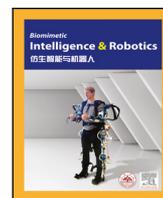


Contents lists available at [ScienceDirect](https://www.sciencedirect.com)

Biomimetic Intelligence and Robotics

journal homepage: www.elsevier.com/locate/birob

A decoupling three-dimensional motion control algorithm for spherical underwater robot

Pengxiao Bao^a, Yao Hu^a, Liwei Shi^{a,*}, Shuxiang Guo^b, Zhengyu Li^a^a Key Laboratory of Convergence Medical Engineering System and Healthcare Technology, Ministry of Industry and Information Technology, School of Life Science, Beijing Institute of Technology, Beijing 100081, China^b Faculty of Engineering, Kagawa University, Hayashi-cho, Takamatsu 2217-20, Japan

ARTICLE INFO

Keywords:

Motion control algorithm
Decoupling control
Spherical underwater robots
Fuzzy PID

ABSTRACT

Underwater spherical robots are good assistants for ocean exploration, where motion control algorithms play a vital role. Conventional motion control algorithms cannot eliminate the coupling relationship between various motion directions, which will cause the motion control of various directions to interfere with one other and significantly affect the control effect. This study proposes a new decoupling motion control algorithm based on the robot attitude calculation for an underwater spherical robot designed for offshore, shallow water, and narrow terrain. The proposed method uses four fuzzy proportional–integral–derivative (PID) controllers to independently control the robot's movement in all directions. Experiments show that the motion control algorithm proposed in this study can significantly improve the flexibility and accuracy of the movement of underwater spherical robots.

1. Introduction

The ocean is a huge treasure house of resources. Conventional marine resource exploitation depends on artificial diving operations, which have various drawbacks [1,2]. First, there is a strict requirement for divers' physical aptitude and professional level, and only a few divers can explore complex caves. Second, the necessity of carrying oxygen supply devices severely limits the time and scope of underwater operations, as well as their flexibility [3]. Third, diving operations expose divers to significant risks, increasing the risk of shallow water decompression sickness [4–6].

Because of the drawbacks of artificial diving operations, various countries have developed underwater robots to replace traditional artificial diving operations. Large autonomous underwater vehicle (AUV) research and development are currently at a fairly advanced stage, however, the flexibility of large AUV movement is still low. In addition, large AUVs have a sonar scanning system and a high-power drive system [7–9]. These characteristics make large AUVs only suitable for deep sea and pelagic environments but are incompatible with shallow water, offshore, and narrow space environments. Consequently, a small-sized, highly flexible underwater robot is urgently needed.

Most underwater robots are designed to be closed and smooth in shape because the open-frame nonstreamlined design is easily entangled by water and grass, which affects the robot's safety. Concurrently, the symmetrical design structure provides better resistance to

the environmental disturbance caused by the disturbance of underwater undercurrent. Thus, spherical underwater robots have emerged as the circumstances require. The spherical structure has some unique properties over other types, such as higher motion stability, easy and rapid situation recovery when the robot collides with obstacles [10,11], and the ability to follow the path with the least resistance [12]. Meanwhile, the spherical robot has the largest volume and highest accommodation than other robots of the same size. It is most suitable for the transition from large to small robots and has both the advantages of the two robot types.

Spherical underwater robots mostly adopt a lightweight design, small size, high symmetry, and smooth surface. This allows them to move in any direction in three-dimensional space, and operate in complex environments with narrow and full water plants, which has expanded the current operating range of underwater robots and has developed into a crucial tool for replacing artificial diving operations [13,14]. Spherical underwater robots mostly adopt miniaturized designs due to structural design and driver output limitations, and their carrying capacity, number of sensors, and processor performance cannot be compared with more advanced large-scale AUVs or land robots [15–19]. Consequently, high-performance control algorithms and target recognition and tracking systems currently used in large-scale AUVs or land robots cannot be applied to spherical underwater robots [15,20–22]. These factors make the ability of independent underwater movement a drawback that restricts the further development

* Corresponding author.

E-mail address: shiliwei@bit.edu.cn (L. Shi).

<https://doi.org/10.1016/j.birob.2022.100067>

Received 14 June 2022; Received in revised form 7 August 2022; Accepted 22 August 2022

Available online 6 September 2022

2667-3797/© 2022 The Author(s). Published by Elsevier B.V. on behalf of Shandong University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).



