Design and implementation of lightweight AUV with multi-sensor aided for underwater intervention tasks

Peng Xu, Jiaxi Zheng, Xinyu Wang, Siyuan Wang, Jianhua Liu, Xiangyu Liu, Guangming Xie, Jin Tao and Minyi Xu

***Abstract*— Autonomous underwater vehicles (AUVs) are commonly used to conduct complex underwater tasks, such as marine infrastructure overhaul and maintenance, environmental monitoring, oceanographic mapping, and organism capture. These tasks require the ability of an AUV to perform autonomous navigation, especially when communication is limited in the underwater environment. This paper developed a new type of lightweight intervention AUV for autonomous navigation using data from multiple inertial sensors, where multi-sensor error state Kalman filter schemes are preferable to standard Kalman Filters in terms of the AUV's motion estimates. Concerning target recognition, a color restoration method is provided for degraded underwater images and a You Only Look Once strategy is combined with topological analysis for object detection. In addition, the proposed design is robust in terms of its software components and mechanical structure, which provides a feasible platform for AUV's secondary development. Experiments of surveying and object manipulation conducted in underwater environments demonstrate the functionality of the entire system and its potential applications in the fields of science and industry.**

I. Introduction

In recent decades, underwater intervention tasks, such as organism catching [1], [2], underwater infrastructures construction and maintenance [3], [4], underwater search and rescue missions [5], [6], have been chiefly conducted under extensive human supervision with the help of remotely operated vehicles (ROVs). To be specific, launched from a dedicated mother vessel with umbilical cables for energy supply and communication, an ROV requires an experienced operator with highly focused attention, which results in increased operational costs and human security protocols.

With technological developments in underwater persistent battery systems and powerful processors, autonomous underwater vehicles (AUVs) have become a promising alternative to ROVs for underwater intervention tasks, especially those not requiring direct human supervision. The AUVs can also be equipped with manipulators, called intervention AUV (I-AUV). Furthermore, AUVs have higher maneuverability due to self-carried batteries and no cable. Therefore, they can automatically sense the environment and perform mechanical work in hazardous areas, such as deep oceans, sub-glacial waters, and natural or artificial catastrophic underwater regions. Given the research of AUVs, the history can be traced back to the 1990s [7], [8], [9], [10]. However, as far as we know, there exist limited successful examples of the implementation of AUVs. The first relevant live demonstration of an AUV was found in the SWIMMER EU project [11]. In this case, AUV was proposed as an ROV carrier. The AUV is responsible for automatically navigating to the offshore infrastructure so as to dock to establish a connection with the operator on the base station, which allows standard ROV operations without tightening the rope on the ground. The following steps, i.e., those toward environmental perception, location, analysis, decision making, as well as autonomous and independent missions execution in complicated environments were presented in [12]. The GIRONA 500 I-AUVs described in [13] symbolize milestone progress for I-AUVs, which can perform underwater tasks involving fully autonomous manipulations.

The accuracy of I-AUV manipulation depends on the manipulator itself and its controller and the precision of the vehicle's sensory equipment. Vision is one of the most widely used abilities regarding temporal or spatial resolution, especially in underwater environments. Its performance is better than sonar or laser range finders. In general, the underwater environment can be characterized by non-uniform lighting and shadows, suspended particles, or marine life, and light attenuation and scattering, requiring AUVs to be able to conduct robust underwater image processing [14]. In a light-attenuation condition, a color registration algorithm was proposed to recover degraded color information [15]. In the past few decades [16], [17], traditional machine learning methods aimed at underwater image-feature extraction have become popular. Moreover, deep-learning algorithms may improve AUV's ability for marine organism perception detection and recognition. For instance, using a faster region-based convolution-neural-network model, different marine organism data-augmentation strategies were evaluated in [18], [19]. This algorithm is stronger robust to varying environments like motion blur, illumination changes, and

*The authors would like to thank the National Key R & D Project from the Minister of Science and Technology (2021YFA1201604), the Dalian Outstanding Young Scientific and Technological Talents Project (2021RJ11), the National Natural Science Foundation of China (62003175, 51879022), and the Academy of Finland (Grant No. 315660). Peng Xu and Jiaxi Zheng contributed equally. (Corresponding authors: Jin Tao and Minyi Xu.)

