LoCO AUV.

Out of five trial runs, the area of interest is defined (*e.g.*, the human pose is correct) four times. An object was located during each trial, and LoCO proceeded to move autonomously towards the perceived object, making task execution successful. Three out of four times, however, markings on the pool wall were located inside the area of interest. These markings were considered to be the object and LoCO moved in that direction. Using a detector trained for specific objects of interest would help mitigate this issue.

3.4 Conclusion

We present a novel communication algorithm, Diver Interest via Pointing (DIP), to signal directional intent to an AUV. By detecting the natural communication vector of pointing, we show that an AUV can infer a diver's area of interest. While working in the challenging underwater environment, we have shown that a triangular area of interest can be found with the use of a monocular camera and an out-of-the-box human pose estimator. Knowing that the area of interest exists, an AUV can perform tasks specifically within that area as determined by a diver. One such task is the inspection of an unknown object. We validate DIP through an integrated system and demonstrate its feasibility onboard the LoCO AUV. This chapter provided a two-dimensional method that allows a diver to inform an AUV of the location of an object. One challenge of this method is the necessity that a single object is located within the area of interest. Our next chapter discusses our work to continue to improve directional gesture communication through the inclusion of 3D scene geometry.

Chapter 4

Diver Interest via Pointing in Three Dimensions

While the lack of crowded objects and uncluttered background of many underwater scenes lends to the ability to perform two-dimensional object location described in Chapter 3, the inclusion of distance in human-robot communication through pointing improves the ability to detect specific objects located in the direction of pointing when multiple objects are in the scene. Underwater distance estimation overall is a challenging problem as significant distortions of incident visual light and optical attenuation negatively impact traditional distance estimation methods (see Fig. 2.2 for an example dense disparity map from an underwater diver scene) and sensors such as light-wave based Time-of-Flight RGB-D cameras become significantly impacted. These limits mean that computer vision techniques used terrestrially must be adapted to provide useful distance information underwater. In our work, we mitigate some challenges of dense stereo reconstruction by introducing the use of a human pose estimator to obtain body joints to use them as relevant corresponding points in a stereo pair (*e.g.*, in [38]). Distance from the diver to the AUV can then be found through the use of sparse stereo triangulation (See Section 4.1.2).

In this work, we significantly improve two-dimensional Diver Interest via Pointing [8] by combining pose-based gesture recognition with underwater stereo distance information to create the Diver Interest via Pointing in Three Dimensions (DIP-3D) algorithm.

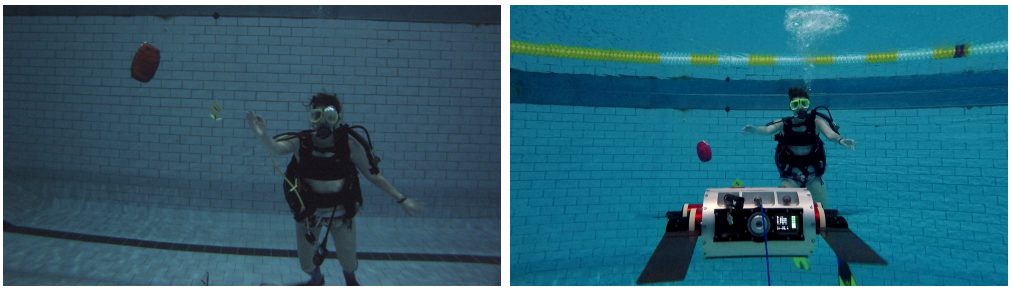


Figure 4.1: Experimental scene of a closed-water test from two view points. Two potential objects of interest are in the scene and a diver points forward to the red cylindrical object. Through the use of DIP-3D, the Aqua AUV is able to locate the correct object.

DIP-3D gives AUVs the ability to determine the position of an object pointed to by a diver in three dimensions (See Fig. 4.1). The contribution of this work is a modular visual framework for AUVs to determine the position of an object of interest pointed at by a human diver when there are multiple objects in the scene. We leverage 2D human body pose and object detection as well as sparse stereo reconstruction to provide scene distance information of an AUV to achieve this goal. We also provide evaluations in the form of a human study and closed-water experiments.

4.1 Methodology

To find objects of interest pointed to by a diver in 3D, we first determine the diver pose and potential objects of interest in two-dimensional image space (x, y). Next, we perform sparse stereo triangulation to extract the corresponding three-dimensional points, (x, y, z) so that we may extend a pointing vector and locate the nearest object. Finally, we recover the (x, y) coordinates of the object of interest so that a controller in image space can be used by the AUV to move towards the object. A schematic of the DIP-3D algorithm can be seen in Fig. 4.2. We make the following assumptions:

- The stereo camera has been calibrated.
- The diver is situated in a pointing pose.

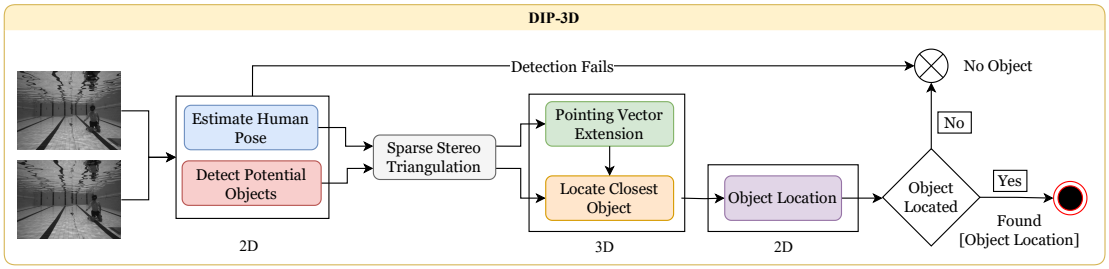


Figure 4.2: Schematic diagram of the DIP-3D algorithm. Utilizing both the left and right images from the robot's stereo camera, we estimate 2D locations of both the human pose keypoints and candidate objects of interest. We then reproject these keypoints to 3D, where we discern the object of interest. Finally, we recover the object of interest in the 2D image plane for our visual servo control scheme. Colors in the diagram are coordinated with colors in Section 4.2.

- The diver is pointing with their right arm, not pointing in front of or cross-body. The algorithm can be easily modified to use the left arm.
- The object of interest is present in both stereo pairs.

4.1.1 2D Diver Pose Estimation and Objects of Interest

Keypoint pose estimation of divers is an open problem both in 2D and 3D. We choose to begin in 2D, as there has been previous work with the use of 2D pose estimation and U-HRI. As pose estimation is challenging, we minimize the number of body keypoints needed for our algorithm so that incorrect detections of unessential body parts will not inhibit use. We use the wrist, elbow, and shoulder as our keypoints. While only the wrist and elbow keypoints are required for pointing vector extension, we include the shoulder to filter infeasible 3D body poses caused by the challenge of underwater distance estimation.

Let ***pose_2D*** define a set of 2D pose keypoints in image coordinates for an image of width W and height H which can be written for both the left and right stereo pairs as

$$\boldsymbol{pose\_2D}_{\text{left,right}} = \{(x_w, y_w), (x_e, y_e), (x_s, y_s)\},$$

where $x, y \in [0, \text{W}] \times [0, \text{H}]$, w, e, and s denote wrist, elbow, and shoulder, respectively.

Assuming a pose has been detected in both the left and right camera images, we proceed to identify potential objects of interest. Similar to [8], we mask a portion of the images to prevent detection of objects within infeasible regions, based on our assumptions. We mask the images from top to bottom, beginning at a constant distance from the right of the wrist to the left side of the image. We discard the region including the diver and anything to the left of the diver.

In principle, the object detection method depends on the actual mission given to the AUV and should be modified accordingly *e.g.*, trash, coral, or artifact detection. For this work, since there is generally little background variation in the scene, and only salient features remain after masking, we use the low-level feature extractor SIFT [70] for our detector. To find matching keypoints between the images, we use brute force k-nearest neighbors matching. After finding keypoint matches, we use Lowe's ratio test as described in [70] with an empirically determined ratio of 0.3 to discard features with poor matches. Let the candidate objects in the left and right images after filtering and masking be

$$\boldsymbol{obj\_2D}_{\text{left,right}} = \{(x_0, y_0), ..., (x_n, y_n)\},$$

where n denotes the number of candidate objects. We assemble the detected pose keypoints into a single set for each image as

$$\boldsymbol{k}_{\text{left}} = \{\boldsymbol{pose\_2D}_{\text{left}}, \boldsymbol{obj\_2D}_{\text{left}}\}$$
$$\boldsymbol{k}_{\text{right}} = \{\boldsymbol{pose\_2D}_{\text{right}}, \boldsymbol{obj\_2D}_{\text{right}}\}.$$

To discriminate between potential objects appearing at a different distance with respect to the AUV, we reproject these points to 3D space to identify the intended object of interest.

4.1.2 Sparse Stereo Triangulation

Typical stereo correspondence algorithms such as Block Matching and Semi-Global Block Matching (SGBM) [1] algorithms rely on a cost function for computing stereo correspondence. This cost function is used to find the best match location between stereo image pairs for each pixel location. Effectively, the minimum cost matching location

is chosen as the pixel match. While this works in the terrestrial domain, particularly in highly salient feature regions, computing this cost in the underwater domain leads to mismatches in correspondences. Fig. 2.2 demonstrates an example disparity map output utilizing an OpenCV [76] implementation of the SGBM algorithm. Notice the inconsistencies in illuminated pixels across the disparity map. This indicates that the algorithm finds many mismatches in corresponding pixels.

The accuracy of the pointing methodology in this work is highly coupled to precise distance estimation. To mitigate the challenges we see with dense stereo reconstruction, we first predict the diver pose and objects of interest in the camera left and right camera images. We then triangulate the camera frame's three-dimensional pose using these body pose and object keypoints alone and not the entire scene. This process works as follows.

Let $(\boldsymbol{x}_{\text{left}}, \boldsymbol{y}_{\text{left}})$ be the set of detected pose keypoints in the camera left image plane, and let $(\boldsymbol{x}_{\text{right}}, \boldsymbol{y}_{\text{right}})$ be the detected set of keypoints in the camera right image. Assuming the images have been rectified using calibrated camera parameters so their epipolar lines are parallel, then we compute the disparity D_i between the detected values in the horizontal coordinate $\boldsymbol{x}_{\text{left}}$ and $\boldsymbol{x}_{\text{right}}$ as

$$D_i = [\boldsymbol{x}_{\text{left}} - \boldsymbol{x}_{\text{right}}]_i = \frac{fB}{Z_i}, \tag{4.1}$$

where f is the left image dominate camera focal length, B is the baseline, and Z_i is the distance to the pose keypoint i in camera frame coordinates. We recover the xy-coordinates for pose keypoints and objects using the reprojection matrix [76] with respect to the left image plane coordinates.

$$\begin{bmatrix} \boldsymbol{X} \\ \boldsymbol{Y} \\ \boldsymbol{Z} \\ \boldsymbol{W} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & -f \\ 0 & 0 & -1/T_x & (c_x - c_x^{\text{right}})/T_x \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{\text{left}} \\ \boldsymbol{y}_{\text{left}} \\ \boldsymbol{D} \\ \boldsymbol{1} \end{bmatrix}. \tag{4.2}$$

Here, T_x is the translation component in the x-direction that translates from the left to the right image plane, c_x and c_y are the principal point components in the left image plane, and c_x^{right} is the x-component of the principal point in the right image plane. We recover the camera frame three-dimensional locations using Eq. 4.2 scaled with respect

to the homogeneous coordinates as $(x, y, z) = (\boldsymbol{X}/\boldsymbol{W}, \boldsymbol{Y}/\boldsymbol{W}, \boldsymbol{Z}/\boldsymbol{W})$.

After performing sparse stereo triangulation, we have the set of locations in 3D with respect to the left camera image. We define the three-dimensional pose locations as

$$\boldsymbol{pose\_3D} = \{(x_w, y_w, z_w), (x_e, y_e, z_e), (x_s, y_s, z_s\},$$

where $(x, y, z) \in \mathrm{R}^3$, and candidate object locations as

$$\boldsymbol{obj\_3D} = \{(x_0, y_0, z_0), ..., (x_n, y_n, z_n\},$$

where n is again the number of candidate objects found in 2D space. It is important to note that these 3D values are with respect to the camera frame and not a global or world frame. Using this information, we can filter the three-dimensional points to assist in limiting or removing incorrect information for the AUV to act upon. We limit the difference in z values between the wrist and elbow (mean = 0.254) and elbow and shoulder (mean = 0.377) as we empirically determine the value should never be larger than 0.5. We also filter for invalid disparity computations which result in $z = \pm\infty$.

4.1.3 Choosing the Correct Object

We are now left with reasonable 3D locations for the diver's elbow and wrist along with potential objects of interest based on the AUV's left image. Let $\boldsymbol{w} = (x_w, y_w, z_w)$ define the vector coordinates of the wrist keypoint, $\boldsymbol{e} = (x_e, y_e, z_e)$ define the elbow keypoint, and $\boldsymbol{o}_i = (x_i, y_i, z_i)$ define candidate object location, where $i \in [1, n]$.

We are now left with reasonable 3D locations for the diver's elbow and wrist, along with potential objects of interest based on the AUV's left image. Let $\boldsymbol{w} = (x_w, y_w, z_w)$ define the vector coordinates of the wrist keypoint, $\boldsymbol{e} = (x_e, y_e, z_e)$ define the elbow keypoint, and $\boldsymbol{o}_i = (x_i, y_i, z_i)$ define candidate object location, where $i \in [1, n]$. In order to determine the most probable object of interest, we first project a line $\boldsymbol{ext}$ in the direction extending from the elbow, through the wrist, and towards the direction of pointing as

$$\boldsymbol{ext} = \boldsymbol{w} + \mathrm{s_f} * (\boldsymbol{w} - \boldsymbol{e}), \tag{4.3}$$

where $\mathrm{s_f}$ is a scaling factor that can be modified as needed. We use an empirically determined scaling factor $\mathrm{s_f} = 3$, as 3 meters is the approximate distance between the

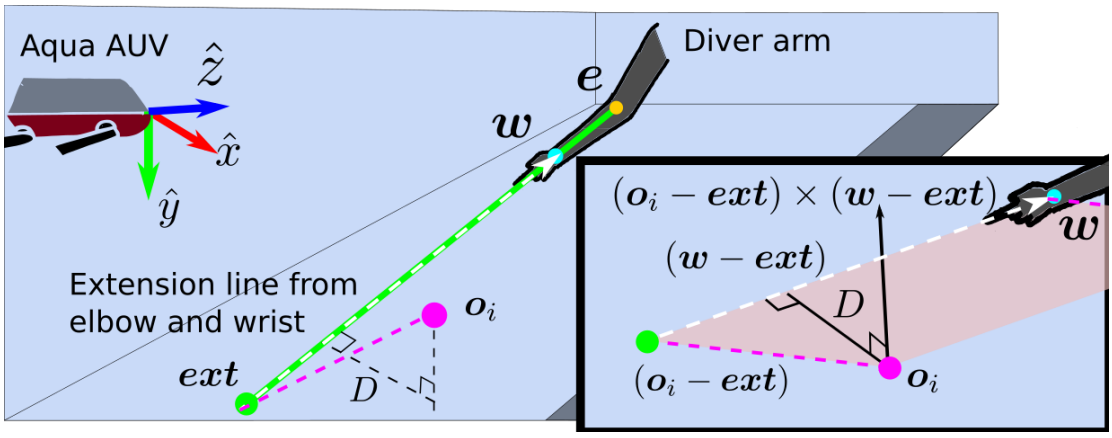


Figure 4.3: In the frame of reference of the robot's camera, we extend the line from the elbow to the wrist. Then we use 3D geometry to compute the perpendicular distance D between each candidate object and the extended pointing line.

camera frame origin and the wrist point. This ensures the extension line intersects the xy-plane, which ensures we can recover objects that appear in the foreground of the robot's image plane.

To find the object of interest, we find the object $o_i^* \in \boldsymbol{obj\_3D}$ that minimizes the perpendicular distance D_i between the object and the line that connects the extension and wrist. This process is demonstrated in Fig. 4.3. From 3D geometry, we know

$$D_i = \frac{\|(\boldsymbol{o_i} - \boldsymbol{ext}) \times (\boldsymbol{w} - \boldsymbol{ext})\|_2}{\|\boldsymbol{w} - \boldsymbol{ext}\|_2}. \tag{4.4}$$

We recover the 2D object location using the index of the minimum object distance, as AUV control methods often use 2D image coordinates.

4.2 Experimental Setup and Evaluations

The objective of DIP-3D is to allow an AUV to determine the direction and location of an object of interest to a diver in three dimensions. As there is no existing research to benchmark against, we measure the efficacy of our method through two different evaluations. A *human survey* provides both a reference for difficulty of determining a 3-dimensional location using two-dimensional images and comparing the results of our

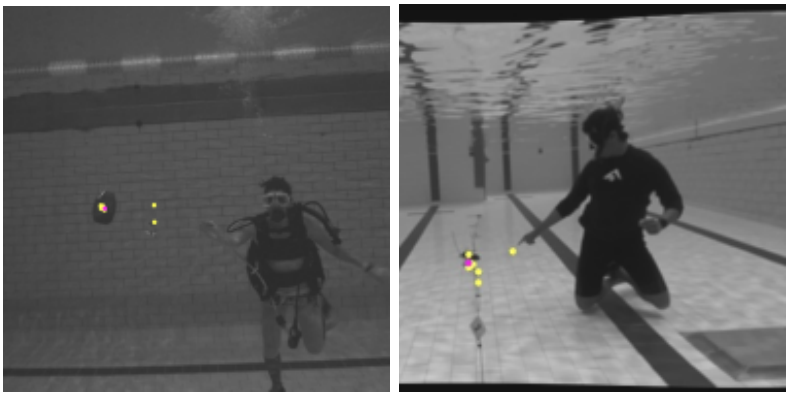


Figure 4.4: Selected images from the human survey. Projected annotations are shown in yellow, and DIP-3D results in purple. Images enhanced and cropped for readability.

method to what a human assesses to be the object of interest. Additionally, a *directional evaluation* validates our method to allow an AUV to locate an object in the direction a diver is pointing. We perform directional evaluations in both closed-water (pool) and open-water (sea) environments to show validity for real-world applications.