Fig. 1. Schematic of an underwater robot working in shallow water terrain. The serial number indicates the robot's working environment, while the black box represents the task requirements.

of spherical underwater robots. Therefore, the key to the current research on spherical underwater robots is how to implement efficient motion control algorithms under unfavorable conditions, such as the loading capacity, the number of sensors, and the limited processor performance, to reduce motion control errors and improve the robot's target recognition and tracking capabilities to enhance the robot's autonomous decision-making and reduce the risk of the dependence on manual operation [23]. The development of spherical underwater robots has advanced in several ways, as evidenced by Groundbot [24], Salamander [25], BYSQ-2 [26], and SHU [27]. These spherical underwater robots have good environmental adaptability but are limited by the volume, power consumption, power computation complexity, and amphibious environment factors [28].

Compared with the application environment of large robots, small robots have more restrictions, such as narrow karst caves, which not only limit the space movement but also form undercurrent. Therefore, higher precision requirements are necessary for the motion control algorithms for small robots. The motion control algorithm is crucial to the industrial application of the spherical robot [29]. The main types of underwater control methods currently used are gravity pendulum-based propulsion and water-jet propulsion. The gravity-pendulum-based control method requires driving the robot shell to rotate at high speed, and the viscous resistance of the water body is used to propel the robot forward. However, this control method applies only to the water surface and cannot be used in the underwater environment [30,31] due to the movement's nonlinearity and cross-coupling in a complex underwater environment, as well as the marine environment [32]. Moreover, the rotating shell makes it difficult to install external expansion modules or operating platforms [33]. The jet-type-based control method is suitable for the water and underwater environments [34], with stronger environmental adaptability, however, the existing underwater motion control method or a fixed driver model [35] cannot adjust the thrust direction in real-time [36], or in certain kinds of drive model switching between different periods [37,38]. These control methods are not for each direction of movement and rotation [39] and are independent of real-time control ability [40], and this ability is crucial for the spherical underwater robot to shoot [41], grab [42], follow [43,44] and perform various tasks. A spherical underwater vehicle motion control method with the ability of real-time and independent control of each motion and rotation direction is urgently required for the needs of ocean exploration in shallow water terrains, as shown in Fig. 1 [45,46] (see Fig. 1).

Because of the low motion control precision of spherical underwater robots and strong coupling between the direction of motion, this study describes the real-time attitude-decoupling control algorithm for spherical underwater robots, which uses four fuzzy PID controllers

for X, Y, and Z axes, as well as the course angle movement. Because any assignment's trajectory in the XOY plane can be divided into straight line, steering, and curvilinear motion, experiments of yaw angle control and linear motion control are designed to evaluate single-variable control effect, and experiments of rectangle motion control and orientation are designed to evaluate multivariable control effect. It is established that the real-time attitude-decoupling control method may increase the precision and adaptability of the motion control of the spherical underwater vehicle compared with the previous H-type control algorithm.

2. Spherical underwater robot platform

Fig. 2(a) shows the structure of the robot platform used in this study. The robot is divided into two hemispherical shells, each measuring 30 cm in diameter with a total mass of 6.6 Kg. The upper hemispherical shell contains a sealed cabin and a water inlet cabin for balancing gravity and buoyancy. The sealed cabin is equipped with a robot motherboard, power control module, brushless motor driver, and NVIDIA Jetson TK1 as the robot's main processor. There are 12 high-precision pressure sensors evenly distributed on the surface, and the robot's depth and relative water flow speed are measured by the average and difference of each sensor's pressure data. The water inlet tank is used to balance the robot's buoyancy and gravity in water. A counterweight and buoyancy block is installed in the water tank to accurately balance the robot's gravity and buoyancy in water. In addition, a small hydroacoustic communication device is installed in the water inlet tank to transmit control signals and robot state data.

The robot's lower hemisphere is equipped with a detachable battery compartment and four composite driving legs with amphibious motion capabilities. The battery compartment is waterproof and can provide the robot with power for more than 1.5 h of continuous operation. Fig. 2(b) shows the structure of the robot's compound-driving leg. Each leg has 3 degrees of freedom and is driven by 3 waterproof-steering gears. A leg-vector water-jet compound propeller is installed at the end of the compound-driving leg. The robot's lower hemispherical shell is an open structure, comprising two quarter-spherical shells that can be opened and closed. The robot can freely switch the opening and closing states of the lower hemispherical shell based on the surrounding environment and task conditions.

