

## Research Article

# Design and Implementation of an Assistive Real-Time Red Lionfish Detection System for AUV/ROVs

**M-Mahdi Naddaf-Sh , Harley Myler , and Hassan Zargarzadeh **

Phillip M. Drayer Electrical Engineering Department, Lamar University, Beaumont, TX, USA

Correspondence should be addressed to Hassan Zargarzadeh; [hzargarzadeh@lamar.edu](mailto:hzargarzadeh@lamar.edu)

Received 11 September 2018; Revised 19 October 2018; Accepted 1 November 2018; Published 11 November 2018

Guest Editor: Li Zhang

Copyright © 2018 M-Mahdi Naddaf-Sh et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent years, the Pterois Volitans, also known as the red lionfish, has become a serious threat by rapidly invading US coastal waters. Being a fierce predator, having no natural predator, being adaptive to different habitats, and being with high reproduction rates, the red lionfish has enervated current endeavors to control their population. This paper focuses on the first steps to reinforce these efforts by employing autonomous vehicles. To that end, an assistive underwater robotic scheme is designed to aid spear-hunting divers to locate and more efficiently hunt the lionfish. A small-sized, open source ROV with an integrated camera is programmed using Deep Learning methods to detect red lionfish in real time. Dives are restricted to a certain depth range, time, and air supply. The ROV program is designed to allow the divers to locate the red lionfish before each dive, so that they can plan their hunt to maximize their catch. Lightweight, portability, user-friendly interface, energy efficiency, and low cost of maintenance are some advantages of the proposed scheme. The developed system's performance is examined in areas currently invaded by the red lionfish in the Gulf of Mexico. The ROV has shown success in detecting the red lionfish with high confidence in real time.

## 1. Introduction

Biological invasions can cause environmental disruption and biodiversity loss, often due to human-caused global change [1–3]. Invasive species are nonnative species that may have serious effects on ecosystems and habitats. Some of these effects can evolve to global consequences [4]. The terrestrial and freshwater systems appropriate the majority of invasions, while, in the last decade, the rates of the marine invasions have dramatically increased and impacted the stability of ecosystems, raising ecological and economic concerns [1, 5]. Overall, recent studies [6] indicate that invasive species cost 120 billion dollars to the environment and the economy. Among marine species, *Pterois volitans* (red lionfish) is the most invasive and aggressive species that has taken only two decades to populate across a significant portion of the US east coast [7–9].

Regardless of how the red lionfish were first introduced [10–12], their rapid reproduction rate, lack of significant predators, and wide range of dietary consumption have made them a serious threat to coral reefs and many other marine

environments [13–16]. As evidence to this threat, in Figure 1, the spread of the red lionfish in 1995 is compared with that of 2015. The red lionfish grow rapidly and reach up to 50 centimeters in 3 years [16]. Both smaller fishes and crustaceans are potential prey for red lionfish [1, 9]; they literally consume anything that they can fit in their mouth. Due to their venomous dorsal, pelvic, and anal spines the red lionfish are fatal [13] to human divers [9] and native predatory species who quickly learn to avoid them. Having no natural predator on one hand and the ability to quickly procreate, which is approximately 2 million eggs per female annually, the red lionfish population is exponentially growing and calling for immediate national resolutions [17].

Being aware of this destructive invasion in recent years, scientists have tried to find effective ways to control the red lionfish to spread and prevent more damage to the ecosystem [17]. One way for controlling the red lionfish population is hunting them by scuba divers equipped with “ZooKeepers” [18] and an “ELF lionfish Spear Tool” [19]. The divers spear the red lionfish and then keep them into the tubular ZooKeeper containments through their one-way

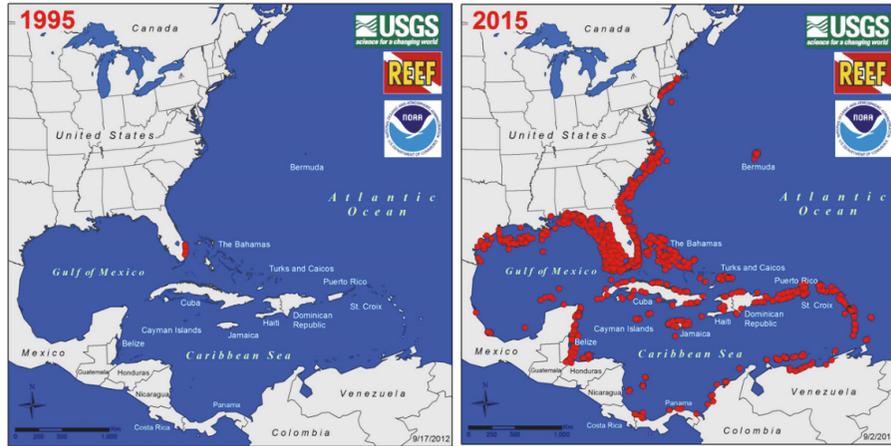


FIGURE 1: Red lionfish occurrence in the Western North Atlantic and Caribbean Sea (USGS-NAS 2015) (figures courtesy of USGS-NAS database found at <http://nas.er.usgs.gov>).

gate that holds the fish until the diver returns to the surface. This method is not cost-efficient due to limitations in the number of divers, diving time and the small number of hunts possible in each dive. Divers face many limitations for finding and locating red lionfish underwater due to significant depth, limited air cylinder capacity, low visibility conditions underwater, temperature differences, high pressure, etc. The fast spreading of the red lionfish calls for more efficient and aggressive solutions for the problem [20, 21]. Hence, introducing assistive technological schemes to detect the red lionfish can help to increase the effectiveness of the hunt in each dive.