Peng Xu, Xinyu Wang, Jianhua Liu, Xiangyu Liu, Siyuan Wang, and Minyi Xu are with Marine Engineering College, Dalian Maritime University, Dalian 116026, China (email: pengxu@dlmu.edu.cn, xinyuwang@dlmu.edu.cn, ykliujianhua@dlmu.edu.cn, simonlxy@dlmu.edu.cn, wsydlmu@dlmu.edu.cn, and xuminyi@dlmu.edu.cn)

Jiaxi Zheng is with Transportation Engineering College, Dalian Maritime University, Dalian 116026, China (email: 2220194051jacky@dlmu.edu.cn)

Guangming Xie is with Intelligent Biomimetic Design Lab, College of Engineering, Peking University, Beijing100871, China (email: xieguangming@pku.edu.cn)

Jin Tao is with College of Artificial Intelligence, Nankai University, Tianjin 300350, China, he is also with Silo AI, Helsinki 00100, Finland (email: taoj@nankai.edu.cn)

This article has been accepted for publication in IEEE Transactions on Circuits and Systems--II: Express Briefs. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSII.2022.3193300

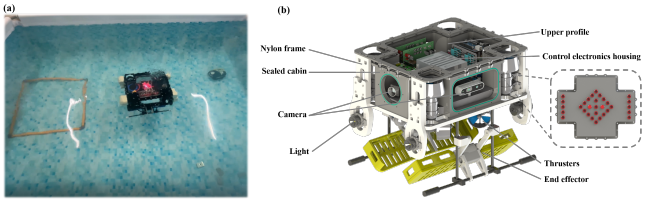

Fig. 1. Mechanical structure of the AUV. (a) Integrated AUV during pool trial. (b) Basic three-dimensional model of AUV.

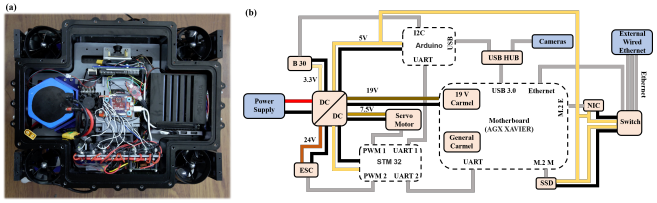

Fig. 2. AUV hardware. (a) Layout of AUV hardware. (b) Schematic of AUV hardware

perspective distortion than traditional algorithms. It is worth noting that from the viewpoint of practical applications, processing speed is an important performance indicator for object detection and recognition. As an end-to-end algorithm, You Only Look Once (YOLO) has advantages in processing speed, which can reach up to 45 frames per second (fps) [20]. Therefore, in our studies, the YOLO model is used to perform vision-based target location.

This paper designs AUVs mechatronic systems including mechanical design, software and hardware integration, end-effector control, and real-time image detection. The main contributions of this paper are listed as follows: (i) Different from the relatively larger size of the proposed AUVs in [11], [13], a new type of lightweight AUV is designed for underwater exploration and exploitation missions. Their weight (12 kg) and dimensions (388*312*121 mm) allow the vehicles to be deployed from small ships, thereby considerably reducing operational costs. (ii) The YOLO strategy is combined with topological analysis for target detection, where a color-restoration method for degraded underwater images is employed. In addition, although deep-learning algorithms are wildly investigated concerning marine organism detection in [18], it is rarely reported that this solution is deployed on AUVs systems for the improvement of perception detection. (iii) With the help of data from multiple inertial sensors, a multi-sensor error-state Kalman filter (MESKF) is proposed for estimating the vehicle's motion. Compared to standard Kalman Filters in [21], the MESKF scheme can significantly reduce navigation errors accumulated over time. Testing results demonstrate successful detection for different positions and target orientations running online with video sequences (12-15 fps).

The remainder of this paper is organized as follows. In Section II, a general description of the platform is presented. Section III presents the different components constituting AUV's software. Details of the experimental results are demonstrated in Section IV. Section IV gives the conclusion of this paper.