2.1. Electrical system

Fig. 3 shows the main functional equipment and structural block diagrams of the circuit and control system of an ASR-IV spherical underwater robot. The ASR-IV circuit system adopts a modular design,

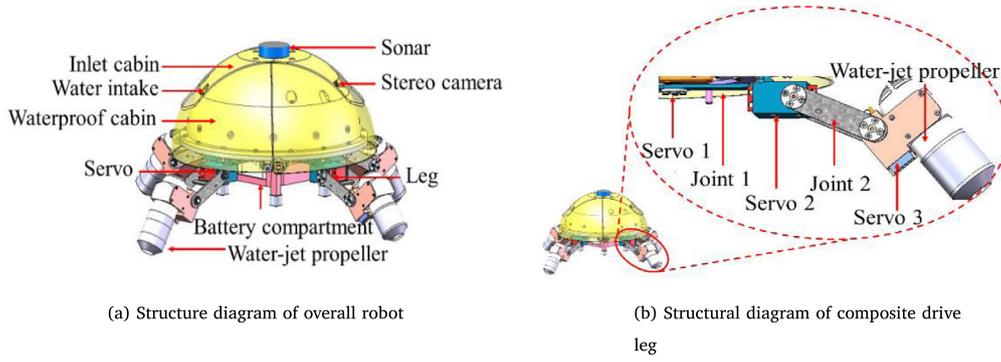


Fig. 2. Schematic diagram of the mechanical structure of an ASR-IV spherical underwater robot.

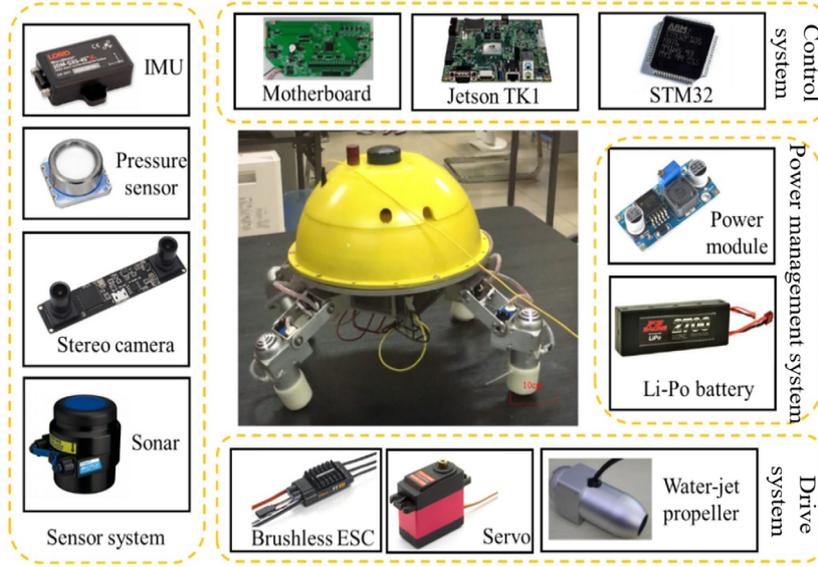


Fig. 3. ASR-IV circuit and control system structure block diagram.

mainly including power management, drive, sensor, and control systems. Each system's modules are independent of each other on the line, and optocoupler isolation measures are taken.

Fig. 4 shows the circuit and control system structure block diagram of ASR-IV. The power management system consists mainly of 3 groups of 2S Li-Po battery packs and power management modules. Each Li-Po battery pack has a rated voltage of 7.4 V and a rated capacity of 4400 mAh, which powers the drive, sensor, and control systems. The control system is composed of a robot motherboard, STM32 and Jetson TK1. The robot's motherboard provides installation platforms and wiring connections for various circuit components. The sensor system consists of an inertial measurement unit (IMU), an underwater acoustic communication module, a binocular camera, and a 12-channel pressure sensor. The driving system provides the robot with the ability to walk on the ground and move downstream. The robot's lower hemispherical shell is equipped with 4 composite driving legs, and each leg has 3 movable joints and a leg-vector water jet. A GWD SHARK waterproof-steering gear is used to drive the movable joints. The leg-vector water jet is equipped with a brushless motor, which is driven by a Hobby Platinum 30 A brushless ESC.