Recently, several methods to detect fishes underwater are used for different purposes such as fishing, biological research, etc. Some of these methods employ high-resolution sonar scanning or vision-based methods. However, sonar-based fish finders are unable to distinguish fish species. Finding a specific species of fish can be very challenging using the available technologies. Using robots instead of humans in harsh environments is a common solution for such problems [22]. Nonetheless, for the robots, there are challenges to automatically detect objects underwater. Low lighting, moving cameras along with moving objects, limited sight, and background color change due to organic and artificial floating debris are some of the technicalities that add more complexity to red lionfish detection. Although some of these challenges had been addressed in the literature [23], applying them in underwater condition is still challenging. There exists an extensive amount of research for offline detection of fish underwater for different purposes such as counting [24, 25] and measurement of their length [26]. In [25] they perform detection, tracking, and counting fish; the method they used for detection was a moving average algorithm applied offline to recorded videos. In [27] a radio-tag system was developed for monitoring invasive fish. Lin Wu et al. [28] developed underwater object detection based on a gravity gradient, which detects objects underwater using a gravimeter. In [29] several methods were proposed

and implemented on offline videos for the purpose of fish classification. In [30] deep Convolutional Neural Networks were used for coral classification. Qin proposed used Deep Learning for underwater imagery analysis [31]. In [32] a survey is conducted over using Deep Learning for various marine objects excluding red lionfish. Moreover, Siddiqui et al. [33] investigated the automated classification systems that have the ability to identify fish from underwater videos and also studied the feasibility and cost efficiency of automated methods like Deep Learning. Although many species are included in [33], unlike the red lionfish, the selected species had a less complex body shape such as *P. porosus* and *A. bengalensis*. An automatic image-based system was proposed in [24] that was used for estimating the mass of free-swimming fish. Qin et al. [34] has proposed live fish recognition using deep architecture where both support vector machines and SoftMax for classification are compared. Similar to [33], the outcome of [34] was tested on fishes with simple body shapes, e.g., oval or semicircle shapes and species with variable body shapes were excluded. The challenge in the lionfish detection is that it takes different gestures in different dispositions such as offending, defending, hiding or normal swimming [1]. In other words, unlike salmon, trout, tuna, etc., the red lionfish does not have a certain body shape. To the best of the author's knowledge, the available underwater object detection methods have not been applied to the detection of red lionfish in real time.

In this work, a compact-sized, open source ROV is employed to assist divers in detecting red lionfish in a more efficient manner. The scheme of the mechanism is as follows. The ROV is tethered to a computer on the surface that receives the video from a camera integrated on the ROV. The computer processes the video frames real-in-time and detects the red lionfish. By prespotting the red lionfish, the divers will not need to consume time and air hunting for them.

The ability of Deep Learning to learn patterns from high-dimensional data especially in image processing problems has made it a major tool for object detection or classification

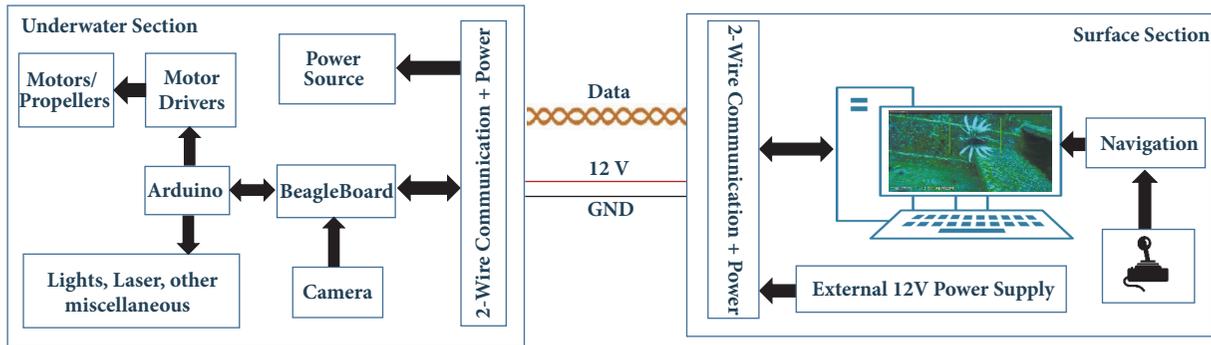


FIGURE 2: Overall system block diagram. ROV, Designed Interface in MATLAB, and Joystick for navigation are main parts of assistive system.

in recent years in several applications [35]. The proposed real-time red lionfish detection scheme employs the Deep Learning method in a MATLAB-based Graphical User Interface (GUI) in a PC. The user employs a joystick to navigate and investigate underwater while the ROV is programmed to send video from its camera in real time. Each frame of the video is processed, and the detected red lionfish are identified on the screen. The ROV uses its Inertial Measurement Unit (IMU) to report the detected lionfish's location to the surface. This scheme offers a unique platform that can be employed for detecting any other recognizable species. Further, finding, locating, and recording the population of the red lionfish are useful for the biologists to determine the red lionfish spreading patterns [7].

The remainder of this paper is organized as follows. Section 2 gives an overview of the robot specifications and navigation; also it discusses the algorithm and methodology that has been used for detecting the red lionfish. In Section 3, Simulation and real-world testing results and challenges are presented. The conclusion follows in Section 4.

## 2. Design

The proposed assistive system is comprised of the following subsystems:

- (1) OpenROV robot that consists of actuators, control boards, lighting system, camera, and navigation system
- (2) Computer and Graphical User Interface
- (3) Red Lionfish detection system

The overall block diagram of the designed system is depicted in Figure 2, and the above subsystems are elaborated in the following subsections.

**2.1. OpenROV 2.8.** The OpenROV 2.8 is a telerobotic submarine with open access to its operational source code. Small dimensions and light weight ( $30 \times 20 \times 15$  cm and 2.6 kg) make it possible to be carried and operated by a single user. It reaches a maximum forward speed of 2 knots, which is sufficient for the calm sea states, where the dives take place. This ROV is a tethered robot and all data are transceived

through a 90-meter two-wire twisted cable with 100Mbps throughput. A Tenda HomePlug and a third party board (Topside Interface Board) designed by OpenROV [36] is used to convert Ethernet to the two-wire connection protocol. The communication channel is depicted in Figure 3. Three 700Kv (rpm/v) brushless motors with electronic speed control (ESC) propel the ROV to move in three dimensions. Figures 4 and 5 show the OpenROV layouts.

The OpenROV 2.8 is powered with six 3.3V batteries that last for at least 30 minutes of surfing, when fully charged. All the electronics are in an acrylic case under vacuum mounted as depicted in Figure 4, on an internal chassis. In addition, the maximum frame rate of the mounted camera is 30 fps at HD 720p quality.