4.2.1 DIP-3D and Human-based Target Location Correlation

A human survey was performed to provide a reference on the difficulty of locating an object in 3D space given a 2D image. Ten images were chosen in which DIP-3D accurately identified an object in the direction the diver was pointing. Each image includes a pointing diver and the potential objects. Nine participants were asked to draw a four-sided bounding box on the RGB color image where they believed the diver was pointing. Post-processing allows us to project the centroid of the bounding box to the same frame of reference as the 2D-detected object found by our algorithm. The average pixel distance from the annotated centroid to our result is 45.9 ± 60.28 pixels. An example of the rectified image results is shown in Fig. 4.4, in which purple marks the predicted object and yellow marks the human annotations. As can be seen, the added dimension of distance perception assists greatly in locating the result of a pointing gesture.Table 4.1 provides a closer look at the results of this survey, including a breakdown by image and participant of the distance from the human annotation and DIP-3D resulting detection. Fig. 4.5 provides all visual results of the survey.

Table 4.1: A comparison of all results from the DIP-3D human study. Euclidean distance from the DIP-3D detected point to the human annotation is shown in pixels. Yellow boxes note the annotations that were farther from the detected point, blue denotes annotations near the detected point.

	P1[a]	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**	**P9**	**Error**[b]
Image 1	26.79	88.15	40.18	44.64	30.48	82.46	47.55	53.19	32.72	49.58± 20.73
Image 2	9.54	84.37	45.27	2.84	52.79	32.04	63.77	43.02	56.03	43.3± 24.19
Image 3	88.72	13.42	27.55	20.03	12.11	61.18	13.07	61.2	12.09	34.37± 26.94
Image 4	56.09	46.99	83.8	77.19	326.83	125.59	75.68	302.99	45.17	126.7± 103.29
Image 5	60.52	105.16	13.12	35.89	62.22	34.41	96.65	23.15	65.55	55.19± 29.82
Image 6	17.32	153.9	13.42	20.75	156.36	14.14	14.34	4.31	8.91	44.83± 59.12
Image 7	7.7	328.67	5.15	6.48	7.74	10.9	7.06	3.55	5.91	42.57± 101.17
Image 8	18.75	39.21	10.72	10.95	11.16	13.54	8.79	20.75	1.59	15.05± 10.02
Image 9	42.33	50.36	20.74	21.99	52.98	38.32	16.05	45.8	15.56	33.79± 14.29
Image 10	6.96	34.56	7.2	11.24	9.75	13.34	23.24	7.11	8.74	13.57± 8.84
Error[b]	33.47± 26.15	94.48± 87.21	26.71± 22.99	25.2± 21.21	72.24± 94.81	42.59± 35.4	36.62± 30.48	56.51± 84.53	25.23± 21.83	45.9± 60.28

[a] P refers to participant identification number. We utilize this shorthand for brevity.

[b] Errors are reported in *mean±standard deviation* format in pixel units.

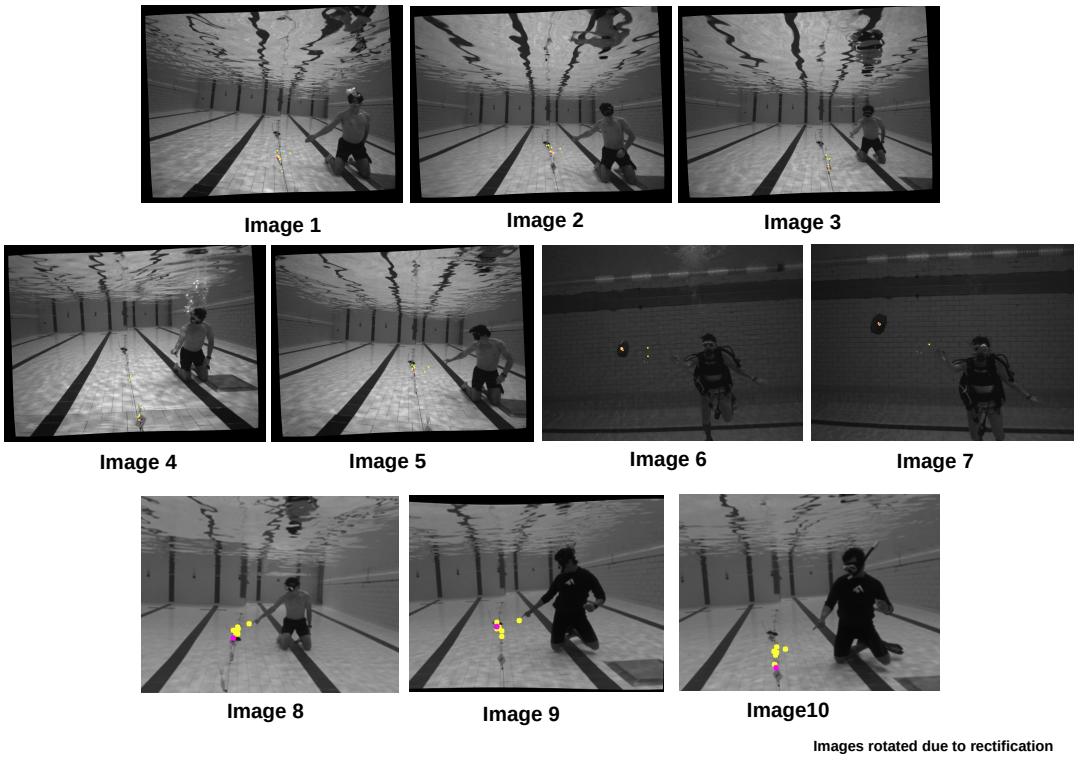


Figure 4.5: Images from the human survey. Projected annotations are shown in yellow, DIP-3D results in purple. Images are enhanced and cropped for readability.

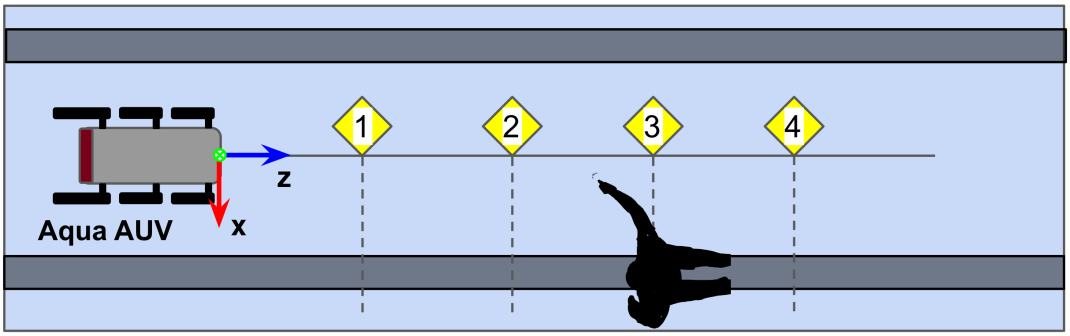


Figure 4.6: Top view of the on-robot experimental setup. A track line with meter markings was positioned along the center line between two lane markers. The human was positioned along the right lane line from the robot's perspective.

4.2.2 Directional Evaluation Setup

There are several challenges in evaluating the algorithm for accuracy of pointing direction: 1. Obtaining precise ground truth measurements underwater for object locations is challenging due to the constant relative motion of the robot, diver, and object, 2. Multiple objects must be located within the AUV's image space while the diver is pointing, with the approximate locations of these objects in relation to each other known as well, and 3. The direction the diver is pointing during the data collection phase must be known, as it can be difficult to determine the pointing locations *a posteriori* in the resulting images.

To minimize the above challenges, we utilized a partially instrumented setup, where a track line with numbered placards using fishing floats to assist with visibility was laid straight on the bottom (shown in Fig. 4.6 between two lane markers in a closed-water environment). The track line delineated meter distances from the robot to objects of interest, showing an approximate distance from the diver to the AUV's camera. Fig 4.7 provides a side-by-side view of the two directional evaluation setups and the differences of water conditions.

Proper calibration has drastic effects on the quality of 3D reconstruction [77, 78] as well, so we collected ample calibration data and ran *in situ* calibration of our stereo cameras. With this information, we now show that the AUV can discern the direction

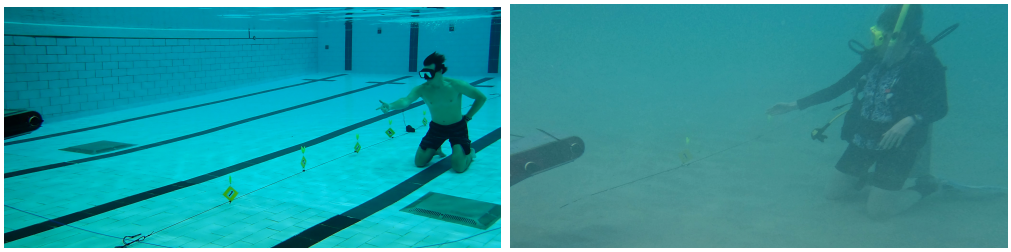


Figure 4.7: Experimental setup of the closed and open-water direction evaluations. DIP-3D is used to determine which yellow tag the diver is pointing towards.

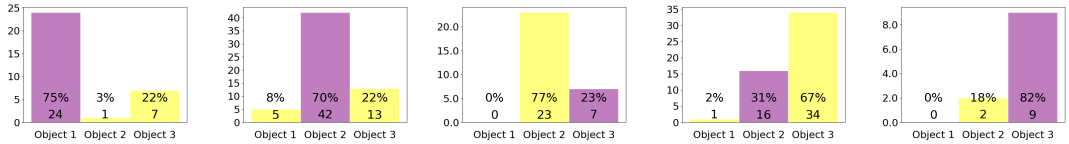
(near or far) of the object of interest from the AUV.

Images for both evaluations were recorded in a ROS [73] bagfile and extracted as image pairs with an image size of 1600 × 1200 pixels. Image pairs with the desired object, pointing diver, and at least one detected feature in the frame were retained. It was not necessary for a possible feature to exist on or near the desired target in the images. During the recording of the bag file, each pose and object location was held for approximately 6 seconds. The quality of pose estimation and feature detection in the stereo pair varies by the distance from the AUV, so the number of valid pairs of detected objects in each location varies.

The images were rectified, and each of the number placards 1, 2, and 3 were annotated with a 2D point. The evaluated distance from the AUV to the diver and object is limited by the detection results on the rectified images (*i.e.*, the pose of the diver was undetectable farther than 3 meters in the closed-water and 2 meters in the open-water evaluation from the AUV). Thus, distances further than 3 and 2 meters, respectively, while gathered and tested for validity, are not included in this evaluation.

4.2.3 DIP-3D Directional Closed-Water Results

Fig. 4.8 shows a breakdown of the count of times the location of the 'closest object' was nearest to the intended target. The closer the object and diver to the AUV, the more accurate the results. The inaccuracies in the farther distance are due to a combination of a lack of potential objects near the target as well as a lack of defined poses for the diver. This is especially clear when both the person and object are located about 3 meters from the AUV, where a total of only 11 images in which a pose and at least one object were



(a) Person Position 2, Pointing at Object 1

(b) Person Position 2, Pointing at Object 2

(c) Person Position 2, Pointing at Object 3

(d) Person Position 3, Pointing at Object 2

(e) Person Position 3, Pointing at Object 3

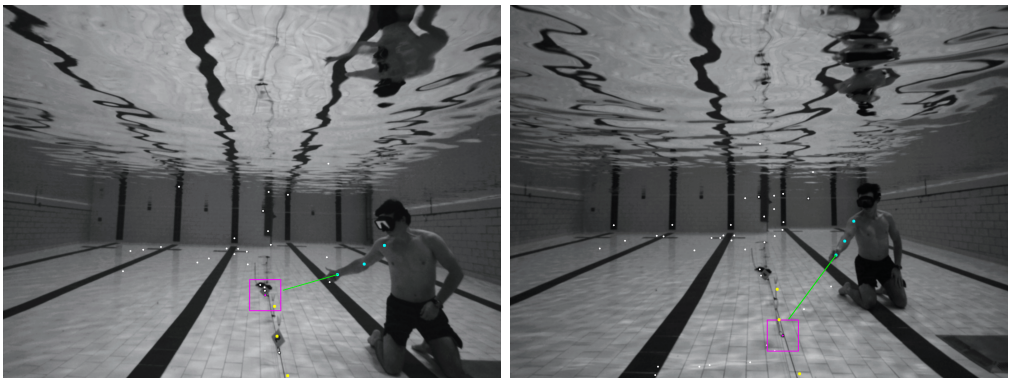
Figure 4.8: Relative frequency that DIP-3D predicts the ground truth object at which the person is pointing during the closed-water testing. Locations given in subcaptions, purple bars indicate the correct object, yellow indicates other objects.

Table 4.2: Closed-water Evaluation: Euclidean distance error between ground truth labels and predicted object locations when correct object location matches predicted.

	Pointing at Object 1	**Pointing at Object 2**	**Pointing at Object 3**	**Error[b] (pixels)**
Person Position 2	65.1 ± 68.5 (**24**)	141.9 ± 122.34 (**42**)	188.59 ± 144.4 (**7**)	121.13 ± 117.77
Person Position 2	-	116.7 ± 153.03 (**16**)	87.47 ± 32.64 (**9**)	106.18 ± 124.77
Error[b] (pixels)	65.1 ± 68.5	134.95 ± 132	131.71 ± 110.62	117.31 ± 119.77

[b] Errors are reported in *mean±standard deviation (observation count)* format in pixel units.

found. Table 4.2 shows the total Euclidean distance error between the located object and the ground truth label when the predicted object is the closest. When the pointing direction and the nearest object are in agreement, the average distance in pixels between the two is 117.31 ± 119.77, or about 10% of the image height of a 1600×1200 pixel image. Samples of the images used for this evaluation can be found in Fig. 4.9. Yellow denotes ground truth points, white points denote all potential features. The sparsity of matching features can be seen in the foreground of Fig. 4.9a. We also include a 2D projection of the 3D pointing direction as the green line segment to show that the object chosen is in the direction of the pointing gesture.



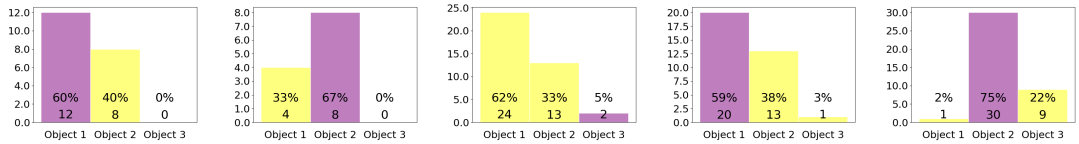
(a) (b)

Figure 4.9: Two examples of images from the DIP-3D closed-water distance evaluation, the diver is pointing away from and then towards the AUV, where DIP-3D is able to accurately locate the direction of the point. Green vector denotes the pointing direction projected to 2D space and white points denote other potential objects. Images enhanced to aid visibility, best seen at 150% zoom.

4.2.4 DIP-3D Directional Open-Water Results

To provide continuity, data collection for this evaluation follows the same guidelines as the closed-water evaluation. We note that although each pose was 'held' for about 6 seconds, movement of the submerged diver and robot due to natural currents and swells caused timing and station-keeping measurements to be less precise.

Fig. 4.10 again shows a breakdown of the count of times the location of the 'closest object' was nearest to the intended target in the sea environment. When both the diver and object of interest are within about 2 meters of the AUV, the correct object is located the majority of the time, even in challenging situations such as Fig. 4.11a where the target is barely in frame. Limited by poor visibility due in part to turbidity, the human pose estimator failed to find corresponding points in both left and right images past the 2 meter marker and the feature detector was unable to locate many corresponding features on the 3 meter marker. Overall, DIP-3D performs accurately when within a reasonable distance to the AUV. We present the total Euclidean distance error for our open-water evaluation in Table 4.3 with the average distance between the



(a) Person Position 1, Pointing at Object 1

(b) Person Position 1, Pointing at Object 2

(c) Person Position 1, Pointing at Object 3

(d) Person Position 2, Pointing at Object 1

(e) Person Position 2, Pointing at Object 2

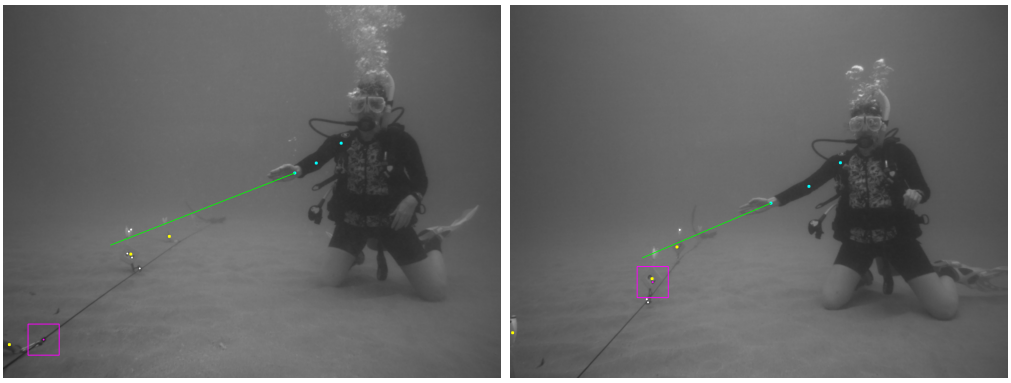
Figure 4.10: Relative frequency that DIP-3D predicts the ground truth object at which the person is pointing during the in-water testing. Locations given in subcaptions, purple bars indicate the correct object, yellow indicates other objects.

Table 4.3: Closed-water Evaluation: Euclidean distance error between ground truth labels and predicted object locations when correct object location matches predicted.

	Pointing at Object 1	**Pointing at Object 2**	**Pointing at Object 3**	**Error[b] (pixels)**
Person Position 1	119.74 ± 84.27(**12**)	61.28 ± 12.8(**8**)	57.78 ± 2.86(**2**)	92.85 ± 69.3
Person Position 2	123.13 ± 33.67(**20**)	31.27 ± 29.7(**30**)	–	68.01 ± 54.84
Error[b] (pixels)	121.86 ± 58.09	37.59 ± 29.67	57.78 ± 2.86	75.6 ± 60.72

[b] Errors are reported in *mean*±*standard deviation (observation count)* format in pixel units.

correct nearest object and annotation being 75.6 ± 60.72 pixels, or about 6% of the image height of a 1600×1200 pixel image. Fig. 4.11 shows two examples of resulting images from this open-water evaluation. The closest object to the pointing vector is in purple while other 'objects' are in white, human pose estimation in blue, annotations in yellow, and a 2D representation of the pointing vector is in green. Fig 4.11a shows the importance of the inclusion of the distance dimension as the 2D pointing vector results much closer to tag 2 rather than the correctly chosen tag 1.