3. Decoupling motion control algorithm

3.1. Kinematics modeling

This section establishes the world and robot coordinate systems (Fig. 5) to obtain the spherical underwater robot's underwater motion

law and improve its underwater motion control performance. The world coordinate system is used to represent information such as the robot's position and posture, whereas the robot coordinate system is used as the reference coordinate for the robot motion control.

In the aforementioned world coordinate system, the robot's position and attitude vectors can be expressed as Eq. (1).

$$\eta = [\eta_1 \quad \eta_2]^T = [x \quad y \quad z \quad \psi \quad \theta \quad \Phi]^T \quad (1)$$

In the above formula, $\eta_1 = [x \quad y \quad z]^T$ represents the robot's three-dimensional position coordinates of the X_w , Y_w , and Z_w axes in the world coordinate system and $\eta_2 = [\psi \quad \theta \quad \Phi]^T$ represents the robot's attitude angle (heading, pitch, and roll angles). The robot's underwater kinematic equation can be obtained as Eq. (2) by analyzing the underwater 6-degree-of-freedom motion of the robot.

$$(M_{RB} + M_A)\dot{v} + (C_{RB}(v) + C_A(v))v + (D_l + D_q(v))v + g(\eta) = \tau \quad (2)$$

where v represents the robot's underwater motion speed and \dot{v} represents the robot's underwater motion acceleration. $M_{RB} \in R^{6 \times 6}$ represents the robot's rigid body mass matrix. $M_A \in R^{6 \times 6}$ represents the robot's additional mass matrix, $C_{RB}(v) \in R^{6 \times 6}$ represents the robot rigid body's Coriolis force matrix, which is generated by the earth rotation, $C_A v \in R^{6 \times 6}$ represents the Coriolis force of the robot's additional mass, and $D_l \in R^{6 \times 6}$ and $D_q(v) \in R^{6 \times 6}$ respectively represent the linear and nonlinear damping matrices of the water when the robot moves in the water. $g(\eta) \in R^{6 \times 6}$ represents the robot's restoring force and moment matrix and $\tau \in R^{6 \times 6}$ represents the thrust and moment matrix generated by the robot driver.

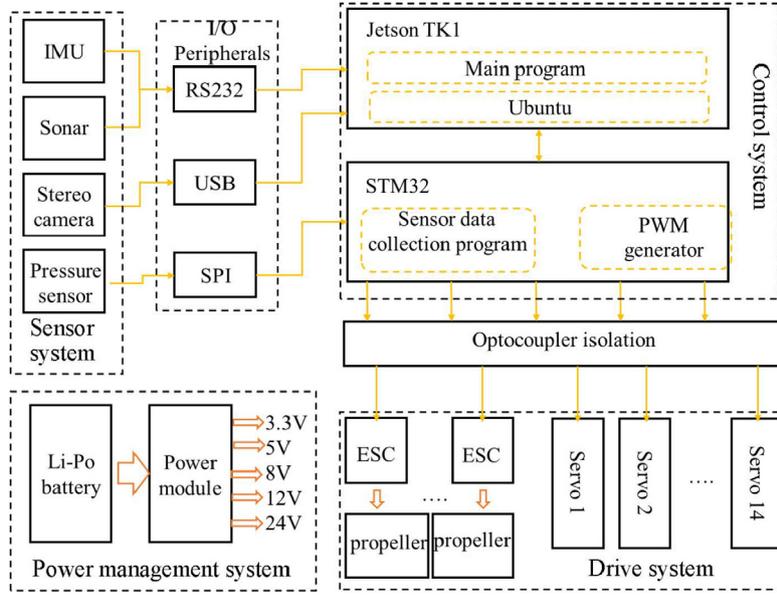


Fig. 4. Structure block diagram of ASR-IV circuit and control system. The circuit system mainly includes power management, drive, sensor, and control systems. Each system's modules are independent of each other on the line, and optocoupler isolation measures are taken.

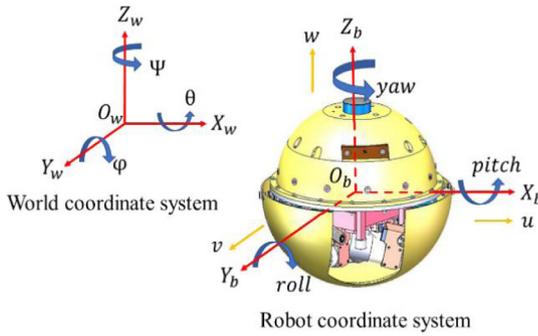


Fig. 5. Coordinate system. Where $O_w - X_w Y_w Z_w$ is the world coordinate system and $O_b - X_b Y_b Z_b$ is the robot coordinate system. The coordinate origin, O_b , is at the robot's center of gravity.