**2.2. User Interface Panel (UIP).** To provide data access and control, a MATLAB-based Graphical User Interface (GUI) was coded that is shown in Figure 6. For all the test and simulations, the program was run on a Core i7 PC with 16 GigaByte of RAM and NVIDIA GTX 745 GPU.

Being open source, the ROV is supported by numerous libraries for accessing motors, camera, etc. They are accessible via socket.io [37] in MATLAB's GUI. A USB joystick was utilized to make the navigation of the ROV more convenient. The joystick is capable of controlling the ROV in all directions by adjusting the thrusters' rotation speed. The horizontal thrusters propel the ROV forward and backward and provide torque to control the yaw. The vertical thruster propels the ROV vertically. In addition, the joystick can control lights, camera recording, and camera tilting upward and downward. A Node.js [38] server was created and ran on BeagleBone Black that is integrated on the ROV.

In Figure 7 the communication process during deployment of the ROV underwater is depicted. Through these interface options, e.g., navigating, monitoring motors, controlling LEDs and lasers, camera tilting and capturing videos or images are accessible. The default method for transmitting data between the ROV and the surface computer is web-based. In this approach, all controlling and feedback values between the ROV and PC are transmitted using Socket.io. In other words, the ROV acts as a server and the browser in the PC is its client. To be able to transmit data without using a web browser, Java code is written for communicating

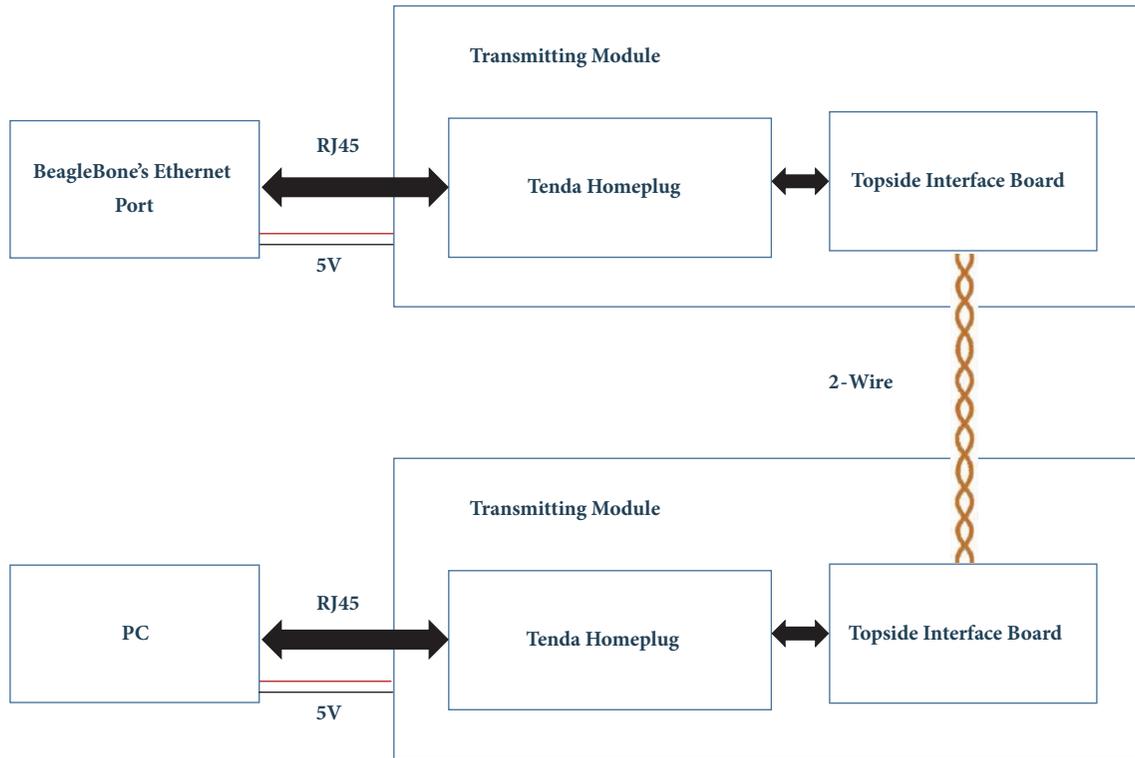


FIGURE 3: Schematic of data transmission channel between ROV and PC.

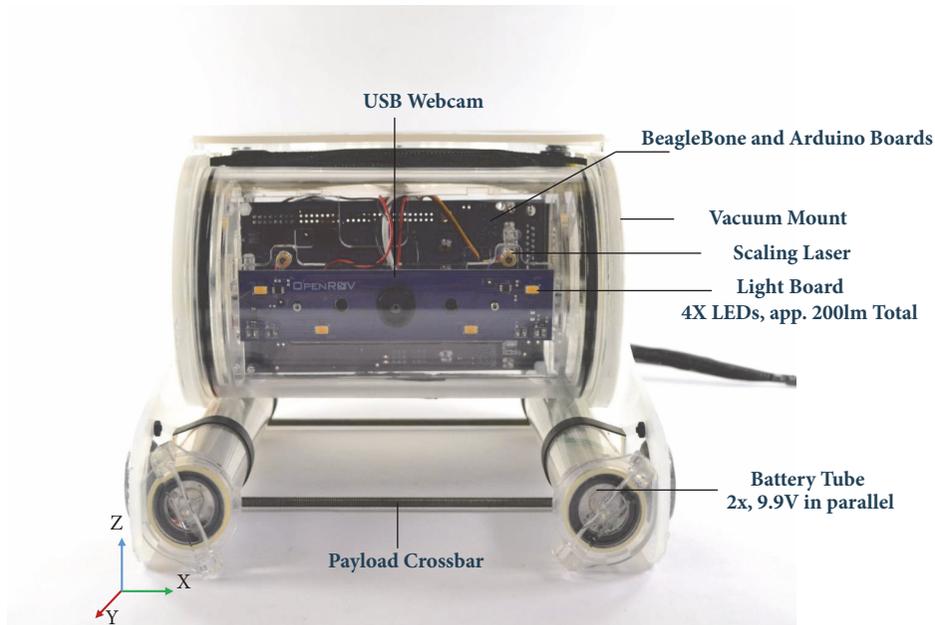


FIGURE 4: Front view of OpenROV 2.8. All main components of the ROV are marked. Two battery holders are for supplying power and balancing underwater.

with Node.js server inside the ROV directly and it bypasses the web browser for reducing delay between the ROV and UIP. This Java code is then used in the MATLAB GUI and provides complete access to the ROV. Navigational and miscellaneous related commands are sent through Socket.io

each 20 milliseconds and the camera image frames are received each 0.8 seconds.