II. Platform Description

A. Mechanical design

As shown in Fig. 1 (a), the integrated AUV is undergoing pool trials. The AUV's mechanical structure mainly includes an ower bottom plate, thrusters, upper-hatch-cover pressure plate, upper hatch cover, a sealed control electronics cabin, and slab connection frame, among other components, as shown in Fig. 1 (b). The external dimension is designed to be 388*312*121 mm. The sealed control electronics cabin's main body (80 mm in height) is made of aluminum alloy, and the surface is processed by computer numerical control (CNC) machining and anodized to prevent corrosion by salt-ion solution. Because of the space limitation, symmetrical circular holes with a diameter of 60 mm are drilled on one pair of sides of the sealed cabin. Rectangular holes with a side length of 166*88 mm are opened on the other pair of sides to seal the cabin and expand the field of view of the camera, as well as facilitate opening and maintenance, and debugging. The top surface is wholly hollowed with O-ring grooves on the edge. The O-ring's static compression design is within $15\% - 30\%$. A transparent acrylic board is applied as a waterproof sealing plate and attached to the upper top surface. In order to evenly stress the sealing plate, it is fastened to an anodized aluminum alloy layer on the upper-hatch-cover pressure plate with screws. The edge of the upper hatch cover is vertically fixed with a nylon board-layer connection frame, and a LED light-source support frame printed with Asa water is set on the board-layer connection frame. The LED light source adjusts the light-source irradiation area by changing the position of the support frame in the slot.

In order to improve its scalability and facilitation of carrying the equipment outside the AUV, the sealed bilge is constructed with 31 fixed threading bolt holes, as shown Fig. 1(b). The positions of anodized aluminum alloy made threading bolts actively avoid the inlet and outlet channels of the thruster. The focus is on the center and edges of the cabin bottom. To guarantee the maneuverability of the AUV, we adopt the structural design of four horizontal propellers and four vertical propellers, where there exits no interference between the inflow and the outflow of vertical and horizontal propellers, thus guaranteeing the motion independence between X, Y, and Z axes. Compared with a vertical double-thruster structure, the vertical four-thruster design considers the stability of depth control. Specifically, we fixed the vertical thruster on the board layer connection frame and the horizontal thruster on the lower base plate. We also equipped the floor layer with a pair of grippers to allow the AUV to maintain its flexibility and stability. Each gripper joint, corresponding to a degree of freedom, has its angle feedback.

B. Hardware platform

The hardware contains the following components, as shown in Fig. 2. A mini motherboard (100*87 mm) based on 64 tensor cores and an eight-core ARM CPU running at 2.26 GHz execute all the operational tasks related to remote communications, sensor data reading, and image processing. A USB 3.0 interface connects the cameras and an Arduino board to the motherboard. The Arduino board was specifically developed and can manage multiple water-leak detectors and a B30 pressure sensor. STM 32 controls thrusters via an electronic speed controller (ESC). A 250-W DC/DC converter is used to power the motherboard, microcontroller boards, hard drives, and switches. All of the elements mentioned above are enclosed in the sealed control electronics cabin. Other structural parameters include the following:

- Load weight in water: 0-10 kg.
- Weight in air: 12 kg.
- Operating temperature: -5 to 35 deg C.
- Power consumption: 60-100 W.

If there is no external power source, the AUV can be powered by an internal battery system to maintain proper autonomous operation. Given that the average power consumption of the AUV is approximately 60 W, a battery with a nominal voltage of 24 V and a nominal current of 10 Ah will provide more than four h of autonomous operation. Any standard device, such as HD, SD card, SSD, or USB drive, can be used in terms of the system's storage capacity. A selection criterion would be based only on storage capacity requirements and specific consumption.

III. Navigation Layer

The AUV's software architecture comprises independent components. Each component is responsible for a specific task, grouped into three different layers according to its role. In particular, the three layers include end-effector, navigation, and perception layers, as shown Supplementary Fig. 3. The leading layer is the navigation layer, which contains the drivers for all the navigation sensors and localization filters for estimating the AUV's pose and velocity. To interact with the sensor hardware, the driver reads the ICM20602 inertial measurement unit (IMU), including a k8975 magnetometer, a three-axis accelerometer, and a three-axis gyroscope. Integrated with the data captured by the above sensors, a quaternion-based MESKF typically consists of two steps, prediction and update are used. In the prediction step, the AUV estimates its immediate motion from both a set of sensor data and dynamic models. In the update step, the AUV is permitted to correct the prediction via real observations. Moreover, two stack vectors, i.e., the nominal states vector and errors vector, are introduced in the process. The state vector $x(k)$ containing the navigation data at the k iteration is given by