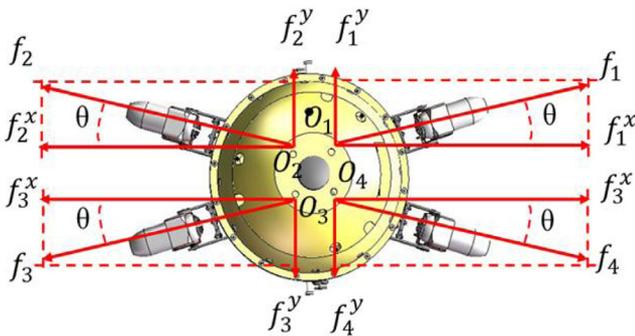


Fig. 6. XOY motion model. The robot's four driving legs are symmetrically distributed in the center, and the rotation angles, θ , are equal, where O_1, O_2, O_3 , and O_4 are the projection positions of the compound-driving legs.

3.2. XOY motion control

Because the robot model is relatively complex, some parameters are difficult to obtain and have little influence on motion control; thus, the model should be simplified appropriately. The ability to independently control the robot's three-dimensional speed and attitude angle is crucial

for ensuring movement flexibility. Fig. 6 shows the XOY motion model, which only considers the thrust of the propeller on the X and Y axes. It is a five-dimensional space composed of the control volume of the robot's XOY motion control process. Suppose the control quantity is K , Eq. (3) is obtained as follows:

$$K = [f_1 \quad f_2 \quad f_3 \quad f_4 \quad \theta]^T \quad (3)$$

where f_1, f_2, f_3 , and f_4 are the thrusts generated by the propeller in the water flow. The thruster received by the robot is the reaction force and θ is the propeller's horizontal angle. Considering the total force of the X and Y axes and the moment of rotation around the Z-axis when the robot is moving underwater, Eq. (4) can be obtained as follows:

$$\begin{cases} -f_1^x + f_2^x + f_3^x - f_4^x = F^x \\ f_1^y + f_2^y - f_3^y - f_4^y = F^y \\ [(-f_1^x + f_2^x - f_3^x + f_4^x) + (f_1^y - f_2^y + f_3^y - f_4^y)] \cdot d = F^{yaw} \end{cases} \quad (4)$$

where F^x, F^y , and F^{yaw} represent the thrust of the robot's X-axis motion, the thrust of the robot's Y-axis motion, and the torque of rotation around the Z-axis, respectively. d represents the distance from the driving leg's mounting point to the robot's center of mass. From the model shown in Fig. 7, the system of Eq. (4) can be simplified to Eq. (5) as follows:

$$\begin{cases} (-f_1 + f_2 + f_3 - f_4) \cdot \cos\theta = F^x \\ (f_1 + f_2 - f_3 - f_4) \cdot \sin\theta = F^y \\ (-f_1 + f_2 - f_3 + f_4) \cdot (\cos\theta - \sin\theta) \cdot d = F^{yaw} \end{cases} \quad (5)$$

f_1, f_2, f_3 , and f_4 are the thrusts produced by the four thrusters on the XOY plane. Through observation, this equation system is found to be an indeterminate system of equations. Set f_4 as a free unknown number, and the solution of the system of Eq. (5) can be obtained as Eq. (6), as follows:

$$\begin{cases} f_1 = \frac{1}{2} \cdot \left(\frac{F^y}{\sin\theta} - \frac{F^{yaw}}{\cos\theta - \sin\theta} \right) \\ f_2 = \frac{1}{2} \cdot \left(\frac{F^x}{\cos\theta} + \frac{F^y}{\sin\theta} \right) \\ f_3 = \frac{1}{2} \cdot \left(\frac{F^x}{\cos\theta} - \frac{F^{yaw}}{\cos\theta - \sin\theta} \right) \\ f_4 = 0 \end{cases} \quad (6)$$

f_1, f_2, f_3 , and f_4 solved in the above formula are not the optimal control quantities for the robot's underwater motion for two reasons.