2.3. *Object Detection Algorithm.* To detect objects underwater, the Deep Learning (DL) method is employed. DL

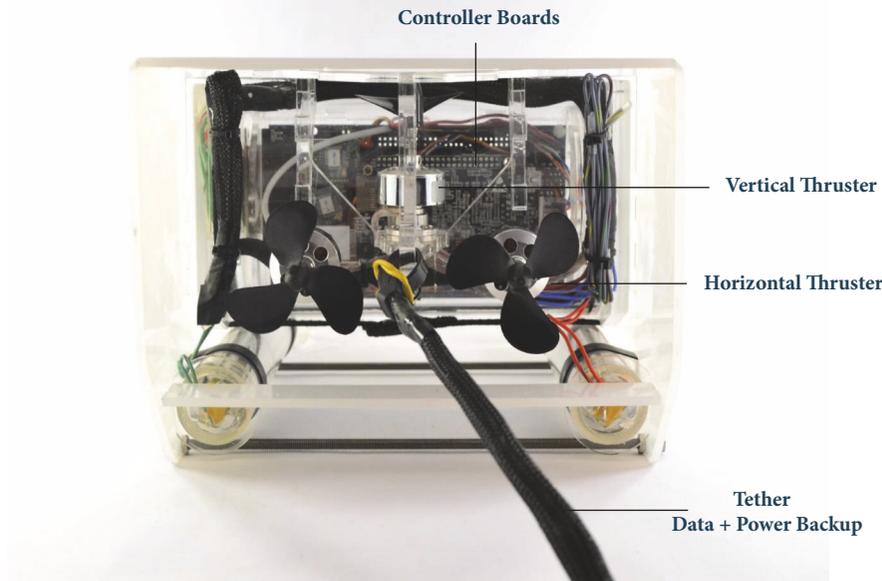


FIGURE 5: Rear view of OpenROV 2.8. There are three brushless motors for moving the device underwater. The tether cable is used for data transfer.

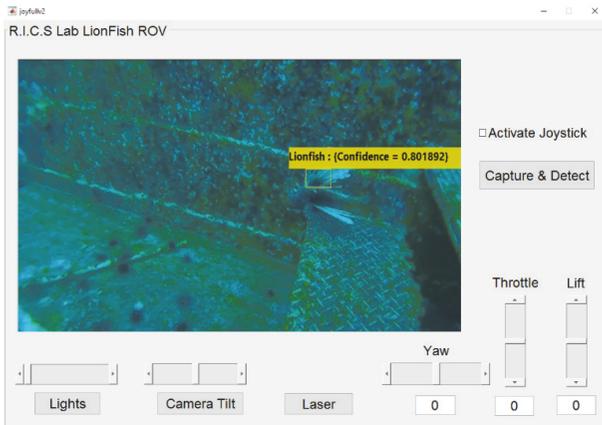


FIGURE 6: Designed UIP in MATLAB. All three brushless motors, LED lights, camera, laser, and camera tilt are accessible via this UIP. Live camera view also is available on this UIP.

utilizes Convolutional Neural Networks for object detection and classification [39]. For detecting objects of interest, it is necessary to gather a database that includes prototypical images of those specific objects. The more images that contain the object of interest, the more accurate the detection will become. There is no limitation on the number of objects of interest to be detected other than memory if sufficient prototypes are available. There are several databases for various objects available online, e.g., human faces in different poses, human body gestures, cars, etc. But unfortunately, because the red lionfish are a specific species, there are no images or video database available for them. To address that

issue, 1500 images were gathered from different royalty free online resources such as ImageNet, Google, and YouTube. Also, the authors contributed to the database by participating in diving excursions in the Gulf of Mexico in different infested areas such as Flower Garden Banks National Marine Sanctuary, artificial reefs off the coast of Pensacola, FL.

Deep Learning consists of many cascaded layers and these layers are nonlinear processing functions used for feature extraction and transformation. The pattern recognition for the database is semisupervised and because of that we introduce the object of interest, which is a red lionfish in this case, and label them in the database. The classification is an unsupervised algorithm that classifies objects of interest from other objects based on defined labels. The database is trained using Regions with Convolutional Neural Networks (R-CNN) [40]. In this project the database images were run through 15 layers of  $5 \times 5$  convolutions, and the filters were trained using Stochastic Gradient Descent with Momentum (SDGM) [41].

MATLAB was used to label the images. Using the “Training Image Labeler” app in MATLAB allows us to specify all rectangular ROIs in images. Most of images suffered from two problems, the first one was background and the second was low quality images requiring preprocessing. For instance, Figure 8(a) depicts a red lionfish next to an artificial reef, so for preparing the image for the database the brightness of image is modified as shown in Figure 8(b). Also, the size of the images is reduced to avoid other unnecessary objects, although the objects in the background or foreground, such as reefs and underwater debris, are impossible to avoid.

CNNs have the ability to build their own features and transform the input signal using convolutional kernels, an