(a) (b)

Figure 4.11: Images from the DIP-3D open-water distance evaluation, the diver is pointing towards the AUV then straight to the side. DIP-3D is able to accurately locate the direction of each point. Images are best seen at 150% zoom.

4.2.5 Runtime

With the current experimental setup, using Mediapipe Pose as the backbone along with the SIFT algorithm, DIP-3D runs at 0.857 fps on an Intel^{TM} i7-6700 CPU, which is acceptable for AUV operations. For each pair of images, both Mediapipe and SIFT run twice, once on each image.

4.3 Challenges

As seen in the bench experiments, we are limited in distance by the accuracy of stereo reconstruction. It is possible that using a wider camera baseline would improve distance results, however, that presents different logistical challenges, particularly in the dimensions of the companion AUV. A further challenge is the detection of an incorrect pose of a diver when they are pointing in front of or cross-body as an incorrect detection will lead to an incorrect pointing location. However, to improve this, the pose estimator itself would need to be greatly improved.

4.4 Conclusion

In this work, we introduce DIP-3D, a modular approach exploiting diver pose information and stereo-visual scene distance information to direct an AUV to a three-dimensional location of interest. Although the use of 3D algorithms in the underwater environment presents many unique challenges, we show through field validations and human assessments that an AUV is able to locate a chosen object in the direction the diver is pointing when multiple objects are in the field of view. When the diver is within mission-specific distance of the AUV, it is highly likely that the direction estimated through sparse stereo triangulation agrees with the direction the diver is pointing. The following chapter, Chapter 5, incorporates the use of diver hand gestures with the previously described methods of locating an object of interest to a diver in order to create a more flexible and situation-dependent communication system between divers and AUVs, allowing a diver to assign a task as needed based on the object of interest.

Chapter 5

Semantic Pointing for Diver-AUV Communication

The previous two chapters (Chapter 3, Chapter 4) provide methods to indicate to an AUV the location of an object that is of interest to the diver. Chapter 3 describes the DIP algorithm, which shows that in the feature-sparse underwater environment, the use of a human pose estimator in conjunction with a single camera is sufficient to create a 2D area of interest from which an AUV can detect an object of interest. Chapter 4, DIP-3D, augments the general premise of pointing to direct an AUV when multiple objects are in the AUV's view by including the distance dimension, *i.e.*, whether the diver is pointing towards or away from the AUV, to locate an object in the 3-dimensional direction of the pointing gesture. These methods are essential to the task-based diver to AUV communication, however as stand-alone communication methods they are still limited to a single pre-determined task such as picking up or taking a close-up video of the object. Most current other methods for the use of pointing gestures (Section 2.3) also rely upon a single specific objective, such as providing a landing location for a drone or picking up an object.

However, in real-world use cases in which AUVs work with divers, such as biological monitoring or infrastructure inspection, it is highly likely that depending on a specific object type, the AUV interaction will change. When we, as humans, utilize a pointing

Figure 5.1: Demonstration of the SPOC communication method in the Caribbean Sea. The LoCO AUV has located the object of interest (yellow box) and received instruction from the diver in the first image. The red light on the AUV in the second image means it received the task "Take a Picture".

gesture to communicate with each other, we often include some form of verbal or contextual input that will allow another person to understand why we are pointing at an object. As an abstract example, in a biological monitoring situation, there may be multiple objectives for a single dive, such as taking close-up images of an invasive species in the habitat as well as collecting a sample. The diver collaborating with the AUV needs an easily usable method to assign a task *in-situ*. Fig. 5.1 shows a diver pointing to an object the AUV should interact with. The diver is making a specialized hand gesture to inform the AUV that the task objective is to take a picture of the object.

In this chapter, we introduce an additional component to task-based communication through pointing, which provides necessary adaptivity for tasks to be allocated as needed. We propose **S**emantic **PO**inting for **C**ommunication (**SPOC**), the use of semantics through a library of hand gestures that instruct the AUV on the operation it will be performing upon an identified object. In addition, the AUV provides feedback to the diver of the action that will be taken. Designed to be integrated with methods such as DIP and DIP-3D for a full communication system, SPOC is made up of three modules:

1. Module 1 identifies where the diver is pointing and locates the object of interest to the diver (DIP or DIP-3D).

2. Module 2 recognizes a hand gesture made by the diver to assign the AUV a task

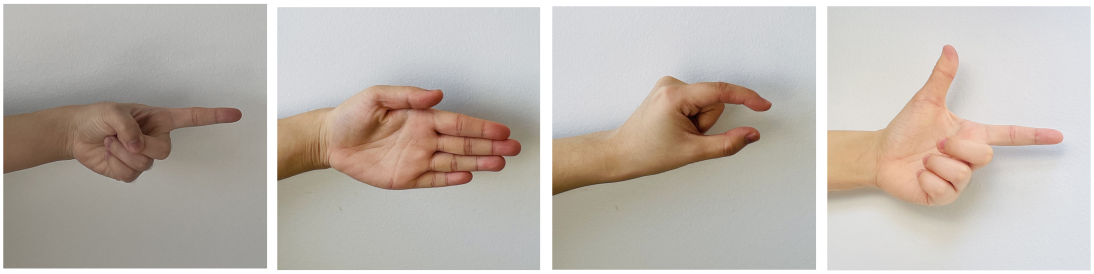
to perform relating to that object.

3. Module 3 provides feedback from the AUV to the diver through LED devices, known as the HREyes [2] system.

In this chapter, we present the second and third modules of the SPOC communication system. For the second module, we create a pointing-based gesture dataset and exploit instance segmentation to extract the diver's hand and classify the extracted segmentation. The third module uses the results of the classified segmentation to notify the diver which task has been assigned through the use of HREyes. Ultimately, we integrate these modules with DIP to provide a flexible communication system. A diver is then able to assign the AUV a task based on objects that are discovered during a mission. Specifically, we discuss dataset creation, provide results of training multiple State-of-the-Art object detection and instance segmentation models on the dataset, describe the integration of the HREyes. We validate the full system as it runs fully onboard the LoCO [12] AUV to provide working demonstrations of the full SPOC communication method in both pool and sea environments.

5.1 Method

The SPOC method is made up of three main components; the first is the module that determines where the diver is pointing and locates an object of interest in that direction (Chapters 3, 4) while the second communicates to the AUV the action that should be taken relating to the identified object. The third module provides feedback to the diver about the action the AUV will be taking. Fig. 5.8 provides an overview of the SPOC communication method. This chapter provides detailed information on the second module, task assignment, and includes creation of a pointing and task-based dataset and methods for gesture recognition as well as the integration of HREyes in the third module to provide feedback from the AUV to the Diver. Additionally, we present the details of the fully integrated system as used in the validation demonstrations onboard the LoCO AUV [12].



(a) General Point (b) Go Here (c) Pick Up (d) Take a Picture

Figure 5.2: Sample hand positions for each of the gestures included in our dataset. Images courtesy of Luoyao Chen.

5.1.1 Task-Oriented Gesture Dataset and Gesture Recognition

We first introduce the SPOC pointing gesture dataset. While other gestural languages such as CADDY [51] provide both direction (left and right) pointing hand gestures and communication gestures such as take a picture, our intent with the hand gesture communication is to seamlessly integrate with a pointing gesture to provide necessary information to an AUV with little additional cognitive load on the diver. These gestures, determined by Walker in [27] and presented in Fig. 5.2 include the following aspects:

- They use a single hand and fingers "point" in some manner, thus integrate with a full-armed pointing gesture.
- The gestures are visually separable, each using different hand and finger configurations, and gestures can be made in any direction.
- The gestures relate to semantics, *i.e.*, "Pick Up" visualizes a gripper and "Take a Picture" relating to English slang *shoot a picture*.
- Finally, the gestures relate to a task that an AUV may need to perform with respect to the object being pointed at.

Additionally, unlike the gesture communication systems described in Section 2.1 where the diver and AUV are assumed to be in close proximity, the gestures for SPOC must be unique enough to be recognized, without additional gear, at the distance a pointing gesture would be considered useful; the diver must have enough distance from the

AUV that both diver and intended object are able to exist in the AUV's view. Our demonstrations occur within a range of distances about 3 to 6 meters between AUV and diver.

Fig. 5.2a represents a general pointing gesture with the index finger pointing outwards and the rest of the fingers and thumb curled inward to the palm. The second (Fig. 5.2b), in which all fingers are extended and held together represent "Go Here", with the intent the AUV should move to the object or location. Fig. 5.2c is "Take a Picture", the index finger is extended similar to the general point while the thumb is also extended at a right angle to the index finger. The final gesture (Fig. 5.2d) denotes "Pick Up" with the intent the AUV will maneuver towards and then pick up the indicated object.

We have compiled an annotated dataset of $6,500$ images containing pointing divers or submerged swimmers. As much of our preliminary testing of algorithms occurs within a closed pool environment, $5,500$ images are from the pool setting. As we intend for this research to assist in real-world situations, we also include $1,000$ images from data we collected off the coast of Barbados in the Caribbean Sea. Each image contains a single diver making a pointing gesture with the right hand. As can be seen in Fig. 5.3 a variety of swimwear and SCUBA gear is represented. In addition, to capture data that will be most similar to that found for AUV use, we include common challenging situations in the dataset, including light refraction and color absorption. As movement between the diver and AUV is expected, images containing lack of distinct boundaries between hand and background due to motion blur are also included (See Fig. 5.4).

DIP or DIP-3D, which both use Mediapipe [42] to locate the joints necessary for pointing, are implemented as the first module. Although the use of pose estimation to recognize the semantic gestures would prevent the necessity of using additional deep methods in our system, joints of the hand are difficult to locate underwater at distance from the AUV as can be seen in Fig. 5.5. Notice all finger joints are located in the palm of the hand. For this reason, pixel-level annotation was chosen to provide an opportunity to evaluate both instance segmentation and object detection models that could be used to recognize the hand gesture. As the appearance of each gesture varies greatly depending on the relation of the location of the diver, AUV, and object, we chose to provide a "Forward" label for images where the gesture is pointing directly toward

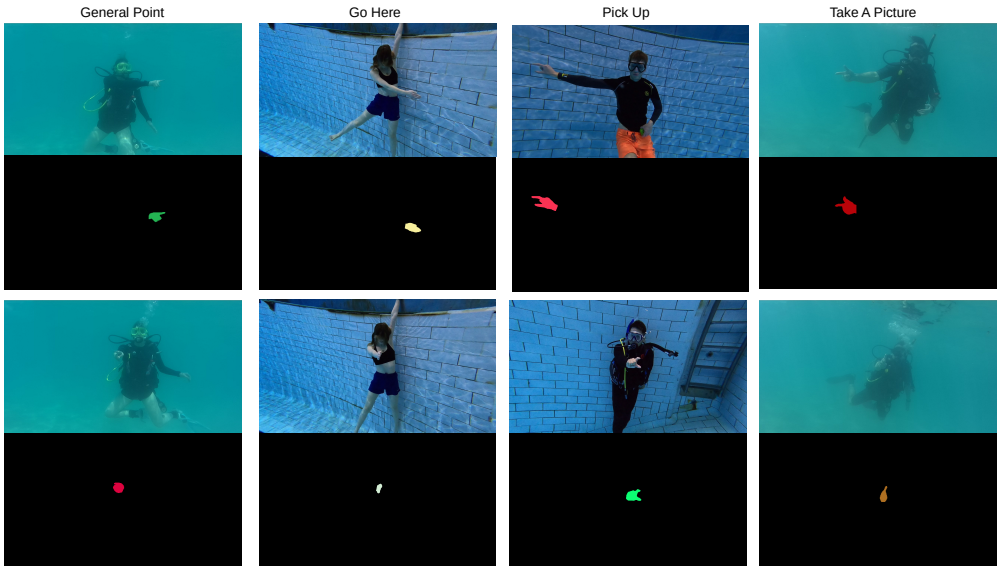


Figure 5.3: Samples of the dataset for semantic pointing for communication. The categories for gestures include: General Point, Go Here, Pick Up, and Take a Picture. The top row shows images where hand gestures are clearly seen pointing to the left or right of the diver. The bottom row is the case where the diver is pointing straight ahead.

Figure 5.4: Examples of challenging images to annotate from the SPOC gesture dataset. In all images the pixel-level boundary between the hand and background is indistinct. Images have been cropped to the gesture and enlarged.

Figure 5.5: A sample image from the SPOC demonstration illustrates the necessity for use of a method other than human pose estimation to recognize a hand gesture. Body joint locations for the thumb, index, and pinky fingers on the right hand are all located in the palm with the use of Mediapipe.

the camera as a distinct object subcategory. While treated as a separate class during the training of models, they are not treated as separate during downstream tasks. Fig. 5.3 shows samples from our dataset of both the underwater images and their corresponding annotated masks, with the top row representing left and right facing pointing gestures and the bottom row the corresponding "forward" direction. The difference in gesture mask shape can be clearly seen. For the challenging scenarios, the best effort was put to annotate the hand, however due to the nature of the challenges the masks cannot be exact. Although we utilize pixel level annotations, our objective is to determine solely which gesture is being performed, not locate the pixels in the image where the gesture exists.

With many recent advances in object detection and image segmentation, we train and evaluate multiple methods of instance segmentation and object detection with our dataset to recognize the hand gesture. As our goal is to recognize the gesture, the specific model and even choice of detection or segmentation is irrelevant other than to determine the gesture more accurately. In all cases, to fit with the end goal of integration with DIP, recognition should rely on a single RGB image. While our implementation of SPOC in the demonstrations use YOLACT [3] with a Darknet-53 backbone, this

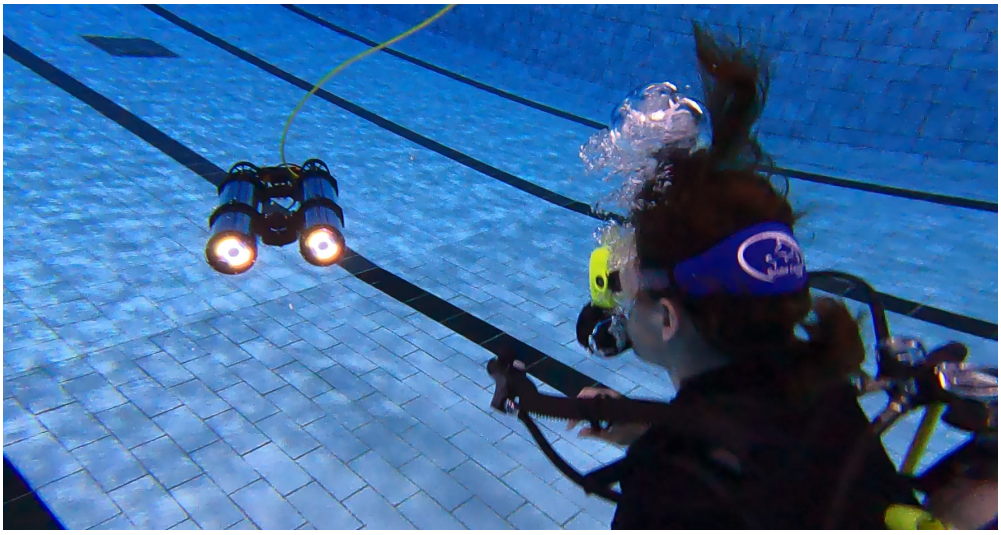
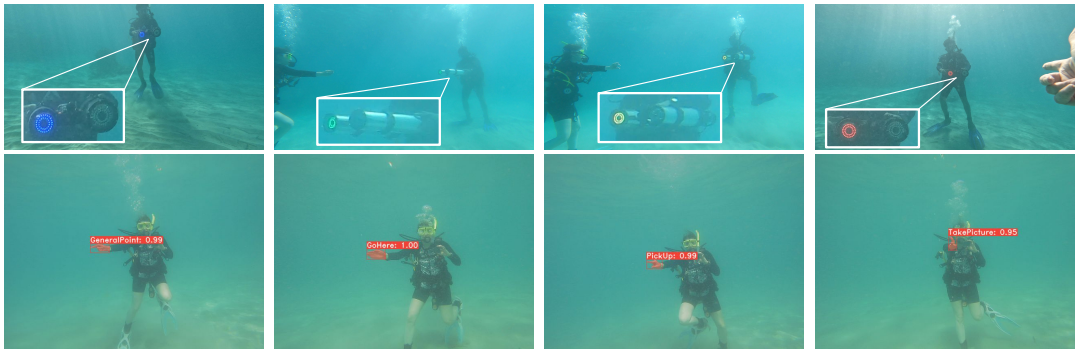


Figure 5.6: The LoCO AUV using HREyes to communicate that it will follow the diver using the *luceme* follow you [2].

is easily substituted as mobile GPUs and detection or segmentation methods improve. Section 5.2 provides evaluation of a variety of segmentation and detection models along with rationale for our use of the YOLACT Darknet-53 model.

5.1.2 AUV Feedback of Task Assignment

To provide feedback to the diver of which task has been assigned we use HREyes [2], a biomimetic LED device, which has been installed on the LoCO AUV for AUV-to-diver communication. Initially designed to communicate through *lucemes*, sequences of lights with specific semantic meanings (See Fig. 5.6), we are able to integrate our gesture library to provide responses to detected hand gestures. The HREyes are composed of LED diodes mapped to tuples with four values (red, green, blue, alpha) allowing us to map each task in our gesture dataset to a specific color. The color mappings for our demonstrations are: blue to "General Point", green to "Go Here", yellow to "Pick Up" and red to "Take a Picture". Fig. 5.7 presents samples of the HREye feedback system with detected gestures from our task-oriented gesture dataset. Although the HREyes are installed in both of the LoCO's enclosures, we utilize only the left HREye as light reflection from the acrylic shell is recorded with the camera, providing unnecessary



General Point: Blue | Go Here: Green | Pick Up: Yellow | Take Picture: Red

Figure 5.7: Demonstration of AUV feedback to the diver using LED light signals. Each light color is assigned to a specific task, as listed in the figure. Demonstration is provided in the Caribbean Sea.

challenges to detection of gestures and object tracking, performed on images taken from the camera in LoCO's right enclosure.