$$x(k) = (p, q, M, v, \omega, a, d, b, c), \tag{1}$$

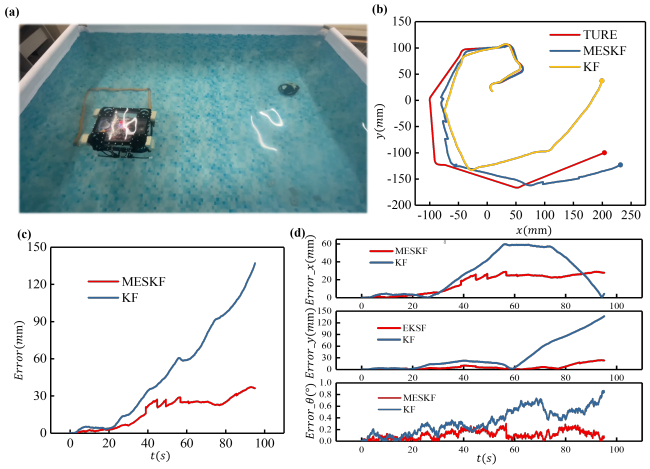

Fig. 3. Experiment using multiple inertial sensors. (a) Experimental pool. (b) Comparison of tracking true trajectory with MESKF and KF. (c) Comparison of distance errors with MESKF and KF. (d) Comparison of error component with MESKF and KF.

where $p = [x, y, z]$ represents the AUV's position, $q = [q_x, q_y, q_z, q_w]$ denotes its attitude as a quaternion, $M = [M_x, M_y, M_z]$ represents the magnetic component; $v = [v_x, v_y, v_z]$, $\omega = \omega_x, \omega_y, \omega_z$, and $a = [a_x, a_y, a_z]$ denote its linear velocity, angular rate, and acceleration, respectively, while $d = [d_x, d_y, d_z]$, $b = [b_x, b_y, b_z]$, and $c = [c_x, c_y, c_z]$ represent the corresponding measurement biases of ω, a, and M, respectively.

From (1), $X_h = M_x cos(\theta) + M_y sin(\theta) sin(\varphi) + M_z sin(\theta) cos(\varphi)$, $Y_h = M_y cos(\varphi) - M_z sin(\varphi)$ and $\xi = arctan(Y_h / X_h)$ are defined, in which θ and φ are the pitch and roll angles, respectively. In addition, the readings of the inertial sensors contain a particular bias and noise that obeys Gaussian statistics with zero means. Thus, the real values of angular rate and acceleration can be achieved via sensor measurements, $\omega = \omega_m - d - \eta_\omega$, and $a = a_m - b - \eta_a$, $M = M_m - c - \eta_M$, where ω_m is the reading of the gyroscope, a_m denotes that of the accelerometer, M_m that of the magnetometer, and η_ω, η_a, and η_M represent the Gaussian noise in the readings of the respective inertial sensors.

Moreover, the accelerometer reading in the body frame with respect to gravity acceleration is typically given as $a_m(k) = g - a_t(k)$, where $g = [0, 0, 9.8\ m/s^2]$ is the the gravity acceleration and $a_t(k)$ denotes the real body acceleration at the kth iteration. The error vector including errors resulting from sensor biases and random noise is written as $\delta(k) = (\delta_p, \delta_q, \delta_M, \delta_v, \delta_\omega, \delta_a)$, where δ_p are errors in position, δ_q denote errors in attitude, δ_M are errors in the magnetic component, and δ_v, δ_ω, and δ_a are errors in linear velocity, angular rate, and acceleration, respectively.

The prediction stage of MESKF relies on a simple constant-velocity kinematics model to predict how the state evolves from time k to time $k+1$. The discrete equation $x(k+1) = f(x(k), \eta)$ that governs the nominal state is based

This article has been accepted for publication in IEEE Transactions on Circuits and Systems--II: Express Briefs. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSII.2022.3193300

on general motion under constant acceleration

$$
\begin{aligned}
&p(k+1) = p(k) + Rv(k)\Delta_t + \frac{1}{2}Ra(k)\Delta_t^2,\\
&\omega(k+1) = \omega_m(k) - d(k),\\
&v(k+1) = v(k) + a(k)\Delta_t,\\
&a(k+1) = R'(k)g - a_m(k) + b(k),\\
&q(k+1) = q_m * q(k), b(k+1) = b(k),\\
&d(k+1) = d(k), c(k+1) = c(k),\\
&\Phi_M(k+1) = \begin{cases} 180^\circ - \xi(k), X_h(k) < 0, Y_h(k) < 0\\ \xi(k), X_h(k) > 0, Y_h(k) < 0\\ 360^\circ - \xi(k), X_h(k) > 0, Y_h(k) > 0\\ 180^\circ + \xi(k), X_h(k) < 0, Y_h(k) > 0, \end{cases}
\end{aligned}
$$