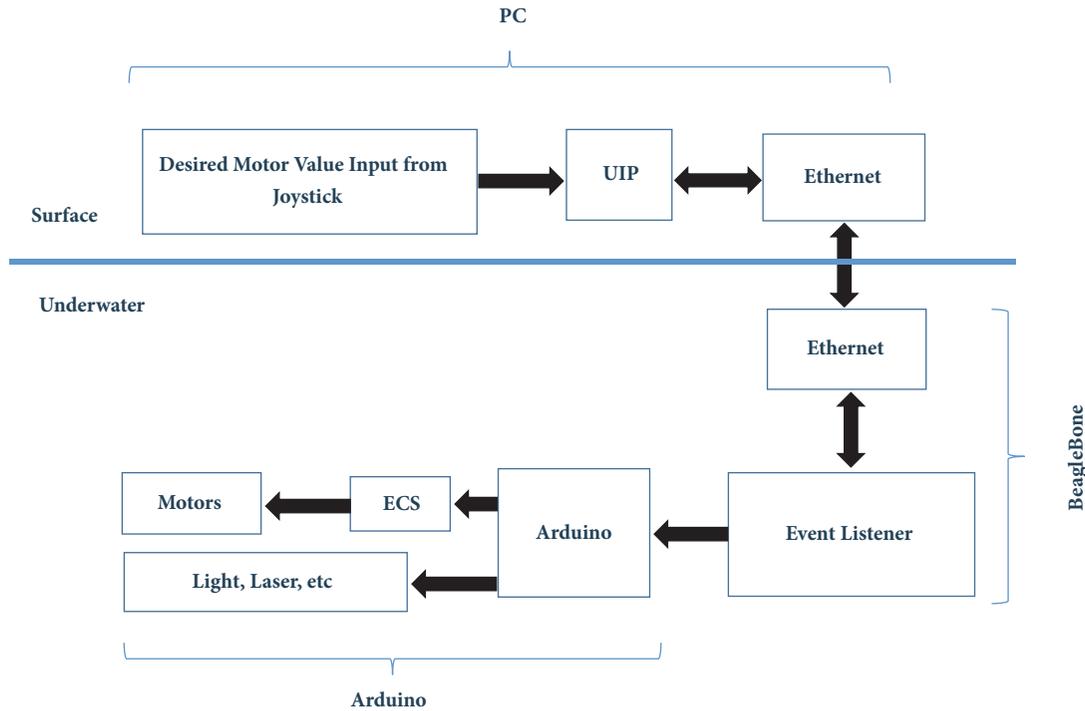


FIGURE 7: Data transmission diagram between UIP and ROV.

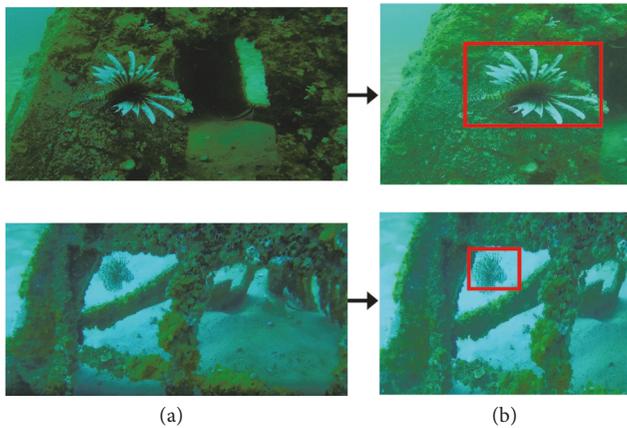


FIGURE 8: Two left images in (a) are original images from red lionfish. The right images (b) are preprocessed to become suitable for the database. Images are taken from in Pete Tide II.

activation function, and a pooling phase. The activation function adds nonlinearity to the input, and the pooling phase is for reducing input size and strengthening learning [35]. Finally, in the last convolutional layer all features are sorted as a vector and sent to next layer. In the training step, the database that contains images and their labels are inputs to CNN. Figure 9 shows the architecture of CNN that was used for detecting the red lionfish. Excluding input layer, there were total of 14 layers. The input size of first layer was set to  $32 \times 32 \times 3$  for all three channels and to have

a coarse-to-fine prediction of features three convolutional layers were used. First convolutional layer size was set to  $5 \times 5$ . Each convolutional layer is followed by a Maxpooling layer. Maxpooling layers were followed by a Rectified Linear Unit (ReLU) layer where this layer is used as a thresholding operator [35, 42].

### 3. Experimental Results

The trained network was tested real-in-time on collected videos with ROV camera from four artificial reefs off the coast of Pensacola, Florida, USA. Figure 10 shows the sites that are visited during the experimental dives. In order to simulate real conditions, no samples from the recorded videos were added to the database. Therefore, the testing procedures could be assumed as a real-world situation, since the results were completely captured in real environment. Figure 11 shows a screenshot from one of the captured videos. Due to algae, green is the dominant color in this video. Also, the background, which is a sunken ship, has some patterns that can be mistaken as red lionfish stripes by a trained CNN. Moreover, as depicted in Figure 11, the similar stripe patterns were detected as a false positive instance. Confidence of false positives was relatively low. Therefore, to avoid false detection, the acceptable confidence level was set to 80%. Figure 12 is a sample of frame that have detected true positive. As depicted in Figure 12, although the red lionfish stripes are not clearly observable, because of other features such as fins, the trained network could successfully distinguish the red lionfish. In addition, as depicted in Figure 13, in a complicated

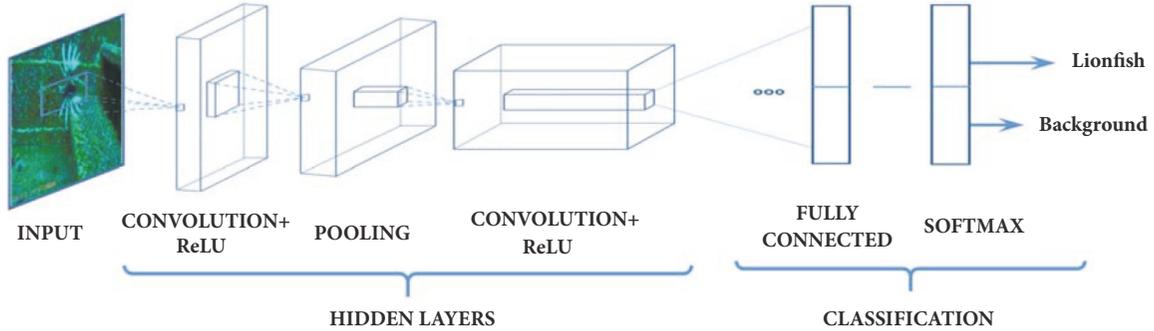


FIGURE 9: The architecture of the CNN that is used to detect red lionfish from other similar fishes and objects underwater.



FIGURE 10: Location of three different sites in Pensacola, Florida, for finding red lionfish.

situation like presence of other fishes, the CNN is capable of detecting the red lionfish with 91% confidence.