5.1.3 SPOC Full System Integration

The SPOC communication system is designed to locate an object of interest to the diver and then instruct the AUV on a specific action to take with relation to that object. As such, all of SPOC's modules must operate completely onboard the AUV. Fig. 5.8 presents the communication system in entirety. The system is entirely modular and can be updated as software improvements are made or as needed for hardware constraints. Our demonstration of SPOC is performed on the LoCO AUV 7.3 and guides design requirements in this section. These requirements include:

- All perception must use monocular vision from a lowlight camera, thus Module 1 uses DIP rather than DIP-3D as our object locator.
- All computation is performed on an NVIDIA Jetson TX2s. Deep networks do not run simultaneously and are selected because of their ability to run on embedded devices.

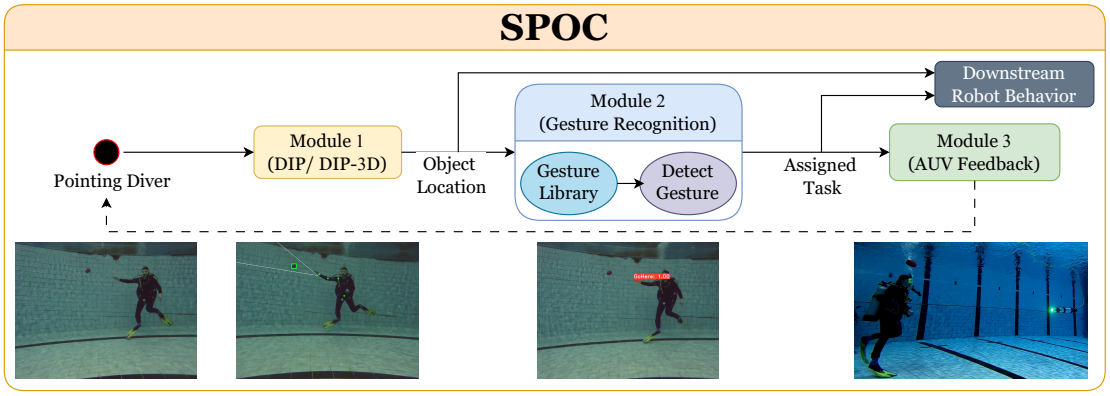


Figure 5.8: Schematic diagram of the SPOC communication system. Module 1 locates an object of interest to the diver through either the DIP or DIP-3D algorithm. Once an object is located, Module 2 uses gesture recognition to determine which task should be performed. Finally, the task information is passed to Module 3 which provides feedback to the diver that a task has been assigned. The object location and task information are also passed to downstream AUV behavior.

- Robot Operating System [73] (ROS1) publisher and subscriber system is used to interface between modules.
- HREyes [2] installed in the front of LoCO AUV's enclosures are used for the AUV to provide feedback to the diver.

The overall SPOC communication system is designed around ROS1's publisher and subscriber system to prevent multiple deep networks from running at the same time. Consecutive modules wait for required information to be passed from the previous module and then begin designated activity. SPOC begins with a pointing diver and object of interest within an AUV's image frame. Images are passed through the first module, DIP, until an object is located. The object's location is then published initiating the second module, task assignment through hand gesture recognition. Once the hand gesture has been recognized, the class information for the gesture is published and then subscribed to by the HREye interface. The class is then mapped to the appropriate color and displayed by the HREye, thus completing the SPOC communication system. Although out of the scope of this project in addition to mapping the task to AUV feedback, the

task library would be mapped to AUV controls systems which, using the located object from module one as a target, would perform the task assigned.

5.2 Evaluation and Demonstrations

In this section we provide multiple evaluations for the SPOC communication method, focusing on the second module, gesture recognition, and overall system. Evaluations for the first module using DIP are found in Section 3.2. We first provide information used in the concept of building of the SPOC gesture recognition module and preliminary results from a subset of the final gesture dataset. We then evaluate multiple methods of instance segmentation and bounding box detection to recognize the hand gesture and provide rationale for our choice of model for demonstrations of the SPOC communication system. We emphasize that the goal is to **correctly identify the gesture**, locating exact boundaries of the gesture or bounding box is secondary.

5.2.1 Preliminary Evaluation

Concept building and preliminary evaluations [79] for the underwater pointing-based dataset and its utilization within the SPOC communication system were performed with a subset of the final dataset. This dataset included 2974 training images and 524 validation images from the **pool environment only and with fewer differences in people performing the gestures**. Multiple segmentation models were trained to determine if gesture recognition at the distance from the AUV needed to be utilized for SPOC was valid. Models trained include Mask R-CNN [80], U-Net [81], PSPNet [82], and DeepLabV3 [83]. All models were trained with and without transfer learning. Quantitative results of preliminary training using Dice Coefficient, which measures the similarity between the ground truth and segmentation, and mean Average Precision as metrics can be found in Table 5.1.

Sample images as seen in Fig. 5.9 show that many networks have difficulty creating an outline of the hand accurately, or they define multiple gestures in a single image, especially in the forward pointing gestures. Moving forward, therefore, we used instance segmentation and bounding box detection rather than semantic segmentation, allowing us to use classification scores to locate the single gesture in the image. We

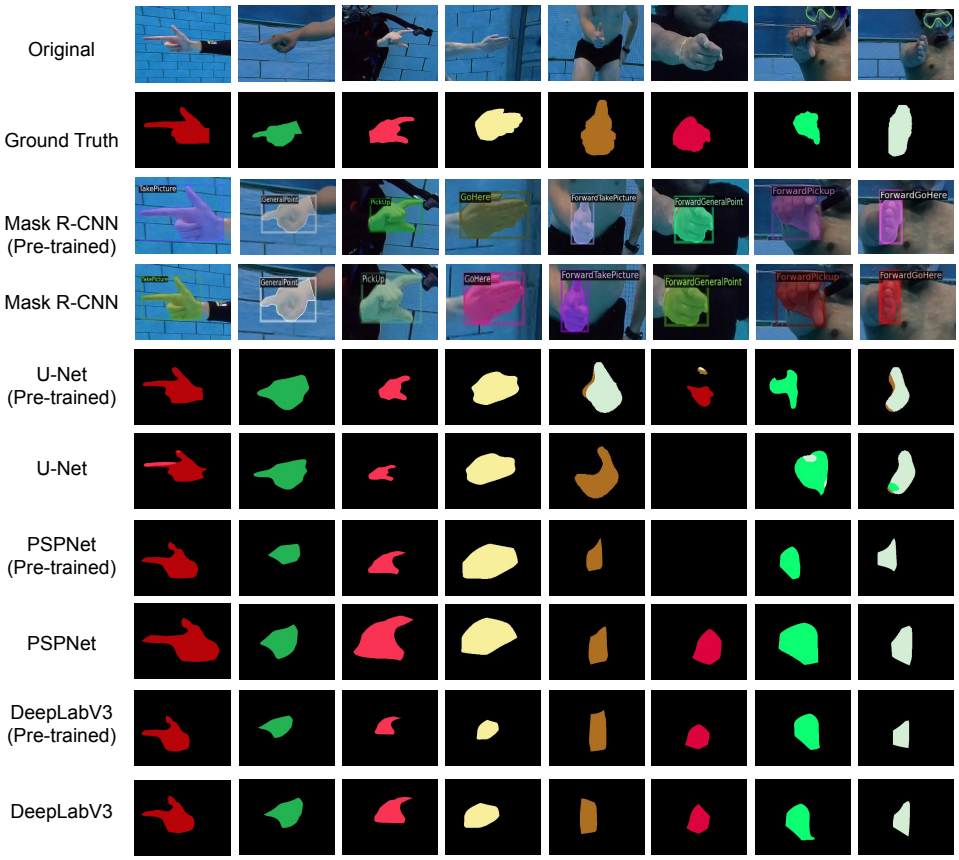


Figure 5.9: Preliminary visual segmentation results for semantic pointing for communication.

Model	Dice Coefficient	AP
Mask R-CNN (Pre-trained)	92.9	92.1
Mask R-CNN	80.5	80.9
UNET (Pre-trained)	92.7	94.2
UNET	93.2	94.3
PSPNet (Pre-trained)	99.8	94.4
PSPNet	98.1	96.2
DeepLabV3 (Pre-trained)	99.8	98.6
DeepLabV3	98.5	97.1

Table 5.1: Preliminary results of segmentation model training for hand gesture recognition on a subset of the final dataset. Quantitative evaluation uses Dice coefficient and Average Precision (AP). Higher scores are better for each metric.

also found that evaluations using images from the DIP application onboard the LoCO AUV (Section 3.3) required scale augmentation during training as the original training had difficulty locating smaller gestures and that a vertical flip augmentation allows for left-handed gestures to be recognized in the case that the diver is left-handed and DIP is altered to the left side.

5.2.2 Evaluation of Gesture Recognition Models

As the overall goal of the second module is to recognize the hand gesture being performed, we are able to train a variety of instance segmentation and object detection models on our gesture dataset and choose the model that fits our needs. All models are trained on the same train, validate, and test split from the gesture dataset and utilize fine-tuning of models trained on the COCO dataset [84]. During training and evaluation all models use augmentations of scaling down the image and mirroring the image to emulate divers that are further away or pointing with their left arms. Instance segmentation models trained include YOLACT [3] with multiple (Darknet-53, Resnet-50, Resnet-101) backbones and Mask R-CNN [80] with a Resnet-50 backbone. Detection-based models

Model	AP50	AP75	AP50-95
YOLACT (Darknet-53)	99.62	96.36	74.66
YOLACT (Resnet-50)	99.47	96.88	75.01
YOLACT (Resnet-101)	99.48	97.26	74.85
Mask R-CNN (Resnet-50)	92.86	79.83	64.38

Table 5.2: Performance comparison of instance segmentation models on SPOC's gesture recognition dataset. Average Precision is given for each trained model backbone. Higher scores represent better results.

Model	AP50	AP75	AP50-95
YOLACT (Darknet-53)	99.62	98.82	83.00
YOLACT (Resnet-50)	99.47	98.83	82.99
YOLACT (Resnet-101)	99.48	98.67	81.18
Faster R-CNN (Resnet-50)	88.28	63.49	57.26
Faster R-CNN (Resnet-101)	88.01	72.12	59.50
RetinaNet (Resnet-50)	93.96	66.25	59.94
YOLOv8 (Nano)	98.86	98.50	91.47

Table 5.3: Performance comparison of object detection models on SPOC's gesture recognition dataset. Average Precision is given for each trained model backbone. YOLACT networks are the same weights as those trained with instance segmentation data. Higher scores represent better results.

include Faster R-CNN [85] with Resnet-50 and Resnet-101 backbones, RetinaNet [66] with a Resnet-50 backbone and YOLOv8 [86] Nano. Mask R-CNN, Faster R-CNN, and RetinaNet are implemented through Detectron2 [87].

Average Precision (AP) scores for the segmentation models are located in Table 5.2 and AP scores for the detection models in Table 5.3. In addition to returning a mask, YOLACT (See Fig. 5.10 also provides bounding box location of objects and thus is included in the bounding box detector results as well.

YOLACT outperforms the majority of other detection and segmentation models, excluding YOLOv8 at AP50-95. As the results for these models are similar at AP50 and AP75, either model could be reliably used for our purposes. As the YOLACT models

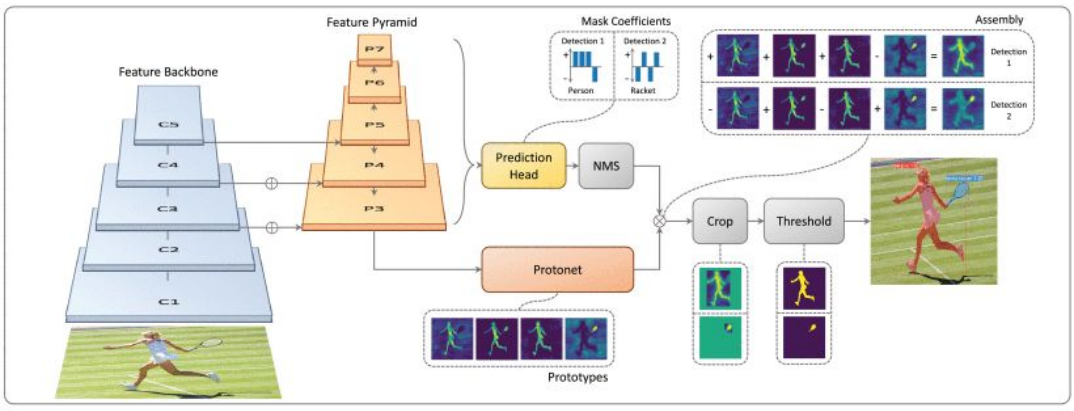


Figure 5.10: YOLACT Architecture from Bolya, *et al.*[3].

are slightly higher we perform our system validation with this model, in particular YOLACT with the Darknet-53 backbone. Ultimately, as we do not create our own model to recognize gestures, future implementations of SPOC can easily be updated with alternate detection models.

5.2.3 Validation Onboard the LoCO AUV

SPOC was deployed for validation onboard the LoCO AUV in both closed-water (pool) and open-water (Caribbean Sea) environments. Once the communication system was activated, the full system ran autonomously. In the pool demonstration a tether was used to activate the system and monitor the results. However, the Caribbean Sea demonstrations were run tetherless, without monitoring, and activated through the use of fiducial markers [21] shown to LoCO's camera. In both sets of demonstrations the SPOC communication system was used successfully three out of three times to locate an object of interest to the diver, recognize the gesture performed, provide feedback of the correct gesture to the diver through the HREye and move towards the object of interest with the use of a Proportional-Integral-Derivative (PID) controller, simulating the downstream task that would be performed upon the object. The top row of Fig. 5.11 shows the resulting images from AUV and diver view as SPOC moves through the modules. The AUV was located 4 to 6 meters away from the object and diver at the beginning of each demonstration. Fig. 5.11 shows the respective images from the sea

Figure 5.11: Demonstration of the SPOC communication system in a closed-water environment is shown in the top row and in the Caribbean Sea on the bottom row. The object is located through DIP in the first module, the task is assigned in the second module, and feedback is provided from the AUV to the diver in the third module.

demonstration on the bottom row. Exact distance between AUV and diver is difficult to determine due to movement of diver and AUV, but they occurred within the smaller end of the spectrum, about 3 to 4 meters. In all demonstrations, about 10 seconds pass between the object location detection in Module 1 and detection of the gesture in Module 2. As can be seen in the differences in the image where the detections occurred, the AUV was able to detect all necessary components for the communication system with more movement between the diver, object and AUV with the AUV being slightly closer to the diver at the time the communication occurred.

5.3 Conclusions

This chapter presents a full communication system to provide an AUV the location of an object as well as the task assigned to that object by a diver. In addition, the AUV is able to provide feedback, so the diver is able to be sure the correct task will be performed. Advantages of this system include being able to provide on site task assignment in an environment that is very apt to change. This method also allows for flexibility as unexpected task parameters may change during a dive, preventing prior

programming from being effective. To do this we create a task-oriented gesture dataset created to be utilized with pointing gestures, integrate a gesture recognition module with a previously described pointing gesture communication method (Chapter 3) and AUV feedback model, and demonstrate the full system in both pool and sea environments onboard an AUV. As this is a vision-based system, it relies on the ability to detect different elements in both module 2 and module 3. Challenges to this and other underwater vision-based systems are often exacerbated by poor visibility. Chapter 6 presents our investigation into task-based underwater image enhancement, an effort to enhance AUV vision to improve the ability to use visual perception in a wider variety of underwater scenarios.

Chapter 6

Task-Driven Image Enhancement

In this chapter, we present or investigation into the topic of task-based image enhancement [10]. For many of the collaborative tasks mentioned throughout this dissertation, visual perception is one of the preferred tools due to the availability of cameras and their passive, energy-efficient sensing capabilities. However, due to environmental factors, such as turbidity and image distortions caused by light absorption, refraction, and attenuation, the utility of visual sensing can be severely hampered. Recent work has thus looked into underwater image enhancement techniques for marine computer vision and has yielded promising results.

As mentioned in Section 2.4, enhancement techniques can be broadly classified into two categories: those taking the classical approach and those using deep machine learning algorithms. The latter methods have seen a rapid rise recently, using both Convolutional Neural Networks (CNNs) [88] and Generative models, *e.g.*, Generative Adversarial Networks (GANs) [89] and Variational Autoencoders (VAEs) [90]. However, the goal of these methods is usually to either alter the image to match in-air images or improve general image quality, as determined by a human viewer, focusing solely on image aesthetics. In addition, many of these methods are designed to be a part of the post-processing of recorded images or video, in which case the real-time usability is of less importance and therefore not prioritized during design. For use with marine robotics, if the method fulfills the requirements necessary to be of use on-board an AUV, the enhanced images can subsequently improve the performance of vision tasks

downstream, such as detection or tracking. However, in this approach, the improvement in vision tasks is a by-product and not an integral component of the enhancement algorithm. Thus, it is possible that general enhancement may not help or may even hinder specific vision tasks, depending on task requirements and environmental effects. As these AUVs in many circumstances work with a team of divers, the ability to detect, track, and follow divers becomes an essential capability.