where $\omega(k+1)$, $v(k+1)$, $a(k+1)$, $p(k+1)$, $q(k+1)$, and $\Phi_M(k+1)$ denote the predicted values of the angular rate, linear velocity, acceleration, position, attitude, and magnetic azimuth, respectively. $b(k+1)$, $d(k+1)$, and $c(k+1)$ are the predicted measurement biases of ω, a, and M, respectively, while R denotes the body rotation matrix corresponding to the quaternion $q(k)$ and R' is its inverse. Δ_t represents the time interval between the kth and $k+1$st iterations, and $q_m = \omega_m\Delta_t$ is the quaternion determined by the AUV's angular motion during Δ_t. Here, the reference frame of $v(k+1)$,$a(k+1)$ is the local body frame, so the term $R'(k)g$ implies that the gravity g in the inertial frame is transformed into the body frame. However, the position in the inertial frame must be provided to perform the navigation tasks. Then the terms $Rv(k)$ and $Ra(k)$ are used to attain the velocity and acceleration in the inertial frame. In addition, the bias terms $b(k)$, $d(k)$, and $c(k)$ are assumed to be constant. The nominal state typically computes the error state and its corresponding covariance. Letting the error state be defined as the difference between the estimated variable and its absolute quantity, the errors in angular velocity, acceleration, and magnetic component are regarded as constant. From [22], the discrete equation that govern the remaining errors state is given by

$$
\begin{aligned}
\delta_p(k+1) =& \delta_p(k) + R\delta_v(k)\Delta_t - R(v(k)\times\delta_q(k))\Delta_t +,\\
&\frac{1}{2}(a(k)\times\delta_q(k))\Delta_t^2 + \frac{1}{2}R\delta_b\Delta_t^2,\\
\delta_v(k+1) =& \delta_v(k) + (R'g)\times\delta_q(k)\Delta_t + \delta_b\Delta_t,\\
\delta_q(k+1) =& D\delta_q(k) + (-I_{3*3}\Delta_t + \frac{1-cos(|\omega(k)|\Delta_t)}{|\omega|^2(k)}S,\\
&-\frac{|\omega(k)|\Delta_t - sin(|\omega(k)|\Delta_t)}{|\omega|^3(k)}S^2)d\Delta_t,
\end{aligned}
\tag{2}
$$

where D represents the rotation matrix corresponding to ω_m, I_{3*3} denotes the $3*3$ identity matrix, $\times$ is the cross-product, and the skew symmetric matrix S of the velocity is given as

$$
\begin{pmatrix} 0 & -v_z & v_y\\ v_z & 0 & -v_x\\ -v_y & -v_x & 0 \end{pmatrix}.
$$

Using (2), when the nominal state is corrected, the estimated error value is always zero. However, the covariance matrix $P(k)$ of the error at the kth iteration is not zero. According to the Kalman equation, the error state covariance matrix can be propagated as

$$
P(k+1) = F(k)P(k+1)F^T(k) + Q_k, \tag{3}
$$

where $F(k)$ is the Jacobian of (2) and Q_k denotes the noise covariance.

Before proceeding, and let $\delta_m = x_m - \hat{x}_m$ denote the measurement error between the sensor readings x_m and their prediction $\hat{x}_m$, the measurement residual in the kth iteration is then given as

$$
y_m(k) = \delta_m(k) - H(k)\hat{\delta}(k), \tag{4}
$$

where $H(k)$ denotes the linearized observation model and $\hat{\delta}(k)$ represents the estimated errors. Note that the prediction errors are updated using the measurement error, and the classical Kalman updated equation [21]. Then, using the error-state vector updated in the previous stage, the nominal state vector is corrected as

$$
x(k) = x(k) + \hat{\delta}(k). \tag{5}
$$

The error state vector is reset to zero when the correction is completed. The algorithm starts again in the next iteration and predicts the new nominal state value based on the corrected vector $x(k)$ and the new inertia input. Subsequently, to verify the proposed MESKF implementation, experiments were conducted in a water tank with dimensions $3000*2000*1000$ mm (shown in Fig. 4(a)) to track the setting truth trajectory. Notably, the AUV hovers over the bottom using data from the multiple sensors and maintains its altitude throughout the experimental process. Thus, the experiments are carried out in the $x-y$ plane. In Fig. 4(b), the setting truth trajectory is plotted in red, while the trajectory of the standard KF is plotted in yellow and that of the MESKF in blue. Fig. 4(c) involves the comparison of tracking accurate trajectory with the MESKF and KF, where the error between the MESKF and KF is 112 mm. Moreover, Fig. 4(d) shows a comparison of the error components with the MESKF and KF (also see supplementary materials). The experimental results demonstrate that the proposed MESKF outperforms the standard KF in the estimated trajectories.