In order to find the accuracy of proposed method, 1000 consecutive frames were selected from one of the captured videos from the Pensacola reefs. According to the selected frames, the red lionfish was available in 88.5% of them. Table 1 shows the number of true positive detected red lionfish in 885 frames that contained red lionfish in them. Among the true positive instances there were some frames that contained false positive instances that means despite the presence of red

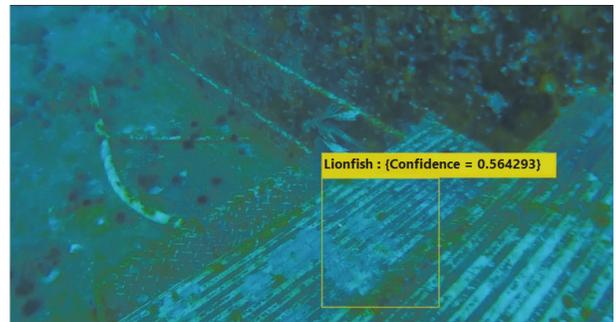


FIGURE 11: An example of false positive detection due to similarity of patterns. Although because of strip patterns this stair was detected as a red lionfish, the confidence was about 0.5 that is below the defined rejection threshold. Location of image is TDC Reef #1.



FIGURE 12: Despite the color change and camouflage, the red lionfish was detected with confidence above 0.8. Location of image is TDC Reef #1.

lionfish in that particular frame another object was wrongly detected as red lionfish with a confidence higher than 80%. Moreover, in some frames like Figure 14, the trained CNN was not able to detect the target due to different conditions like instantaneous turning of the red lionfish that cause a blurry image of it, very low light condition, or far distance. However, since these frames sporadically occur in the video, the overall continuousness of the lionfish tracking is not affected.

TABLE 1: Results of the trained CNN on 1000 frames.

Frames	True Positive Detected	False Positive Detected	Missed Red lionfish
red lionfish	93%	4%	3%

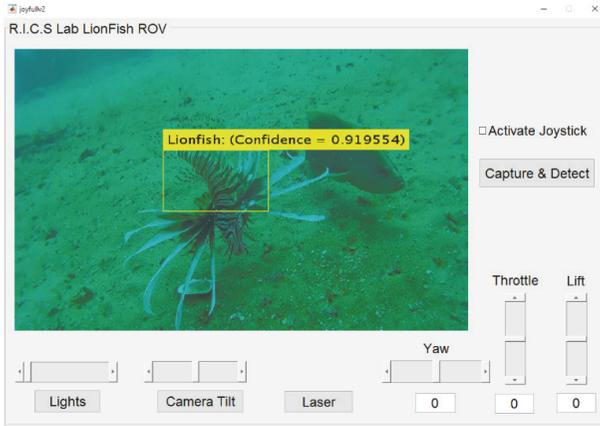


FIGURE 13: The red lionfish was detected correctly despite the presence of other species in the image. Location of image is SE Navy YDT15.

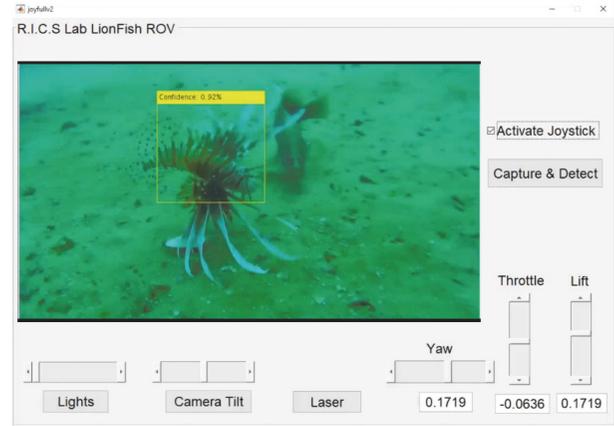


FIGURE 15: The processed video is available at [youtu.be/j43Og--d\\_SQ](https://youtu.be/j43Og--d_SQ). Location of video is SE Navy YDT15.

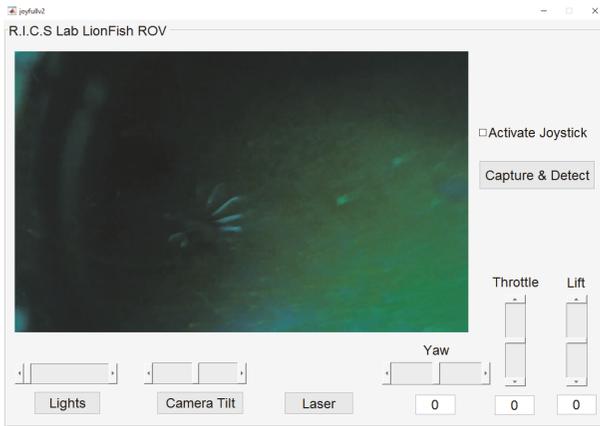


FIGURE 14: The ROV lights were off while the red lionfish tried to hide in a dark environment. Location of image is TDC Reef #1.

In Figure 15, a video containing 500 frames in presence of a red lionfish was selected for evaluating the real-time performance of the trained CNN real-in-time. The average time for processing each frame was measured as 0.097 seconds that leads to at least 10 frames per seconds. Since the red lionfish swim at low speed, the detection system can still notify the user about the presence of the red lionfish in a real-time manner. Figure 16 depicts the processing time for each of the 500 frames in the processed video. Moreover, Figure 17 shows the confidence percentage of detected red lionfish in each frame. In each frame, if the detected object has a confidence level lower than the threshold of 85%, then it is discarded as false positive. The total number of true positive detected objects in 500 frames was 461, which is 92% of the whole frames. Finally, four live performances of the trained

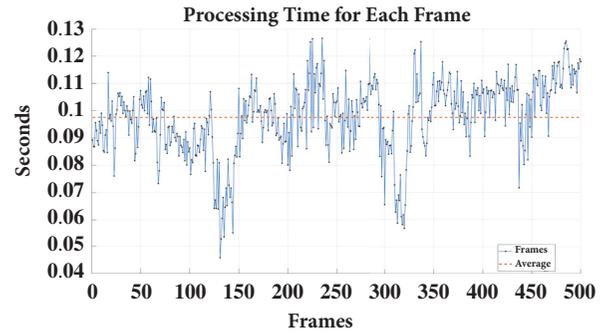


FIGURE 16: Processing time for each frame in the Figure 15's video. The average time for processing frames is 0.097 seconds.