For underwater diver tracking, traditional methods and more recent tracking-by-detection approaches have been successfully used. Islam et al. [17], extending the Fourier Tracker algorithm [91], uses frequency-domain motion signatures of divers as a 'filtering' criteria to identify tracks extracted using Hidden Markov Models [92]. Current methods of visual diver detection and tracking include training state-of-the-art (SOTA) detection models on diver datasets (*e.g.*, [93]) using CNN-based models [94] and tracking algorithms that take into account diver motion (*e.g.*, [95, 96]). As these methods are dependent on the dataset used in training, there will be situations, such as due to environmental factors (see Fig. 6.1), where the trained detector does not work adequately without pre-processing or re-training. These are the cases, as explained above, where it is shown that first using an image enhancement model and then performing diver detection can improve results [60, 97]. Here, we attempt to **improve diver detector performance using a generative image enhancement model that incorporates a detector's feedback during the training stage**. In other words, instead of relying on generically-improved images, we specifically enhance images such that diver detection is improved. Methods for integrating information, such as object location and classification with GAN image enhancement, have been used for improving small-object detection [98, 64, 65, 63], and reducing blurriness of images [99]. However, due to the properties of the underwater environment and run-time requirements, the availability of these models to work within the constraints of AUVs is unknown. As a basis for the method we investigated, we attempt to build upon a network that has been shown to have real-time use capabilities and incorporate a diver detector directly during the training process. In summary, we make the following contributions in this chapter:

- We discuss a proposed method for improving diver detection through a **D**etection-driven **Un**derwater **I**mage **E**nhancement **GAN**, which we refer to as **DUnIE-GAN**. Rather than re-training an image detector for differing water environments,

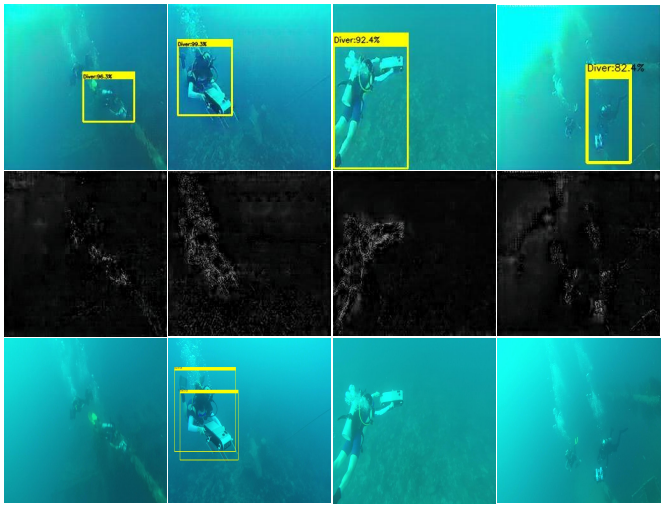


Figure 6.1: A few samples showing the gain in diver detection accuracy on the detection-driven enhanced images (top row) compared to the raw images (bottom row) from our first detector experiment. Their respective *enhanced* pixel differences (middle row) show that the underlying image statistics in the foreground regions are improved, which positively impacts the detection performance.

our model focuses on improving the images captured by the AUV before they are sent to the diver detector. We do this through learning image-to-image translation from a *distorted* to *enhanced* domain, while also making use of bounding box locations and classification information provided by a trained diver detection model to guide the enhancement towards better detection results at inference time.

- We experimentally attempt to validate the positive effects of incorporating detection information into a GAN-based image enhancement model [4] in evaluations with two different pre-trained diver detectors.

6.1 Methodology

As shown in Fig. 6.3, our proposed DUnIE-GAN model integrates a trained diver detector with a GAN-based image enhancement model. Similar to the existing paired learning-based models [4, 60], given a source domain X of distorted underwater images and target domain Y, our goal is to learn a mapping for an image generator: $G : X \rightarrow Y$.

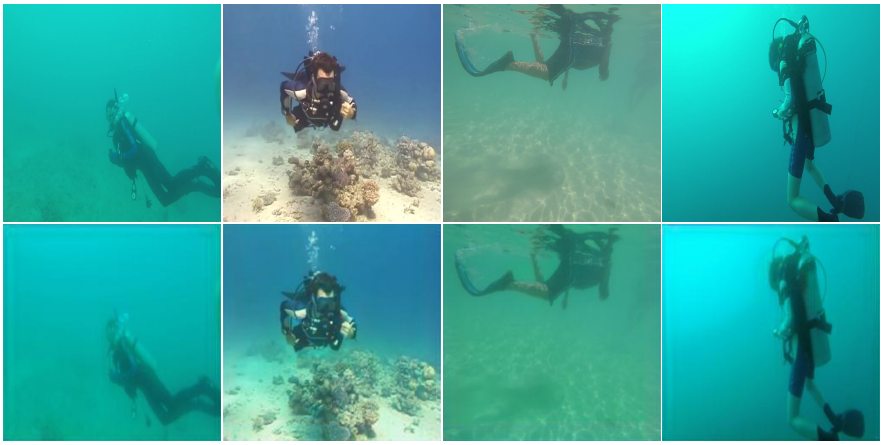


Figure 6.2: A few sample training pairs are shown; images on the top and bottom row belong to the *target domain* and *distorted domain*, respectively.

However, rather than learning a map that perceptually enhances the resulting image, our goal is to enhance the images in a way that focuses enhancement on the ability to detect divers. **It should be noted that the enhancement is learned on the underlying image statistics guided by a specific diver detector's feedback, hence the resulting images may not appear perceptually enhanced**. For this research thread, we investigated the use of two diver detectors, both trained Single Shot Detectors (SSD) [28] model with a MobileNet V2 backbone.

6.1.1 Data Preparation

Our supervised training dataset is prepared with the end goal of better diver detection performance. Our training and ablation datasets utilize images containing single divers in a variety of oceanic environments found in [93] and [94]. To limit the cases of no divers being detected during training, we use the trained diver detector model to select images in which the diver is detected with a minimum *confidence score* of 0.55. These images are then distorted based on the standard *style transfer* method adopted in [4] to create paired instances; some samples are shown in Fig. 6.2. A second test set, made up of randomly selected images from [93] and [94], is utilized for the generalized experiments. Hand-labeled annotations of all images are used as ground truth bounding box data both during the training process and experimental analyses.

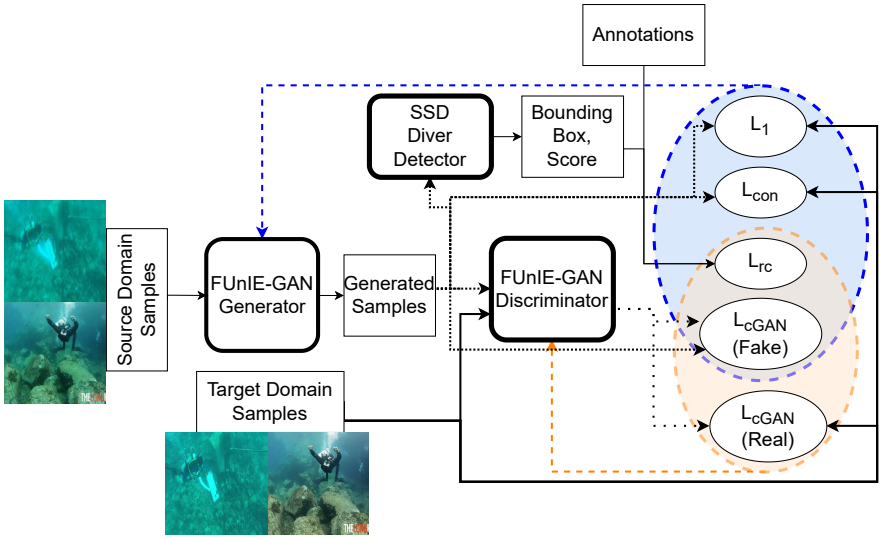


Figure 6.3: An outline of the proposed learning pipeline: we use the generator and discriminator networks of a pre-trained FUnIE-GAN model [4] alongside a trained SSD-based diver detector. Note that these modules can be replaced with any other image enhancement and object detection models. In Sec. 6.2 we present results using results of two different trained SSD-based diver detectors.

6.1.2 Overarching Model

Following the standard paired training pipelines of [60, 4], we use distorted images as input to the generator. The generator then produces synthetic images, which are sent to the discriminator to determine their *validity*. As mentioned earlier, the generator and discriminator then play a min-max game to eventually learn the mapping $G : X \rightarrow Y$. In our model, in addition to the discriminator, the synthetic images are passed through a trained diver detector to generate bounding boxes and classification scores. These are, in turn, utilized in an additional detection-based objective function.

6.1.2.1 Generative Adversarial Network Architectures

The GAN portion of our model is based on the FUnIE-GAN [4] architecture. The generator network follows the U-Net [81]-like encoder-decoder structure with an input resolution of $256 \times 256 \times 3$; the encoder eventually learns 256 feature-maps of size 8×8.

The decoder uses these feature maps to learn to generate an enhanced $256 \times 256 \times 3$ image. The discriminator uses the Markovian Patch-GAN [59] architecture, which takes as input the combination of real and generated images in $256 \times 256 \times 6$ form and outputs a $16 \times 16 \times 1$ real or fake response. Specific network details on FUnIE-GAN architecture can be found in [4]. As the generator and discriminator networks are not altered, we can take advantage of seeding our model with a pre-trained FUnIE-GAN model to initiate training.

6.1.2.2 Diver Detector

As shown in Fig. 6.3 we incorporate a trained SSD (MobileNet V2) diver detector model outside the GAN architecture, *i.e.*, independent of the generator and discriminator networks. This design is intentional for creating a modular network that could be used with various types of detectors or image enhancement models. For the training process, we lower the detection threshold to maximize detection potential. However, we choose to return only the detected bounding box with the highest classification score as all images contain a single diver. For each batch during training, the detector runs on every generated image and outputs a bounding box, $b = (x_{min}, y_{min}, w, h)$, as well as the confidence score, $s \in [0, 1]$; here, (x_{min}, y_{min}) is an image coordinate, whereas w and h represent width and height of the box, respectively, and s represents the probability of a correct detection. During our investigation, we look at the effects of using two different trained models, the first used in [94][1] and the second trained on images from the VDD-C dataset [93].

6.1.2.3 Objective Function

Similar to the learning pipeline of FUnIE-GAN [4], our objective function incorporates global similarity and image content losses, as well as the conditional adversarial loss to maintain the original image statistics. Additionally, we design a detection feedback objective function based on the combination of bounding box regression and classification loss terms.

Base GAN Loss: We use the standard conditional adversarial loss [100] to learn

[1] The model weights are imported from `https://github.com/IRVLab/deep-diver-following`.

the mapping of $G : \{X, Z\} \rightarrow Y$, where $X(Y)$ represents the distorted (target) domain and Z represents random noise. With D representing the discriminative model, the conditional GAN loss is defined as

$$\begin{aligned}\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{X,Y}\big[\log D(Y)\big] \\ + \mathbb{E}_{X,Y}\big[\log(1 - D(X, G(X, Z)))\big],\end{aligned} \tag{6.1}$$

whereas the loss terms for global similarity ($\mathcal{L}_1$) and image content ($\mathcal{L}_{con}$) are defined as follows:

$$\mathcal{L}_1(G) = \mathbb{E}_{X,Y,Z}\big[\|Y - G(X, Z)\|_1\big], \tag{6.2}$$

$$\mathcal{L}_{con}(G) = \mathbb{E}_{X,Y,Z}\big[\|\Phi(Y) - \Phi(G(X, Z))\|_2\big]. \tag{6.3}$$

Here, Φ refers to high-level feature contents extracted by the `block5_conv2` layer of a pre-trained VGG-19 model.

RC Loss: The regression and classification loss, $\mathcal{L}_{rc}$, is determined by the output of the diver detector. Let s be the confidence score, and b_d and b_a be the bounding box information from the detection and ground truth annotations, respectively. The detection loss is then calculated as

$$\mathcal{L}_{rc} = Smooth_{L_1}(b_d, b_a) + \mathcal{L}_{cls}(s), \tag{6.4}$$

where $Smooth_{L_1}(b_d, b_a)$ is defined as follows [101]:

$$Smooth_{L_1}(b_d, b_a) = \begin{cases} 0.5(b_d - b_a)^2 & \text{if } |b_d - b_a| < 1, \\ |b_d - b_a| - 0.5 & \text{otherwise.} \end{cases} \tag{6.5}$$

This is used to determine a bounding-box regression loss, while $\mathcal{L}_{cls} = -\log s$ allows our model to take advantage of the confidence score to compute classification loss. The formulation of these loss terms is inspired by the methods used in [98, 64] and [63]. However, our application of these loss terms differs in that we use information directly from the diver detector rather than introducing additional branches to the discriminator or to the fully-connected layers. Moreover, when the detector is unable to locate a diver in the generated image, an artificial bounding box and score are introduced to create a high $\mathcal{L}_{rc}$ to penalize inaccurate detections. Our ablation experiments investigate these effects during the training phase (see Section 6.2.1.2).

Objective function: Using the loss components defined in [4] as well as the new detector-based loss formulated in Eq. 6.4, our final objective function is expressed as

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda_1 \mathcal{L}_1(G) + \lambda_c \mathcal{L}_{con}(G) + \mathcal{L}_{rc}. \tag{6.6}$$

$\lambda_1 = 0.7$ and $\lambda_c = 0.3$ are the loss scaling factors as determined by [4].

6.2 Evaluations

As mentioned in the introduction, this chapter provides research into an exploration of a task-based image enhancement model therefore, although no firm conclusion can be made about the efficacy of this type of architecture, in this section we provide the experimental evaluation of the results of training our model with two different diver detection models. The first subsection presents original results using the [94] model. As can be seen in the following section, the model was unable to match or improve upon image detection against raw images, this led to our secondary investigation utilizing a different trained detector.

6.2.1 Evaluation on First Pre-trained Diver Detector

6.2.1.1 Implementation Details

Our model is implemented using PyTorch libraries[2] for the majority of the optimization pipeline. However, the trained diver detector is implemented in Tensorflow [102], so our environment includes both libraries. Each DUnIE-GAN model is trained on a Linux Machine using an NVIDIA GTX 1080 GPU, on 8k images with single annotated divers. The various categories of test sets are created by images from [93] and [94] that were excluded from the training. More detailed information on these test sets can be found in Section 6.2.1.3.

Our model uses pre-trained weights of FUnIE-GAN on the EUVP dataset[3] , and then it is further tuned with our training pipeline in Eq. 6.6. The convergence of different training models falls between 80-160 epochs in addition to the original FUnIE-GAN

[2] PyTorch libraries: `https://pytorch.org/`

[3] FUnIE-GAN (PyTorch) model is available at: `https://github.com/xahidbuffon/FUnIE-GAN`

training. In particular, the model using $\mathcal{L}_{rc}$ to optimize the generator and discriminator is trained for 120 epochs, while the model with $\mathcal{L}_{rc}$ in the discriminator is trained for 140 epochs. A batch size of 32 is used in all the training modes; here, we use a larger batch size than is used in [4] to increase the possibility of detectable divers in each batch.

6.2.1.2 Loss Ablation

We first present an ablation study based on how the object detection loss, $\mathcal{L}_{rc}$, is integrated with the GAN architecture. We abbreviate the location of the application of $\mathcal{L}_{rc}$ loss optimizations as follows:

- Applied to generator and discriminator: DUnIE-GAN-B
- Applied to only generator: DUnIE-GAN-G
- Applied to only discriminator: DUnIE-GAN-D
- Applied to neither: DUnIE-GAN-N

As we are interested in increasing the diver detection accuracy, we use DUnIE-GAN in conjunction with the diver detector used in training our model in our ablation study. We use Average Precision (AP) [103] and Intersection over Union (IoU) as our performance metrics. Mean IoU per image is found, with penalties of an additional score of 0 included for each extra detection per image. The mean of all IoUs over the dataset is then taken. All tests are performed with the diver detector score threshold of 0.55, as in the real-world use case we prefer an above average confidence score for the AUV to act. The overall performance is determined based on 415 distorted single diver images that are not used for training or validation. The results are illustrated in Table 6.1. The baseline for this evaluation is determined by using the diver detector directly on the distorted images.

It can be seen that while all models increase detection results, DUnIE-GAN-B and DUnIE-GAN-D have the highest scores of IoU and AP, respectively. To test these results on a more generalized task, we use these two models in comparison with other underwater enhancement tasks on non-distorted images. Liu *et al.* [65] mentions that adding a loss similar to $\mathcal{L}_{rc}$ to the generator may minimize performance on real images,

Table 6.1: Ablation results for $\mathcal{L}_{rc}$: AP and mean IoU on distorted underwater images of single divers.

Model	AP (%)	IoU (%)
Distorted Images	63.38	33.9
DUnIE-GAN-B	70.08	44.2
DUnIE-GAN-G	69.34	43.8
DUnIE-GAN-D	71.26	43.8
DUnIE-GAN-N	69.78	43.2

as the model will be tuned towards good detections on synthetic images. However, Zhang *et al.* [63] suggests including a similar loss in optimizing the generator to help enforce a semantic level similarity constraint. These insights also lend to the interest of examining the results of both DUnIE-GAN-B and DUnIE-GAN-D in further experiments. In addition, we make note that as our base network architecture and losses are inspired by FUnIE-GAN, results of DUnIE-GAN-N would be similar to those of pre-training FUnIE-GAN with the EUVP dataset and then additionally training with our diver dataset.

6.2.1.3 Ocean Image Test Set and Comparison Models

We compare DUnIE-GAN with several underwater image enhancement models, based on both the results of the task of diver detection and the results of image quality metrics. The test set, like the training set, contains images taken directly from [94] and [93] in various categories based on thresholding the detected divers at 0.55 with the detector. The various categories include:

- Single Diver-Detected (SD-D): 415 images
- Single Diver-Undetected (SD-U): 1014 images
- Multiple Divers-All Detected (MD-D): 1021 images
- Multiple Divers-Some Undetected (MD-U): 1025 images

Table 6.2: Quantitative comparison of diver detection performance on enhanced images based on AP and mean IoU. Higher scores are better.

	Dataset	Unenhanced	MS-Retinex	WaterNet	UGAN	FUnIE-GAN	Deep SESR	**DUnIE-GAN-B**	**DUnIE-GAN-D**
AP(%)	SD-D	91.86	11.24	56.46	62.22	64.32	69.88	75.68	68.73
	SD-U	25.56	1.08	7.89	9.18	19.84	20.23	18.36	19.03
	MD-D	92.00	2.83	41.11	51.66	66.67	82.27	71.76	76.34
	MD-U	54.45	2.42	21.43	29.38	40.78	51.62	44.89	47.01
IoU(%)	SD-D	67.3	12.5	33.0	38.3	42.3	45.5	48.2	44.0
	SD-U	2.8	1.6	2.3	3.2	5.0	5.3	9.2	7.5
	MD-D	64.9	4.5	23.9	27.5	36.5	49.3	41.8	44.2
	MD-U	25.4	3.4	12.0	14.6	18.2	22.1	21.5	21.3

These sets are not synthetically distorted; also note that Single Diver-Detected are the undistorted images used in the creation of images for *testing* DUnIE-GAN ablation and have not been seen by the model during training. The test set is broken down in this manner to more clearly see results in relation to how detections on enhanced images relate to how well the diver detector performs on the unenhanced images. The enhancement models that we use for comparison include the physics-based model multi-scale Retinex (MS-Retinex) [104] and the learning-based models WaterNet [58], Underwater GAN (UGAN) [60], Fast Underwater Image Enhancement GAN (FUnIE-GAN) [4], and Deep Simultaneous Enhancement and Super-Resolution (Deep SESR) [57]. All the learning-based models are trained holistically on the EUVP (paired) dataset [4].