IV. Experimental results

A wide range of experiments was carried out in an indoor pool with the system described in Section III. First, the AUV's underwater target-detection capabilities using the proposed vision system were tested. As shown in Supplementary Movie 1, after visual detection of the target, the AUV kept a fixed distance relative to the red square while waiting for the intervention to begin. Once the AUV detected the moving square, the intervention was executed in a free-floating mode that controlled the vehicle to track the moving square and set its distance from the court. Despite overshooting in the control process, the AUV accomplished several vision-guided tracking trials.

The robustness of the AUV's anti-disturbance control was subsequently tested. In the anti-disturbance testing, a random disturbance was used to change the position and attitude

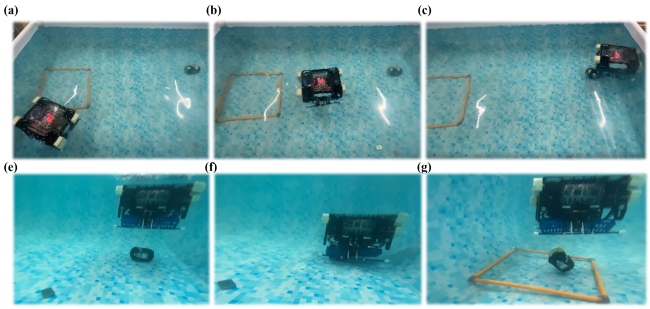

Fig. 4. Robotic fish recovery task grasping sequence. From (a) to (c): searching the target in free-floating mode and initiating the grasp. From (d) to (f): Grasp completed and recovery.

of both the AUV and the end effector during this final demonstration (see Supplementary Movie 2). The AUV could successfully maintain the desired position and attitude with data from inertial sensors and the proposed algorithm.

Furthermore, a robotic fish recovery task was automatically completed without any outer guideline, where a training set of robot fish is provided to the YOLO network, resulting in the AUV with the ability of robot fish detection (see Supplementary perception layer section. The AUV conducted a floating-mode search for the target object on the water surface using the bottom camera. Once the AUV detected the robotic fish, the AUV moved directly above the target object. When reaching the target position, the end effector initiated the pre-grasp sequence until the contact with the object was deleted. In the grasp phase, before applying the pressure required to lift the object, the contact points were analyzed to assess the stability of the grasp. In addition, we also used the information of the encoder to estimate the grab weight of the AUV. After confirming that the grab was successful, the AUV started the recovery phase, took the object, and placed it in the yellow square (see Supplementary Movie 3). The sequence of images in Fig. 4 shows the grasping operation from searching for the target to grasp recovery.

V. Conclusions

A new class of lightweight intervention AUVs highlighting successful hardware and software integration is described in this paper. The designed intervention AUV is incredibly flexible, equipped with several sensing equipment, external power supply, and Ethernet connections, thus avoiding hardware and software structure changes. Data from multiple inertial sensors rather than a single sensor enable the AUV to navigate in performing underwater tasks autonomously. Error states are used to design the navigation algorithm. Cameras can be placed in different positions depending on the specific missions to be carried out. The YOLO strategy combined with topological analysis can be applied for object detection and recognition. Experimental results from tests conducted in water tanks demonstrated the system's usefulness in tasks like object identification, surveying, anti-disturbance control, and manipulation.

References

[1] H. Huang et al., "A review on underwater autonomous environmental perception and target grasp, the challenge of robotic organism capture," *Ocean Eng.*, vol. 195, no. 1, pp. 106644, Jan.2020.

[2] J.Y. Li, H. Zhou, H. Huang, Y. Xu, Z. L. Wan and L. Wan, "Design and vision-based autonomous capture of sea organism with absorptive type remotely operated vehicle," *IEEE Access*, vol. 6, pp. 73871-73884, Nov. 2018.