CNN the red lionfish detection scheme can be found in a YouTube video with the address provided in Figure 18.

As one can recognize from the results, the proposed system shows success in real-time detection of red lionfish in a variety of environments and lighting conditions. However, further investigations are required to prove the effectiveness of the system on the number of catches, the time for each dive, and the number of divers needed in each excursion.

#### 4. Conclusions

This study was a proof of concept for the design and implementation of a real-time CNN-based assistive robotic system for divers to locate the red lionfish. The assistive robot is able to find red lionfish at up to 30 meters in depth. The streaming videos from underwater are sent to the surface and processed in real time to detect the red lionfish. The overall design was driven by the needs of portability, high manoeuvrability, low energy consumption, and user

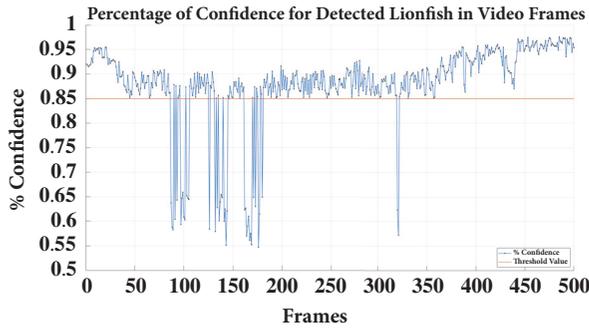


FIGURE 17: The percentage of confidence in each of 500 frames. Detected objects with confidence lower than 85% discarded as nonred lionfish.

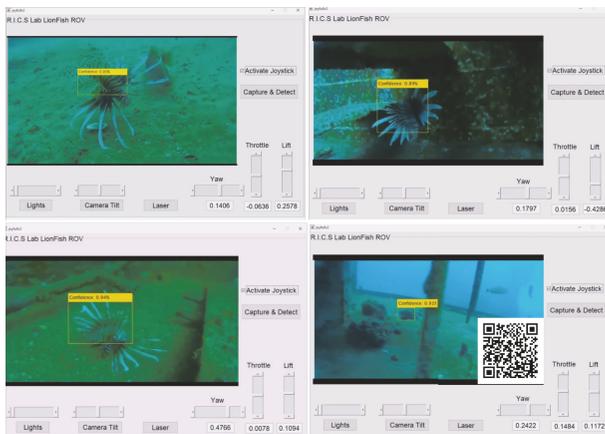


FIGURE 18: Four CNN live performance on three different sites can be found by scanning the QR code or on YouTube at [youtu.be/j43Og\\_-dLSQ](https://youtu.be/j43Og_-dLSQ). Locations of videos are SE Navy YDT15, TDC Reef #1, Pete Tide II, SE Navy YDT15.

friendliness. The proposed scheme is able to perform red lionfish detection in real time, while diving in an underwater environment. The detection system was implemented on an open source, low cost ROV equipped with a camera to collect live videos underwater. Experiments were conducted on recorded videos from marine environments. The main achievement of this work was developing a computer-aided scheme on an affordable set of hardware that can be used environmentalist to detect and remove the lionfish. In the next phase of this research, the focus will be on (1) developing a custom-built ROV especially designed for lionfish detection and removal and (2) investigating the effect of utilizing the proposed assistive scheme on the number of hunts in diving excursions.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This paper was supported by an internal grant from Lamar University, College of Engineering.

## References

- [1] M. A. Albins and M. A. Hixon, "Worst case scenario: Potential long-term effects of invasive predatory lionfish (*Pterois volitans*) on Atlantic and Caribbean coral-reef communities," *Environmental Biology of Fishes*, vol. 96, no. 10-11, pp. 1151-1157, 2013.
- [2] C. S. Elton, *The Ecology of Invasions by Animals and Plants*, University of Chicago Press, USA, 2000.
- [3] R. N. Mack, D. Simberloff, W. M. Lonsdale, H. Evans, M. Clout, and F. A. Bazzaz, "Biotic invasions: Causes, epidemiology, global consequences, and control," *Ecological Applications*, vol. 10, no. 3, pp. 689-710, 2000.
- [4] L. Pejchar and H. A. Mooney, "Invasive species, ecosystem services and human well-being," *Trends in Ecology & Evolution*, vol. 24, no. 9, pp. 497-504, 2009.
- [5] G. M. Ruiz, J. T. Carlton, E. D. Grosholz, and A. H. Hines, "Global invasions of marine and estuarine habitats by non-indigenous species: mechanisms, extent, and consequences," *American Zoologist*, vol. 37, no. 6, pp. 621-632, 1997.
- [6] D. Pimentel, R. Zuniga, and D. Morrison, "Update on the environmental and economic costs associated with alien-invasive species in the United States," *Ecological Economics*, vol. 52, no. 3, pp. 273-288, 2005.
- [7] P. E. Whitfield, J. A. Hare, A. W. David, S. L. Harter, R. C. Muñoz, and C. M. Addison, "Abundance estimates of the Indo-Pacific lionfish *Pterois volitans*/miles complex in the Western North Atlantic," *Biological Invasions*, vol. 9, no. 1, pp. 53-64, 2007.
- [8] W. J. Sutherland, "Horizon scan of global conservation issues for 2011," *Trends in ecology & evolution*, vol. 26, no. 1, pp. 10-16, 2011.
- [9] P. J. Schofield, "Geographic extent and chronology of the invasion of non-native lionfish (*Pterois volitans* [Linnaeus 1758] and *P. miles* [Bennett 1828]) in the Western North Atlantic and Caribbean Sea," *Aquatic Invasions*, vol. 4, no. 3, pp. 473-479, 2009.
- [10] R. M. Hamner, D. W. Freshwater, and P. E. Whitfield, "Mitochondrial cytochrome b analysis reveals two invasive lionfish species with strong founder effects in the western Atlantic," *Journal of Fish Biology*, vol. 71, pp. 214-222, 2007.
- [11] B. X. Semmens, E. R. Buhle, A. K. Salomon, and C. V. Pattengill-Semmens, "A hotspot of non-native marine fishes: Evidence for the aquarium trade as an invasion pathway," *Marine Ecology Progress Series*, vol. 266, pp. 239-244, 2004.
- [12] R. Ruiz-Carus, R. E. Matheson Jr., D. E. Roberts Jr., and P. E. Whitfield, "The western Pacific red lionfish, *Pterois volitans* (Scorpaenidae), in Florida: Evidence for reproduction and parasitism in the first exotic marine fish established in state waters," *Biological Conservation*, vol. 128, no. 3, pp. 384-390, 2006.