6.2.1.4 Diver Detection Comparisons

For the comparison of diver detection performance, we use the same metrics as in the loss ablation: mean AP and mean IoU. We evaluate the results of the diver detector after enhancing the images using the various models. The baseline is created by running the diver detector directly on the unenhanced dataset images. The results of this comparison can be seen in Table 6.2. We find an interesting result that in this dataset, none of the enhancement models return AP results better than the unenhanced images and so is something to improve upon. However, both DUnIE-GAN model results are competitive with the enhancement models, while DUnIE-GAN-B achieves the highest scores on AP and IoU for the SD-Detected dataset.

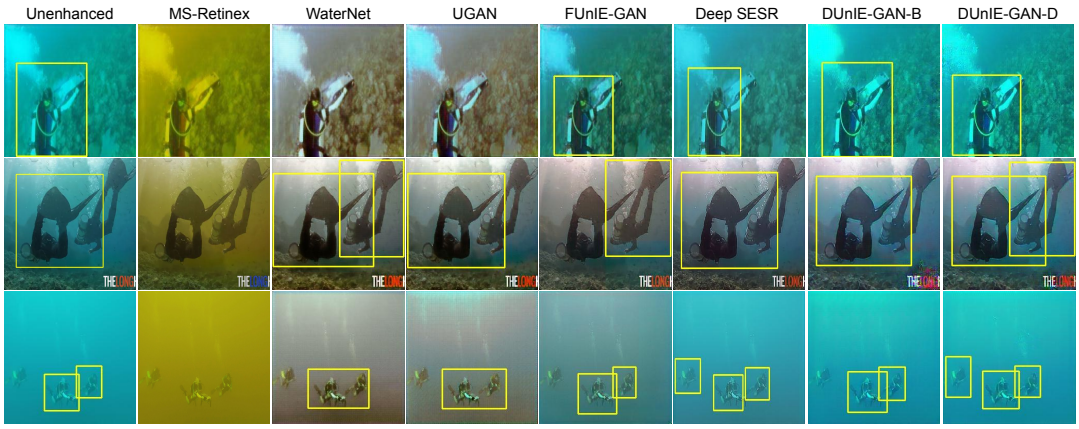


Figure 6.4: Sample qualitative comparisons of diver detection performances by the enhancement models; all detections with a confidence score over 0.55 are shown.

In regard to mean IoU, we again observe favorable results with both DUnIE-GAN models, particularly in the case of SD-Undetected, having the highest scores. As we intentionally train the model for minimization of distance between annotated and detected bounding boxes, an increase in IoU when objects are detected is an encouraging result. This is also important as the return of accurate and concise bounding boxes by a detector is integral in AUV tasks such as diver following.

6.2.1.5 Image Quality Analysis

To analyze enhancement, we conduct a performance comparison of all models using the underwater image quality metric (UIQM) [105]. We choose this evaluation in particular as it is inspired by properties of human vision systems and evaluates image quality in methods that are shown to relate to human preference. A higher overall score lends itself to a higher correlation with perceived image quality. Results of this comparison are found in Table 6.3. As our goal is to show task-specific enhancement, rather than enhancement for human perception, we are not looking to have the *best* score, but rather are interested in these results in conjunction with the detection results of the above section. We find that while WaterNet and UGAN receive high UIQM scores, this does not translate to high detection results with an off-the-shelf detector. See Fig. 6.4 for a visual comparison. As our model is designed for task-specific goals, depending on

Table 6.3: Comparison of average UIQM, PSNR, and SSIM, scores of enhancement models over the test set; scores are shown as $mean \pm \sqrt{variance}$.

	Dataset	Unenhanced	MS-Retinex	WaterNet	UGAN	FUnIE-GAN	Deep SESR	**DUnIE-GAN-B**	**DUnIE-GAN-D**
UIQM	SD-D	1.05 ± 0.37	1.01 ± 0.38	2.48 ± 0.40	2.58 ± 0.36	$2.37 \pm .048$	1.87 ± 0.42	1.46 ± 0.34	1.40 ± 0.36
	SD-U	1.20 ± 0.36	1.16 ± 0.39	2.78 ± 0.36	2.82 ± 0.37	2.57 ± 0.44	2.05 ± 0.44	1.52 ± 0.33	1.48 ± 0.33
	MD-D	0.87 ± 0.31	0.83 ± 0.29	2.31 ± 0.26	2.44 ± 0.25	2.15 ± 0.37	1.66 ± 0.31	1.33 ± 0.33	1.27 ± 0.32
	MD-U	0.90 ± 0.37	0.86 ± 0.37	2.40 ± 0.33	2.51 ± 0.30	2.18 ± 0.46	1.70 ± 0.40	1.34 ± 0.36	1.28 ± 0.37
PSNR	SD-D	—	29.09 ± 1.1	21.06 ± 2.0	23.14 ± 2.0	27.70 ± 2.1	31.09 ± 2.1	31.23 ± 1.9	32.12 ± 1.9
	SD-U	—	29.90 ± 1.1	21.61 ± 2.1	24.83 ± 2.4	29.68 ± 3.1	31.42 ± 2.6	30.01 ± 3.4	30.89 ± 3.2
	MD-D	—	28.85 ± 1.2	20.73 ± 1.5	22.58 ± 1.3	27.71 ± 2.3	31.26 ± 1.9	31.01 ± 1.9	32.06 ± 1.7
	MD-U	—	28.98 ± 1.2	20.67 ± 1.8	22.98 ± 1.8	27.97 ± 2.4	31.62 ± 2.1	31.30 ± 2.1	32.38 ± 2.0
SSIM	SD-D	—	0.43 ± 0.04	0.58 ± 0.03	0.59 ± 0.03	0.68 ± 0.03	0.75 ± 0.04	0.84 ± 0.05	0.86 ± 0.04
	SD-U	—	0.43 ± 0.01	0.56 ± 0.02	0.58 ± 0.02	0.66 ± 0.03	0.74 ± 0.06	0.83 ± 0.06	0.83 ± 0.07
	MD-D	—	0.43 ± 0.04	0.59 ± 0.02	0.59 ± 0.03	0.69 ± 0.03	0.76 ± 0.03	0.84 ± 0.03	0.85 ± 0.03
	MD-U	—	0.43 ± 0.03	0.58 ± 0.02	0.58 ± 0.03	0.69 ± 0.03	0.76 ± 0.04	0.84 ± 0.05	0.86 ± 0.04

the task, it may be preferable for the enhanced image to have UIQM scores similar to those of the original image.

We also consider two standard image quality metrics named Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity(SSIM) [106]. PSNR is based on MSE and measures the reconstruction quality of the enhanced images as compared to the ground truth, in this case unenhanced images. As the results in Table 6.3 indicates, DUnIE-GAN scores competitively with Deep SESR for this metric. On the other hand, SSIM evaluates the luminance, contrast, and image structure of a given image. Both DuNIE-GAN networks perform best for this metric. Detailed results for all SOTA models are provided in Table 6.3.

6.2.2 Evaluation on Second Pre-trained Diver Detector

6.2.2.1 Implementation Details

This version of our algorithm is implemented using a different pre-trained diver detector with all modules utilizing PyTorch libraries[4] . Each DUnIE-GAN model is trained using an NVIDIA v100 GPU on 8k images with single annotated divers from [93] and [94]. As the pre-trained detector was trained on data from [93], we include these images in our training set to seed raw images where the diver is detected, however we exclude these images from the validation and test sets using only data from [94] that were excluded

[4] PyTorch libraries: `https://pytorch.org/`

from the training.

This model again uses pre-trained weights of FUnIE-GAN on the EUVP dataset[5] and then is further tuned with our training pipeline in Eq. 6.6. A batch size of 32 is again used in all the training modes to increase the possibility of detectable divers in each batch. The results of training this second version of the DUnIE-GAN algorithm are presented in the following section.

6.2.2.2 Validation and Testing Conclusions

Here we present evaluation results of our second implementation of DUnIE-GAN. For these results, we focus on the different models as DUnIE-GAN is trained with our detector-based loss optimization, $\mathcal{L}_{rc}$, in different locations: located in both the generator and discriminator, only generator, only discriminator, and neither location. We first present the validation results of an ablation study performed by training all models for 200 epochs. As can be seen in the plot in Fig. 6.5, the resulting IoUs do not converge during training. However, the models that include our additional loss function outperform the model that does not. See Table 6.4 for the IoU scores for the most successful epoch during training.

Table 6.4: Validation ablation results for locations of $\mathcal{L}_{rc}$: Highest mean IoU on underwater images of single divers and the epoch at which it occurs.

Model	Epoch	IoU
Original Images	–	0.478
DUnIE-GAN	100	0.513
DUnIE-GAN (No Generator Detector Loss)	190	0.499
DUnIE-GAN (No Discriminator Detector Loss)	100	0.503
DUnIE-GAN (No Detector Losses)	10	0.412

When evaluating the test set, however, we found inconsistencies in the resulting detections. The DUnIE-GAN model with the additional losses in either the generator or discriminator provided better detection results than when the detection losses were

[5] FUnIE-GAN (PyTorch) model is available at: `https://github.com/xahidbuffon/FUnIE-GAN`

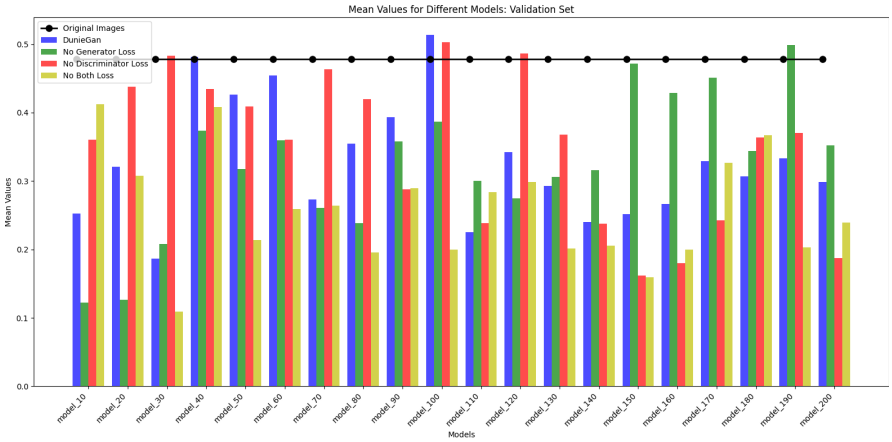


Figure 6.5: IoU results of diver detection on the validation set for all models throughout training. The x-axis represents epochs trained, y-axis represents the mean IoU of the detections by the pre-trained diver detector on generated images, and the black line along the top of the plot denotes the mean IoU of unenhanced images.

located in both the generator and discriminator. In all cases, models that included some form of detection-based loss performed better than training without the additional loss (See Table 6.5). The results of this second implementation show that there is merit to the use of detection-based losses when implementing underwater image enhancement models.

Table 6.5: Test set ablation results for locations of $\mathcal{L}_{rc}$: Mean IoU of the test set of images with single divers utilizing the training epoch determined appropriate through validation.

Model	Epoch	IoU
Original Images	–	0.469
DUnIE-GAN	100	0.433
DUnIE-GAN (No Generator Detector Loss)	190	0.479
DUnIE-GAN (No Discriminator Detector Loss)	100	0.476
DUnIE-GAN (No Detector Losses)	10	0.386

6.3 Conclusions and Discussion

In this chapter, we present an approach towards the improvement of diver detection for AUVs. The proposed model, DUnIE-GAN, though inspired by current image enhancement techniques in creating a GAN-based architecture, incorporates a detector's feedback information during the training process. Due to the modularity of the DUnIE-GAN pipeline, our model can be altered to fit a variety of detection-driven enhancement tasks, and we present results on experiments performed for pipeline development. For the diver detection task, our first evaluations demonstrate improvement when compared to existing general image enhancement models over a variety of scenarios, however, no enhancement model is shown to improve detection results over unenhanced images. In response, we investigated the results of utilizing a different pre-trained diver detector. While the most beneficial location to add the detection loss proved inconclusive, results showed improved detection over both unenhanced images and images enhanced with a model that does not include our additional loss functions. Research that we have completed with this thread has led to new interesting challenges for future work including potentially updating the current GAN architecture or using another generative network.

All of our research has centered around applications for use on board AUVs. In Chapter 7 we discuss in greater detail our contributions to the various sensor and robotic platforms used throughout this dissertation.

Chapter 7

Hardware Systems

In this chapter, we outline the various sensor and robotic systems that were designed and used in the data collection and testing stages of our work. While our work is designed to be robot agnostic, we tested our work on two main platforms. We first briefly introduce the Aqua, a pre-built AUV capable of stereo vision, then discuss our stereo data collection rig, HydroEye, and our contributions to its creation and validation. Finally, we present our contributions to the LoCO AUV, which was designed in-house in the IRVLab. All three platforms are shown in 7.1

Figure 7.1: From left to right, the hardware platforms used in our research: Aqua, HydroEye, and LoCO AUV. This image was taken during a field trial in Barbados.

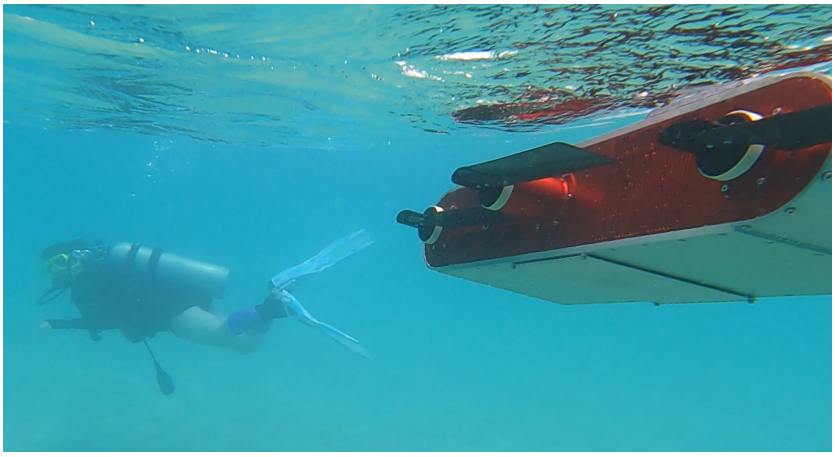


Figure 7.2: We test a diver-tracking algorithm and Aqua autonomously follows a diver in the Caribbean Sea.

7.1 Aqua

Aqua [11], developed by researchers at McGill University, is one of the autonomous underwater vehicles used for data collection as well as real-world validation. The AUV has many attributes that make it appropriate as a platform for testing diver-AUV collaborative methods; Fig. 7.2 shows the validation of an autonomous diver following algorithm. To carry out this and other such evaluations, the AUV has the following features:

- Aqua moves in five degrees of freedom (roll, pitch, yaw, surge, and heave) through the use of six flippers.
- The AUV is equipped with three cameras, including a pair of stereo-synchronized cameras on the front to allow for data collection and validation of methods requiring depth information such as DIP-3D (Chapter 4). A single camera in the rear can also be used for visual perception and U-HRI.
- Aqua has three onboard computers including a control stack to control the commands for AUV movement, a vision stack which processes sensor information such as from the cameras, and an NVIDIA Jetson Tx2 which permits for the onboard processing of deep learning models.

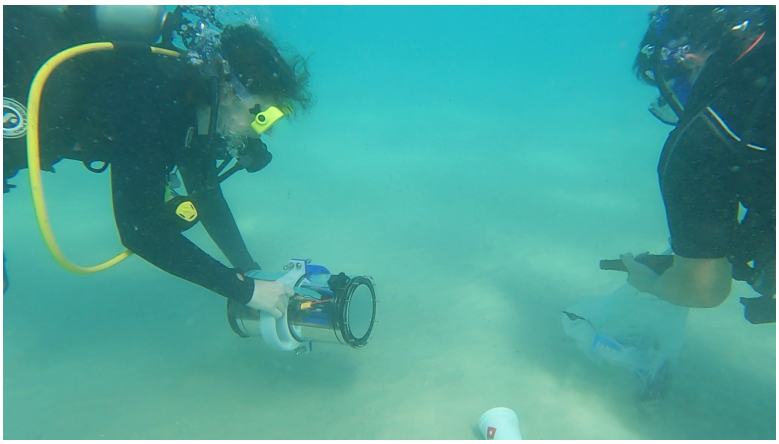


Figure 7.3: Here we validate the recording capabilities of HydroEye and capture data for other members of the IRVLab off the coast of Barbados in the Caribbean Sea.

7.2 HydroEye

HydroEye is an underwater stereo-vision data collection sensor. While data collection with an AUV is necessary for some tasks, this device was developed by members of the IRVLab to simplify the process for underwater visual data collection and reduce the need to deploy an AUV when mobility processes are not needed (See Fig. 7.3). This data collection rig is easily deployed by a single diver, runs on its own battery source, and can be deployed with or without a tether dependent on data collection needs. The overall design of the HydroEye sensor includes a watertight enclosure with a transparent acrylic front plate for clear camera vision, a stereo camera and a mobile GPU for image processing. HydroEye uses the Robot Operating System (ROS) [73], which allows us to save data in the same form that would be used by the AUV systems. Our main contribution to the HydroEye platform includes testing various stereo cameras and general software improvements. HydroEye has been used to collect for much of the research discussed in this dissertation. Although stereo data was unnecessary for DIP (Chapter 3) and SPOC (Chapter 5, data collected with HydroEye was used during the development of the algorithm and for the hand gesture dataset, respectively. We collected stereo data for the evaluation of the DIP-3D (Chapter 4) algorithm.