[3] M. Dan, J. John, H. Mark, and E. Carl, "Autonomous inspection using an underwater 3D LiDAR," in 2013 *OCEANS-San Diego*, 2013, pp. 1-8.

[4] C. Mai, S. Pedersen, L. Hansen, K. L. Jepsen, and Z. Y. Yang, "Subsea infrastructure inspection: A review study," in *2016 IEEE Int. Conf. Underwater Syst. Technol. (USYS)*, 2016, pp. 71-76.

[5] A. Matos et al., "Multiple robot operations for maritime search and rescue in euRathlon 2015 competition," in *OCEANS 2016-Shanghai*, 2016, pp. 1-7.

[6] A. Birk, B. Wiggerich, H. Bülow, M. Pfingsthorn and S. Schwertfeger, "Safety, security, and rescue missions with an unmanned aerial vehicle (UAV)," *J. Intell. Robot. Syst.*, vol. 64, pp. 57-76, Jan. 2011.

[7] H. H. Wang, S. M. Rock, and M. J. Lee, "Experiments in automatic retrieval of underwater objects with an AUV," in *Proc. MTS/IEEE Challenges Changing Global Environ. Conf.(OCEANS'95 MTS/IEEE)*, 1995, pp. 366-373.

[8] S. Choi, G. Takashige and J. Yuh, "Experimental study on an underwater robotic vehicle: Odin," in *Proc. Symp. Auton. Underwater Veh. Technol.(AUV'94)*, 1994, pp. 79-84.

[9] V. Rigaud et al., "UNION: Underwater intelligent operation and navigation," *IEEE Robot. Autom. Mag.*, vol. 5, no. 1, pp. 25-35, May 1998.

[10] D. M. Lane, et al., "AMADEUS: Advanced manipulation for deep underwater sampling," *IEEE Robot. Autom. Mag.*, vol. 4, no. 4, pp. 34-45, Dec. 1997.

[11] J. Evans, K. Keller, J. Smith, P. Marty, and O. Rigaud, "Docking techniques and evaluation trials of the SWIMMER AUV: An autonomous deployment AUV for work-class ROVs," in *Proc. MTS/IEEE Conf. Exhib. OCEANS*, 2001, pp. 520-528.

[12] C, Barb alat a, M. W. Dunnigan and Y. Petillot, "Dynamic coupling and control issues for a lightweight underwater vehicle manipulator system," in *2014 Oceans-St. John's*, 2014, pp. 1-6.

[13] R. David et al., "I-AUV mechatronics integration for the TRIDENT FP7 project," *IEEE-ASME T. Mech.*, vol. 20, no. 5, pp. 2583-2592, Feb. 2015.

[14] R. Schettini and S. Corchs, "Underwater image processing: state of the art of restoration and image enhancement methods," in *EURASIP J. Adv. Signal Process.*, vol. 2010, no. 746052, Feb. 2010.

[15] A. Yamashita, M. Fujii and T Kaneko, "Color registration of underwater images for underwater sensing with consideration of light attenuation," In *Proc. IEEE Int. Conf. Robot. Autom.*, 2007, pp. 4570-4575.

[16] J. Garca, J. Fernandez, P. Sanz and R. Marn, "Increasing autonomy within underwater intervention scenarios: the user interface approach," in *Proc. IEEE Syst. Conf.*, 2010, pp. 71-75.

[17] F. Sun, J. Yu, S. Chen and D. Xu, "Active visual tracking of free-swimming robotic fish based on automatic recognition," In *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 2879-2884.

[18] H. Huang, H. Zhou, X. Yang, L. Zhang, L. Qi and A. Y. Zang, "Faster R-CNN for marine organisms detection and recognition using data augmentation," *Neurocomputing*, vol. 337, pp. 372-384, Apr. 2019.

[19] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only Look once: unified, real-time object detection," In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 779-788.

[20] J. Redmon and F. Ali, "Yolov3: An incremental improvement," arXiv preprint arXiv:1804.02767, 2018.

[21] A. J. Izenman, "An introduction to Kalman filtering with applications," *Technometrics*, vol. 30, no. 2, pp. 243-244, 1988.

[22] N. Trawny and S. I. Roumeliotis, "Indirect Kalman filter for 3D attitude estimation," *University of Minnesota, Dept. of Comp. Sci. Eng., Tech. Rep*, vol. 2, no. 2005, 2005.