- [13] M. A. Albins and M. A. Hixon, "Invasive Indo-Pacific lionfish *Pterois volitans* reduce recruitment of Atlantic coral-reef fishes," *Marine Ecology Progress Series*, vol. 367, pp. 233–238, 2008.
- [14] A. B. Barbour, M. L. Montgomery, A. A. Adamson, E. Díaz-Ferguson, and B. R. Silliman, "Mangrove use by the invasive lionfish *Pterois volitans*," *Marine Ecology Progress Series*, vol. 401, pp. 291–294, 2010.
- [15] S. J. Green, J. L. Akins, A. Maljković, and I. M. Côté, "Invasive lionfish drive Atlantic coral reef fish declines," *PLoS ONE*, vol. 7, no. 3, 2012.
- [16] J. A. Morris Jr. and J. L. Akins, "Feeding ecology of invasive lionfish (*Pterois volitans*) in the Bahamian archipelago," *Environmental Biology of Fishes*, vol. 86, no. 3, pp. 389–398, 2009.
- [17] L. C. T. Chaves, J. Hall, J. L. L. Feitosa, and I. M. Côté, "Photo-identification as a simple tool for studying invasive lionfish *Pterois volitans* populations," *Journal of Fish Biology*, vol. 88, no. 2, pp. 800–804, 2016.
- [18] A. ElHage, *Lionfish Containment Unit, commonly known as the ZooKeeper*, Google Patents, 2015.
- [19] G. Waugh, *System for harvesting marine species members including those that present a danger to a harvester*, Google Patents, 2014.
- [20] F. Ali, K. Collins, and R. Peachey, "The role of volunteer divers in lionfish research and control in the caribbean," *Joint International Scientific Diving Symposium*, p. 7.
- [21] F. Ali, K. Collins, and R. Peachey, *The role of volunteer divers in lionfish research and control in the Caribbean*, 2013.
- [22] M. A. Goodrich and A. C. Schultz, "Human-robot interaction: a survey," *Foundations and Trends in Human-Computer Interaction*, vol. 1, no. 3, pp. 203–275, 2007.
- [23] Y. LeCun, F. J. Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 2, pp. II-97–II-104, Washington, DC, USA, June 2004.
- [24] J. A. Lines, R. D. Tillett, L. G. Ross, D. Chan, S. Hockaday, and N. J. B. McFarlane, "An automatic image-based system for estimating the mass of free-swimming fish," *Computers and Electronics in Agriculture*, vol. 31, no. 2, pp. 151–168, 2001.
- [25] C. Spampinato, Y.-H. Chen-Burger, G. Nadarajan, and R. B. Fisher, "Detecting, tracking and counting fish in low quality unconstrained underwater videos," in *Proceedings of the 3rd International Conference on Computer Vision Theory and Applications, VISAPP 2008*, pp. 514–519, Portugal, January 2008.
- [26] E. Harvey, M. Cappel, M. Shortis, S. Robson, J. Buchanan, and P. Speare, "The accuracy and precision of underwater measurements of length and maximum body depth of southern bluefin tuna (*Thunnus maccoyii*) with a stereo-video camera system," *Fisheries Research*, vol. 63, no. 3, pp. 315–326, 2003.
- [27] P. Tokekar, E. Branson, J. Vander Hook, and V. Isler, "Tracking aquatic invaders: Autonomous robots for monitoring invasive fish," *IEEE Robotics and Automation Magazine*, vol. 20, no. 3, pp. 33–41, 2013.
- [28] L. Wu, X. Tian, J. Ma, and J. Tian, "Underwater object detection based on gravity gradient," *IEEE Geoscience and Remote Sensing Letters*, vol. 7, no. 2, pp. 362–365, 2010.
- [29] J. Anderson, B. Lewis, and C. O'Byrne, *Intelligent Fish Classification in Underwater Video*, Research Experience for Undergraduates (REU) and the University of New Orleans funded by grant from the National Science Foundation, 2011.
- [30] M. Elawady, *Sparse coral classification using deep convolutional neural networks*, 2015, arXiv preprint arXiv:1511.09067.
- [31] H. Qin, X. Li, Z. Yang, and M. Shang, "When underwater imagery analysis meets deep learning: A solution at the age of big visual data," in *Proceedings of the MTS/IEEE Washington, OCEANS 2015*, USA, October 2015.
- [32] M. Moniruzzaman, S. M. Islam, M. Bennamoun, and P. Lavery, "Deep Learning on Underwater Marine Object Detection: A Survey," in *Advanced Concepts for Intelligent Vision Systems*, vol. 10617 of *Lecture Notes in Computer Science*, pp. 150–160, Springer International Publishing, Cham, 2017.
- [33] S. A. Siddiqui, A. Salman, M. I. Malik et al., "Automatic fish species classification in underwater videos: Exploiting pre-trained deep neural network models to compensate for limited labelled data," *ICES Journal of Marine Science*, vol. 75, no. 1, pp. 374–389, 2018.
- [34] H. Qin, X. Li, J. Liang, Y. Peng, and C. Zhang, "DeepFish: Accurate underwater live fish recognition with a deep architecture," *Neurocomputing*, vol. 187, pp. 49–58, 2016.
- [35] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [36] "OpenROV (n.d.)," <https://www.openrov.com/>.
- [37] "Socket.io (n.d.)," <https://socket.io/docs/>.
- [38] "Node.js (n.d.)," <https://nodejs.org/en/>.
- [39] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS '12)*, pp. 1097–1105, Lake Tahoe, Nev, USA, December 2012.
- [40] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, pp. 91–99, 2015.
- [41] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York, NY, USA, 2006.
- [42] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, Cambridge, Mass, USA, 2016.