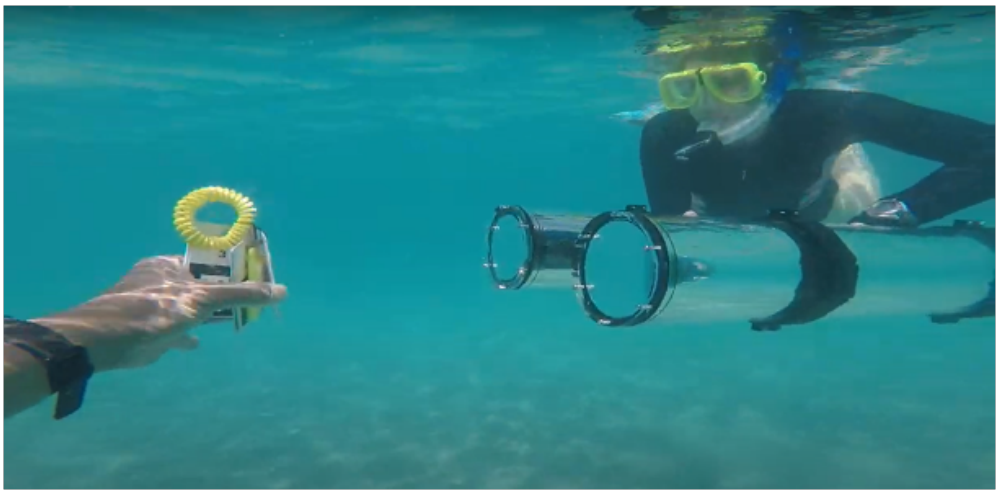


Figure 7.4: We tested the autonomous capabilities of the LoCO AUV off the coast of Barbados in the Caribbean Sea.

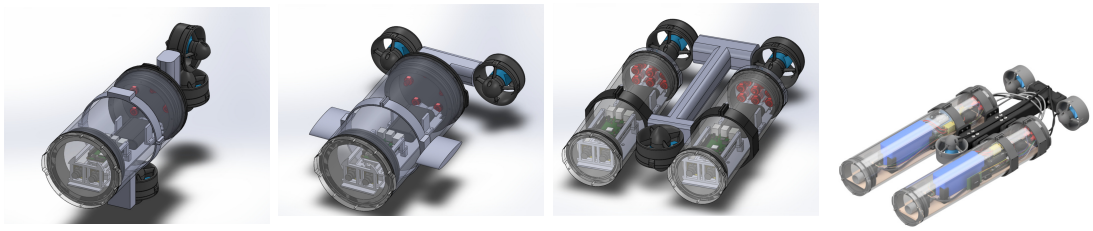
7.3 LoCO AUV

The LoCO AUV [12], designed by members of the IRVLab, originated to lower the barrier to underwater robotic research through the creation of a relatively inexpensive AUV that can be easily built and modified. The design, parts list, and building instructions have been released under open source licenses [1] (we have GPL V3 and Creative Commons V4) for any who would build the AUV. We participated not only in the design and building of the AUV, but also in testing the AUV and making improvements during field trials (See Fig. 7.4). LoCO has been the main research platform for both DIP (Chapter 3) and SPOC (Chapter 5). The work presented in this chapter highlights our specific contributions, some of which is sourced from the collaborative paper.

7.3.1 Overall System

LoCO is designed to be an all-purpose AUV, adaptable to a variety of missions. In the standard configuration, LoCO is a dual-camera, vision-guided AUV with three thrusters. One of our main contributions to LoCO is the concept of the overall design of the AUV, the two-tube "binocular" design. As such, LoCO consists of two water-tight

[1] https://loco-auv.github.io/



(a) "Helicopter" style. (b) "Submarine" style. (c) "Binocular" style. (d) Final design.

Figure 7.5: CAD renderings showing the development of LoCO AUV.

enclosures, each containing various components, with one thruster mounted between the enclosures, and two mounted behind, as seen in Fig. 7.5d. While several designs were considered (see Fig. 7.5), our two-enclosure design was selected for a variety of reasons. Firstly, it allowed for a reasonable placement of a pitch-control thruster, along with providing space in between the enclosures which provides space to mount sensors, thrusters, or manipulators in the future. Additionally, the design called for two enclosures side by side, narrower than the enclosures used for the other designs, which would reduce the robot's forward profile, allowing it to move through the water with less resistance. Finally, the separation into two enclosures enforces a base level of modularity. Most control-related electronics are in the left-hand enclosure, with the computational hardware for deep learning inference in the right-hand enclosure. While any type of hardware modification requires changes throughout the system, changes or replacements can be made with minimal impact on the layout of components internally.

7.3.2 Simulator

In addition to releasing the specifications of the AUV, a base Gazebo simulator [107] is released as well, using ROS for AUV modeling and control. Our contribution to the simulator is focused on sensor integration and includes the integration of usable camera sensors that follow the specifications of those in the LoCO AUV.

The SolidWorks CAD design model shown in Fig. 7.5d was used as the template for creating the robot's Universal Robotic Description Format (URDF) model file. To improve simulation efficiency while not affecting simulation physics, the mesh file that

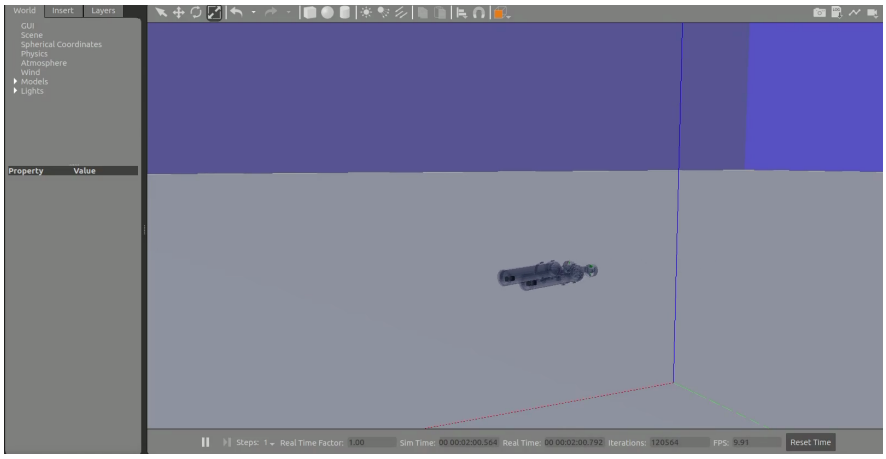


Figure 7.6: The LoCO AUV in Gazebo simulation.

provides the visual representation of the robot does not include internal components other than the front cameras. The model is shown in the Gazebo simulator in Fig. 7.6. We added camera sensors directly to the URDF model file through a Gazebo plugin (libgazebo_ros_camera.so), and these can be modified to match the specifications of the cameras used on the physical robot.

Chapter 8

Conclusion

8.1 Summary of Presented Research

The research presented in this dissertation has focused on meeting the unique challenges of communication between humans and robots in the underwater domain. We presented methods for improving visual communication for divers and Autonomous Underwater Vehicles (AUVs) for underwater task-based collaboration. Centering around natural human communication to facilitate ease of use and understanding to minimally add to a diver's cognitive load, we primarily explored the use of pointing gestures to facilitate task-completion communication and use scene information to guide the AUV. In Chapter 3 we presented Diver Interest via Pointing (DIP), the first use of a pointing gesture to indicate an object of interest to an AUV. We found that the region of interest created with the use of single-image human pose estimation and feature detection, provided sufficient information to enable the AUV to locate the object indicated by the diver. Diver interest via Pointing in Three Dimensions (Chapter 4) introduced the third dimension of distance to diver-to-AUV pointing communication. This is a necessary capability to determine the direction the diver is pointing, *i.e.*, forwards, to the side, or backwards, especially if there are multiple objects that would be found in the same region of interest in a 2D image. Design choices for the creation of both of these communication methods were made with functionality by both the diver and the AUV in mind. Both methods have also been validated through the use of AUV onboard testing. Chapter 5, Semantic Pointing for Communication (SPOC) greatly improved the utility of visual

communication through pointing gestures. SPOC provided a system that identified an object of interest, assigned the task that should be performed, and provided feedback to the diver that the information was received. The system runs fully onboard and was validated using the LoCO AUV. As vision-based methods for collaboration improve, techniques to compensate for poor visibility are needed for optimal communication in visually degraded environments, such as underwater. Enhancing images prior to their use in perception-based tasks provides one way to meet these challenges. Chapter 6 presented our findings on an investigation into the addition of a task-relevant component, in our case diver detection, to a deep learning based image enhancement method. While our investigation provided interesting results, more research into this thread would be needed prior to utilizing onboard an AUV. All work presented in this dissertation is designed to be used onboard an AUV, and consequently the development and use of robotic platforms has been an integral part of our research. In Chapter 7 we discussed the hardware systems that allow us to collect data and evaluate our Underwater HRI methods onboard an AUV. In particular, we discussed our contributions to the design and construction of HydroEye sensor and the LoCO AUV platforms.

8.2 Future Research Directions

8.2.1 Pointing Gestures to Provide Directional Cues

Without the use of satellite navigation systems underwater, such as GPS, AUVs often use dead-reckoning or surface to locate waypoints to navigate. If the AUV is working with a collaborative diver, the diver should be able to make on-the-fly decisions about where the AUV should navigate to. Current hand gesture systems 2.1 often include gestures for the AUV to move left or right, however specific directions are not given. Methods using pointing gestures to provide a direction rather than simply interact with an object are as yet unexplored.

8.2.2 On-Site Instruction for AUV Tasks

Underwater environmental characteristics can change rapidly and circumstances may be different between an exploratory dive and a planned future mission. For example, a new

species may be found in an unexpected area and the diver may want an AUV to check for other creatures in nearby locations. In this case, the object located through pointing is an exemplar of the object to be located during the mission. When combined with a system such as SPOC (Chapter 5) a single dive mission would be able to use on-site information to provide multiple object-dependent tasks to the AUV, such as collecting samples of a type of object while collecting close-up images of a different object.

8.2.3 Risk Aware Diver-AUV Communication

Many AUV tasks require interaction with their environment, such as picking up a piece of trash or collecting a sample. The target of the task is often located in close proximity to objects that should not be interacted with in the same way, such as if the trash collection mission takes place on a coral reef. Confirmation that the correct action is being taken with relation to the detected object is one potential solution, while another may be a defined library of tasks available to be performed on an object with a notification that the task is unable to be performed. A similar addition to Diver-AUV communication systems may build more trust in field experts, thus expanding the use of marine robotics into a wider variety of uses.

8.3 Concluding Remarks

The use of AUVs in fields such as biological monitoring, infrastructure inspection, and more has shown growing potential in recent years, however there is still a great need for AUV collaboration with divers to perform many tasks. As this potential for collaboration improves, methods to facilitate communication are essential. It is imperative that these methods are easily-used and trustable by non-roboticists in order to meet the full potential of AUVs to assist with dangerous and time-consuming tasks. Our research provided new methods for task-relevant communication between a diver and AUV that have been designed for ease of use and with flexibility to change and update as needed. We hope that aspects of our research contribute to a wider adoption of diver and AUV collaboration.

References

[1] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2007.

[2] Michael Fulton, Aditya Prabhu, and Junaed Sattar. HREyes: Design, development, and evaluation of a novel method for AUVs to communicate information and gaze direction. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7468–7475, 2023.

[3] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. YOLACT: Real-time instance segmentation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9156–9165, 2019.

[4] Md Jahidul Islam, Youya Xia, and Junaed Sattar. Fast underwater image enhancement for improved visual perception. *IEEE Robotics and Automation Letters*, 5(2):3227–3234, 2020.

[5] Y. R. Petillot, S. R. Reed, and J. M. Bell. Real time AUV pipeline detection and tracking using side scan sonar and multi-beam echo-sounder. In *OCEANS '02 MTS/IEEE*, volume 1, pages 217–222 vol.1, 2002.

[6] MD Modasshir, Sharmin Rahman, Oscar Youngquist, and Ioannis Rekleitis. Coral identification and counting with an autonomous underwater vehicle. In *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 524– 529, Kuala Lumpur, Malaysia, (Finalist of T. J. Tarn Best Paper in Robotics), Dec. 2018.

[7] Brian Bingham, Brendan Foley, Hanumant Singh, Richard Camilli, Katerina Delaporta, Ryan Eustice, Angelos Mallios, David Mindell, Christopher Roman, and Dimitris Sakellariou. Robotic tools for deep water archaeology: Surveying an ancient shipwreck with an autonomous underwater vehicle. *Journal of Field Robotics*, 27:702–717, 11 2010.

[8] Chelsey Edge and Junaed Sattar. Diver interest via pointing: Human-directed object inspection for AUVs. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3146–3153, 2023.

[9] Chelsey Edge, Demetrious Kutzke, Megdalia Bromhal, and Junaed Sattar. Diver interest via pointing in three dimensions: 3D pointing reconstruction for diver-auv communication. *arXiv preprint arXiv:2310.11536*, 2023.

[10] Chelsey Edge, Md Jahidul Islam, Christopher Morse, and Junaed Sattar. A generative approach for detection-driven underwater image enhancement. *arXiv preprint arXiv:2012.05990*, 2020.

[11] G. Dudek, P. Giguere, C. Prahacs, S. Saunderson, J. Sattar, L. Torres-Mendez, M. Jenkin, A. German, A. Hogue, A. Ripsman, J. Zacher, E. Milios, H. Liu, P. Zhang, M. Buehler, and C. Georgiades. AQUA: An amphibious autonomous robot. *Computer*, 40(1):46–53, Jan 2007.

[12] Chelsey Edge, Sadman Sakib Enan, Michael Fulton, Jungseok Hong, Jiawei Mo, Kimberly Barthelemy, Hunter Bashaw, Berik Kallevig, Corey Knutson, Kevin Orpen, and Junaed Sattar. Design and experiments with LoCO AUV: A low cost open-source autonomous underwater vehicle. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1761–1768, Las Vegas, Nevada, USA, October 2020.

[13] Junaed Sattar, Gregory Dudek, Olivia Chiu, Ioannis Rekleitis, Philippe Giguere, Alec Mills, Nicolas Plamondon, Chris Prahacs, Yogesh Girdhar, Meyer Nahon, et al. Enabling autonomous capabilities in underwater robotics. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3628–3634, Nice, France, 2008. IEEE.

[14] Arturo Gomez Chavez, Andrea Ranieri, Davide Chiarella, and Andreas Birk. Underwater vision-based gesture recognition: A robustness validation for safe human–robot interaction. *IEEE Robotics & Automation Magazine*, 28(3):67–78, 2021.

[15] Junaed Sattar and Gregory Dudek. On the performance of color tracking algorithms for underwater robots under varying lighting and visibility. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3550–3555, Orlando, FL, USA, 2006. IEEE.

[16] Md Jahidul Islam, Jungseok Hong, and Junaed Sattar. Person-following by autonomous robots: A categorical overview. *The International Journal of Robotics Research (IJRR)*, 38(14):1581–1618, 2019.

[17] Md Jahidul Islam and Junaed Sattar. Mixed-domain biological motion tracking for underwater human-robot interaction. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4457–4464, Singapore, 2017. IEEE.

[18] Junaed Sattar and Gregory Dudek. Underwater human-robot interaction via biological motion identification. In *Proceedings of the International Conference on Robotics: Science and Systems V, RSS*, pages 185–192, Seattle, Washington, USA, June 2009. MIT Press.

[19] Junaed Sattar and Gregory Dudek. Robust servo-control for underwater robots using banks of visual filters. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3583–3588, Kobe, Japan, 2009. IEEE.

[20] Andreas Birk. A survey of underwater human-robot interaction (U-HRI). *Current Robotics Reports*, 3(4):199–211, Dec 2022.

[21] Gregory Dudek, Junaed Sattar, and Anqi Xu. A visual language for robot control and programming: A human-interface study. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 2507–2513, 2007.

[22] Bart Verzijlenberg and Michael Jenkin. Swimming with robots: Human robot communication at depth. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4023–4028, 2010.

[23] Dula Nad, Fausto Ferreira, Igor Kvasic, Luka Mandic, Christopher Walker, Derek Orbaugh Antillon, and Iain Anderson. Diver-robot communication using wearable sensing diver glove. In *OCEANS 2021: San Diego – Porto*, pages 1–6, 2021.

[24] Md Jahidul Islam, Marc Ho, and Junaed Sattar. Dynamic reconfiguration of mission parameters in underwater human-robot collaboration. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6212–6219, 2018.

[25] Md Jahidul Islam, Marc Ho, and Junaed Sattar. Understanding human motion and gestures for underwater human-robot collaboration. *Journal of Field Robotics (JFR)*, pages 1–23, 2018.

[26] Davide Chiarella, Marco Bibuli, Gabriele Bruzzone, Massimo Caccia, Andrea Ranieri, Enrica Zereik, Lucia Marconi, and Paola Cutugno. A novel gesture-based language for underwater human–robot interaction. *Journal of Marine Science and Engineering*, 6(3), 2018.

[27] Andrea Maree Walker. Towards natural underwater human-robot interaction: Pointing gesture recognition for autonomous underwater vehicles. Master's thesis, University of Minnesota, 2021.

[28] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017.

[30] Michal Tölgyessy, Martin Dekan, František Duchoň, Jozef Rodina, Peter Hubinský, and L'uboš Chovanec. Foundations of visual linear human–robot interaction via pointing gesture navigation. *International Journal of Social Robotics*, 9(4):509–523, Sep 2017.

[31] Jan Richarz, Andrea Scheidig, Christian Martin, Steffen Müller, and Horst-Michael Gross. A monocular pointing pose estimator for gestural instruction of a mobile robot. *International Journal of Advanced Robotic Systems*, 4(1):17, 2007, https://doi.org/10.5772/5700.

[32] Shaukat Abidi, MaryAnne Williams, and Benjamin Johnston. Human pointing as a robot directive. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 67–68, 2013.

[33] Anna C. S. Medeiros, Photchara Ratsamee, Jason Orlosky, Yuki Uranishi, Manabu Higashida, and Haruo Takemura. 3D pointing gestures as target selection tools: Guiding monocular UAVs during window selection in an outdoor environment. *ROBOMECH Journal*, 8(1), April 2021.

[34] Yucheng Chen, Yingli Tian, and Mingyi He. Monocular human pose estimation: A survey of deep learning-based methods. *Computer Vision and Image Understanding*, 192:102897, 2020.

[35] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2D human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[36] Nikolaos Sarafianos, Bogdan Boteanu, Bogdan Ionescu, and Ioannis A. Kakadiaris. 3D human pose estimation: A review of the literature and analysis of covariates. *Computer Vision and Image Understanding*, 152:1–20, 2016.

[37] Arturo Gomez Chavez, Christian A. Mueller, Andreas Birk, Anja Babic, and Nikola Miskovic. Stereo-vision based diver pose estimation using LSTM recurrent neural networks for AUV navigation guidance. In *OCEANS 2017 - Aberdeen*, pages 1–7, 2017.

[38] Md Jahidul Islam, Jiawei Mo, and Junaed Sattar. Robot-to-robot relative pose estimation using humans as markers. *Autonomous Robots*, 45(4):579–593, May 2021.

[39] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1):172–186, 2021.

[40] Michael Fulton, Jungseok Hong, and Junaed Sattar. Using monocular vision and human body priors for AUVs to autonomously approach divers. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 1076–1082, 2022.

[41] TRT Pose. `https://github.com/NVIDIA-AI-IOT/trt_pose`. 2020.

[42] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. Mediapipe: A framework for perceiving and processing reality. In *Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR) 2019*, 2019.

[43] Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, and Matthias Grundmann. Blazepose: On-device real-time body pose tracking. *CoRR*, abs/2006.10204, 2020, 2006.10204.

[44] Dadhichi Shukla, Ozgur Erkent, and Justus Piater. Probabilistic detection of pointing directions for human-robot interaction. In *2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, page 1–8, Nov 2015.

[45] E. Littmann, A. Drees, and H. Ritter. Robot guidance by human pointing gestures. In *Proceedings of International Workshop on Neural Networks for Identification, Control, Robotics and Signal/Image Processing*, page 449–457, Aug 1996.

[46] Bjarne Großmann, Mikkel Rath Pedersen, Juris Klonovs, Dennis Herzog, Lazaros Nalpantidis, and Volker" Krüger. Communicating unknown objects to robots through pointing gestures. In Michael Mistry, Aleš Leonardis, Mark Witkowski, and Chris Melhuish, editors, *Advances in Autonomous Robotics Systems*, pages 209–220, Cham, 2014. Springer International Publishing.

[47] Jeffrey Delmerico, Stefano Mintchev, Alessandro Giusti, Boris Gromov, Kamilo Melo, Tomislav Horvat, Cesar Cadena, Marco Hutter, Auke Ijspeert, Dario Floreano, Luca M. Gambardella, Roland Siegwart, and Davide Scaramuzza. The Current State and Future Outlook of Rescue Robotics. *Journal of Field Robotics*, 36(7):1171–1191, 2019, https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.21887.

[48] David Droeschel, Jörg Stückler, and Sven Behnke. Learning to interpret pointing gestures with a time-of-flight camera. In *2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 481–488, 2011.

[49] Bita Azari, Angelica Lim, and Richard Vaughan. Commodifying pointing in HRI: Simple and fast pointing gesture detection from RGB-D images. In *2019 16th Conference on Computer and Robot Vision (CRV)*, pages 174–180, 2019.

[50] César Silva, Ricardo Aires, and Flávio Rodrigues. A compact underwater stereo vision system for measuring fish. *Aquaculture and Fisheries*, 2023.

[51] Arturo Gomez Chavez, Andrea Ranieri, Davide Chiarella, Enrica Zereik, Anja Babić, and Andreas Birk. CADDY underwater stereo-vision dataset for human–robot interaction (HRI) in the context of diver activities. *Journal of Marine Science and Engineering*, 7(1), 2019.

[52] M. Bryson, M. Johnson-Roberson, O. Pizarro, and S. B. Williams. True color correction of autonomous underwater vehicle imagery. *Journal of Field Robotics (JFR)*, 33(6):853–874, 2016.

[53] D. Akkaynak and T. Treibitz. A revised underwater image formation model. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6723–6732, Salt Lake City, UT, USA, 2018. IEEE.

[54] D. Berman, D. Levy, S. Avidan, and T. Treibitz. Underwater single image color restoration using haze-lines and a new quantitative dataset. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[55] K. He, J. Sun, and X. Tang. Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12):2341–2353, 2010.

[56] Javier Perez, Pedro J Sanz, Mitch Bryson, and Stefan B Williams. A benchmarking study on single image dehazing techniques for underwater autonomous vehicles. In *OCEANS 2017-Aberdeen*, pages 1–9, Aberdeen, UK, 2017. IEEE.

[57] Md Jahidul Islam, Peigen Luo, and Junaed Sattar. Simultaneous enhancement and super-resolution of underwater imagery for improved visual perception. In *Proceedings of Robotics: Science and Systems*, Corvalis, Oregon, USA, July 2020.

[58] C. Li, C. Guo, W. Ren, R. Cong, J. Hou, S. Kwong, and D. Tao. An underwater image enhancement benchmark dataset and beyond. *IEEE Transactions on Image Processing*, 29:4376–4389, 2020.

[59] P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, Honolulu, HI, USA, 2017. IEEE.

[60] Cameron Fabbri, Md Jahidul Islam, and Junaed Sattar. Enhancing underwater imagery using generative adversarial networks. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 7159–7165, Brisbane, Queensland, Australia, May 2018. IEEE.

[61] Hanyu Li, Jingjing Li, and Wei Wang. A fusion adversarial underwater image enhancement network with a public test dataset. *arXiv preprint arXiv:1906.06819*, 2019.

[62] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 2642?2651, Sydney, NSW, Australia, 2017. JMLR.org.

[63] Yongqiang Zhang, Yancheng Bai, Mingli Ding, and Bernard Ghanem. Multi-task Generative Adversarial Network for Detecting Small Objects in the Wild. *International Journal of Computer Vision*, 128(6):1810–1828, June 2020.

[64] Jianan Li, Xiaodan Liang, Yunchao Wei, Tingfa Xu, Jiashi Feng, and Shuicheng Yan. Perceptual generative adversarial networks for small object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1951–1959, Honolulu, HI, July 2017. IEEE.

[65] Lanlan Liu, Michael Muelly, Jia Deng, Tomas Pfister, and Li-Jia Li. Generative modeling for small-data object detection. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6072–6080, Seoul, Korea (South), October 2019. IEEE.

[66] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, Venice, Italy, 2017. IEEE.

[67] Chelsey Edge and Junaed Sattar. Diver interest via pointing: Human-directed object inspection for auvs. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3146–3153, 2023.

[68] Junaed Sattar, Eric Bourque, Philippe Giguère, and Gregory Dudek. Fourier tags: Smoothly degradable fiducial markers for use in human-robot interaction. In *Proceedings of the Fourth Canadian Conference on Computer and Robot Vision*, pages 165–174, Montréal, QC, Canada, 5 2007.

[69] Md Jahidul Islam, Ruobing Wang, and Junaed Sattar. SVAM: Saliency-guided visual attention modeling by autonomous underwater robot. In *Proceedings of Robotics: Science and Systems*, New York City, NY, USA, June 2022.

[70] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision(IJCV)*, 60(2):91–110, 2004.

[71] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: an efficient alternative to SIFT or SURF. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2564–2571. IEEE, 2011.

[72] John Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986.

[73] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. ROS: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.

[74] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[75] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[76] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Inc., 2008.

[77] Shortis, Mark. *Camera Calibration Techniques for Accurate Measurement Underwater*, pages 11–27. Springer International Publishing, Cham, 2019.

[78] Jean-Marc Lavest, Gérard Rives, and Jean-Thierry Lapresté. Underwater camera calibration. In *Computer Vision—ECCV 2000: 6th European Conference on Computer Vision Dublin, Ireland, June 26–July 1, 2000 Proceedings, Part II 6*, pages 654–668. Springer, 2000.

[79] Preeti Pidatala. *Evaluation of Deep Object Detectors for Pointing Gesture Classification for Underwater Human-Robot Communication*. Honor's thesis, University of Minnesota, 2023.

[80] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 2018.

[81] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, pages 234–241. Springer, 2015.

[82] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890, 2017.

[83] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *ArXiv preprint arXiv:1706.05587*, 2017.

[84] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Doll'a r, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014, 1405.0312.

[85] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017.

[86] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLO 8.0.0. `https://github.com/ultralytics/ultralytics`, January 2023.

[87] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

[88] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[89] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[90] Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[91] Junaed Sattar and Gregory Dudek. Underwater human-robot interaction via biological motion identification. In *Proceedings of the International Conference on Robotics: Science and Systems V, RSS*, pages 185–192, Seattle, Washington, USA, June 2009. MIT Press.

[92] L.R. Rabiner et al. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[93] Michael Fulton, Karin de Langis, and Junaed Sattar. *The VDD-C Video Diver Detection Dataset*. University of Minnesota Twin Cities, 2020. The Interactive Robotics and Vision Laboratory. `http://irvlab.cs.umn.edu/vddc`. Accessed 10-31-2020.

[94] Md Jahidul Islam, Michael Fulton, and Junaed Sattar. Toward a generic diver-following algorithm: Balancing robustness and efficiency in deep visual detection. *IEEE Robotics and Automation Letters*, 4(1):113–120, 2019.

[95] 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 *OCEANS 2015 - Genova*, pages 1–7, Genova, Italy, 2015. IEEE.

[96] H. Buelow and A. Birk. Diver detection by motion-segmentation and shape-analysis from a moving vehicle. In *OCEANS'11 MTS/IEEE KONA*, pages 1–7, Waikoloa, HI, USA, 2011. IEEE.

[97] Fenglei Han, Jingzheng Yao, Haitao Zhu, and Chunhui Wang. Underwater image processing and object detection based on deep cnn method. *J. Sensors*, 2020:6707328:1–6707328:20, 2020.

[98] Wenqing Huang, Mingzhu Huang, and Yuting Zhang. Detection of traffic signs based on combination of GAN and faster-rcnn. *Journal of Physics: Conference Series*, 1069:1–8, August 2018.

[99] Charan D. Prakash and Lina J. Karam. It GAN DO Better: GAN-based detection of objects on images with varying quality. *arXiv:1912.01707 [cs]*, December 2019. arXiv: 1912.01707.

[100] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014, 1411.1784.

[101] R. Girshick. Fast R-CNN. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, Santiago, Chile, 2015. IEEE.

[102] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[103] R. Padilla, S. L. Netto, and E. A. B. da Silva. A survey on performance metrics for object-detection algorithms. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 237–242, Niteroi, Brazil, 2020. IEEE.

[104] Shu Zhang, Ting Wang, Junyu Dong, and Hui Yu. Underwater image enhancement via extended multi-scale retinex. *Neurocomputing*, 245:1–9, 2017.

[105] K. Panetta, C. Gao, and S. Agaian. Human-visual-system-inspired underwater image quality measures. *IEEE Journal of Oceanic Engineering*, 41(3):541–551, 2016.

[106] A. Horé and D. Ziou. Image quality metrics: PSNR vs. SSIM. In *2010 20th International Conference on Pattern Recognition*, pages 2366–2369, Istanbul, Turkey, 2010. IEEE.

[107] N. Koenig and A. Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 3, pages 2149–2154, Sep. 2004.

[108] Fulton, Michael and Edge, Chelsey and Sattar, Junaed. Robot communication via motion: Closing the underwater human-robot interaction loop. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4660–4666, 2019.

[109] Michael Fulton, Chelsey Edge, and Junaed Sattar. Robot communication via motion: A study on modalities for robot-to-human communication in the field. *J. Hum.-Robot Interact.*, 11(2), feb 2022.

[110] Islam, Md Jahidul and Edge, Chelsey and Xiao, Yuyang and Luo, Peigen and Mehtaz, Muntaqim and Morse, Christopher and Enan, Sadman Sakib and Sattar, Junaed. Semantic segmentation of underwater imagery: Dataset and benchmark. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1769–1776, 2020.

Appendix A

Additional Contributions

In addition to the previous work detailed in the main body of the proposal, we had the great opportunity to support fellow lab members on the following research threads:

A.1 Robot Communication via Motion

This thread [108, 109] presents the use of *kinemes* or language through movement for use with robot-to-human communication by non-humanoid robots. While human-to-AUV communication has been explored through many methods, such as those described in 2.1, communication from an AUV to a diver has been less studied. Devices such as lights, speakers, or viewing screens could be added to an AUV, however our intent with this project was to design a method easily comprehensible to divers that a field robot could implement without additional sensors or equipment. Kinemes use the motion of the robot itself to pass along information, such as moving in a manner that is similar

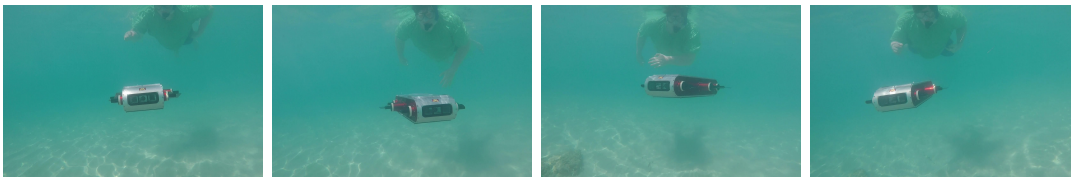
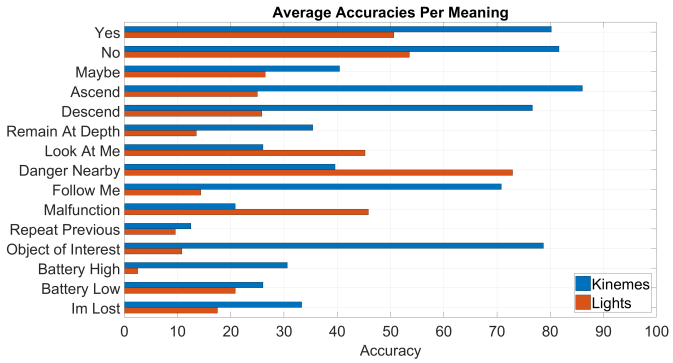
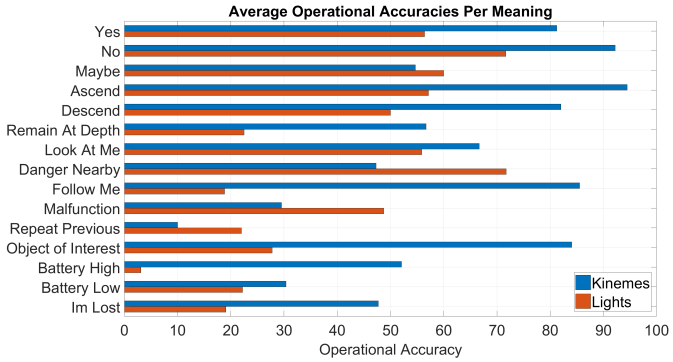


Figure A.1: The Aqua AUV in the Caribbean, indicating a "No" by shaking its "head" back and forth.



(a) Average Accuracy per meaning.



(b) Average Operational Accuracy per meaning.

Figure A.2: Average Accuracy and Operational Accuracy per meaning.

to a head nodding for "yes" or shaking for "no" (See Fig. A.1 for an example kineme onboard the Aqua AUV).

For this project I was involved in the creation of the original library of terms and their movements. As this work heavily relies on human understanding of the movements, a human survey was used, part of our work with this project was to oversee in-person surveys. The list of kinemes created along with resulting accuracy of the kinemes and a baseline light setup can be found in Fig. A.2. Fig. A.2b shows the operational accuracy of both lights and methods. Operational accuracy is determined to be when the study participant would choose to act on communication from the AUV based on how confident they are in understanding the phrase implemented through either the lights or the kineme.

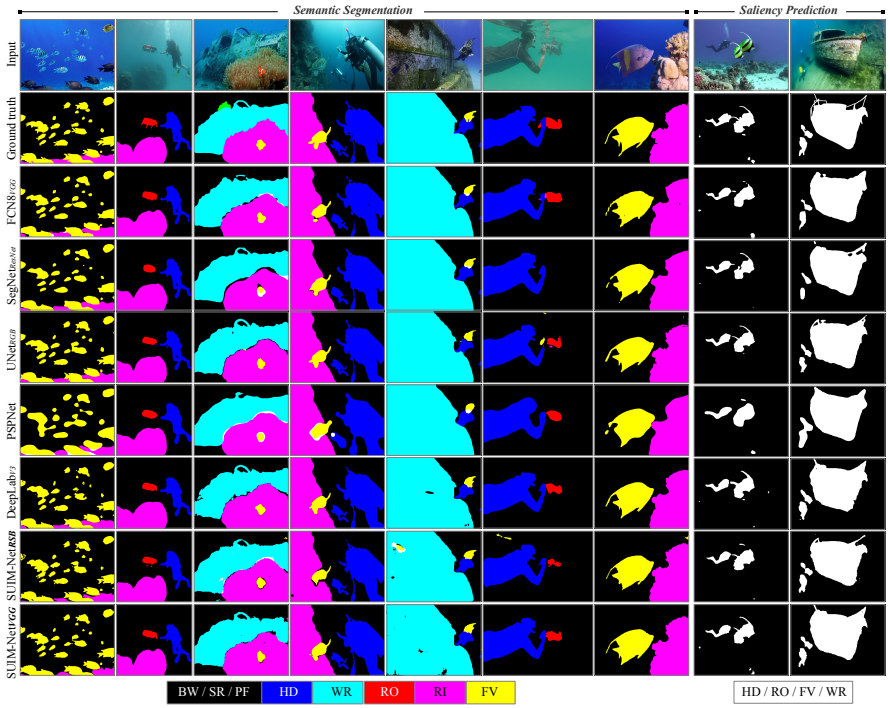


Figure A.3: A few qualitative comparisons for the benchmarking networks: (left) semantic segmentation with `HD`, `WR`, `RO`, `RI`, and `FV` as object categories; (right) saliency prediction with `HD=RO=FV=WR=1` and `RI=PF=SR=BW=0`. Results for the top performing models are shown; best viewed digitally by zoom for details.

A.2 Underwater Image Segmentation

This thread [110] resulted in the first large-scale dataset of underwater images semantically annotated with pixel annotations of eight categories: fish or vertebrates (FV), reefs or invertebrates (RI), aquatic plants or sea-grass (PF), wrecks or ruins (WR), human divers (HD), robots (RO), sea-floor or rocks (SR), and background waterbody (BW). In addition, a benchmark comparison of 8 state-of-the-art semantic segmentation and saliency prediction networks along with SUIM-Net, a novel encoder-decoder model was performed. I participated in this project by assisting in the compilation, training and evaluation of the benchmarking networks including FCN8, UNet, and DeepLabv3. Fig. A.3 shows qualitative results of the benchmark experiments.