First, $f_1, f_2, f_3,$ and f_4 have negative numbers, and the thrusters installed on the ASR-IV cannot produce reverse thrust. Second, since most of the robot's power consumption is used to operate the thrusters when the robot is moving in the water, it is necessary to reduce the thrust of each thruster as much as possible to minimize this power consumption. Reduced power consumption, load on circuit parts and thrusters, failure rate, and negative impacts of nonlinearity on the robot motion control due to the thruster approaching the thrust saturation zone can all be achieved by reducing the thrust of the thrusters.

Eq. (6) must be adjusted to obtain the optimal control amount of the robot underwater. Therefore, f_0 is constructed as shown in Eq. (7):

$$f_0 = \min \left\{ \begin{array}{l} \frac{1}{2} \cdot \left(\frac{F^y}{\sin\theta} - \frac{F^{yaw}}{\cos\theta - \sin\theta} \right) \\ \frac{1}{2} \cdot \left(\frac{F^x}{\cos\theta} + \frac{F^y}{\sin\theta} \right) \\ f_3 = \frac{1}{2} \cdot \left(\frac{F^x}{\cos\theta} - \frac{F^{yaw}}{\cos\theta - \sin\theta} \right) \\ 0 \end{array} \right\} \quad (7)$$

Eq. (6) can be transformed into Eq. (8) as follows:

$$\begin{cases} f_1 = \frac{1}{2} \cdot \left(\frac{F^y}{\sin\theta} - \frac{F^{yaw}}{\cos\theta - \sin\theta} \right) - f_0 \\ f_2 = \frac{1}{2} \cdot \left(\frac{F^x}{\cos\theta} + \frac{F^y}{\sin\theta} \right) - f_0 \\ f_3 = \frac{1}{2} \cdot \left(\frac{F^x}{\cos\theta} - \frac{F^{yaw}}{\cos\theta - \sin\theta} \right) - f_0 \\ f_4 = -f_0 \end{cases} \quad (8)$$

To obtain the optimal solution of thrust, it is necessary to minimize the maximum value of $f_1, f_2, f_3,$ and f_4 to obtain the minimum thrust combination and successfully reduce the power consumption of the robot's underwater motion control. Therefore, the free unknown θ in Eq. (8) can be transformed into Eq. (9) as follows:

$$\text{minimize} \left\{ \max \left\{ \begin{array}{l} \frac{1}{2} \cdot \left(\frac{F^y}{\sin\theta} - \frac{F^{yaw}}{\cos\theta - \sin\theta} \right) - f_0 \\ \frac{1}{2} \cdot \left(\frac{F^x}{\cos\theta} + \frac{F^y}{\sin\theta} \right) - f_0 \\ \frac{1}{2} \cdot \left(\frac{F^x}{\cos\theta} - \frac{F^{yaw}}{\cos\theta - \sin\theta} \right) - f_0 \\ -f_0 \end{array} \right\} \right\} \quad (9)$$

Because of the unusually complex nature of Eq. (9), it might be challenging to calculate its analytical solution using standard derivative computation techniques. The numerical calculation method can be used to solve the approximate solution of θ iterative calculation under the condition of controllable error.

3.3. Z Axis motion control

Establish the model shown in Fig. 7 to represent the thrust generated by the thruster on the Z-axis. Eq. (10) illustrates the mechanical relationship derived from the graphical model:

$$\begin{cases} f_1^z + f_2^z + f_3^z + f_4^z = F^z \\ f_1^z - f_2^z - f_3^z + f_4^z = \frac{F^{pitch}}{d} \\ f_1^z + f_2^z - f_3^z - f_4^z = \frac{F^{roll}}{d} \end{cases} \quad (10)$$

where $F^z, F^{pitch},$ and F^{roll} represent the thrust of the robot along the Z-axis direction, the torque rotating around the Y-axis, and the torque rotating around the X-axis, respectively. d represents the distance from the far end of the thruster to the robot's center of mass. A stable value of d is used to simplify the equation because the change in d caused by the rotation of the robot driving leg is minimal. F^{pitch} and F^{roll} are the respective control variables of the pitch and roll angles of the robot moving in the water. Practically, because the robot's center of gravity and center of buoyancy are on the same vertical line, with the center of gravity located below the center of buoyancy, the robot can automatically stabilize the pitch and roll angles and maintain a horizontal posture. Therefore, the control model can ignore the power consumption of the robot's underwater motion, implying that $F^{pitch} = 0$

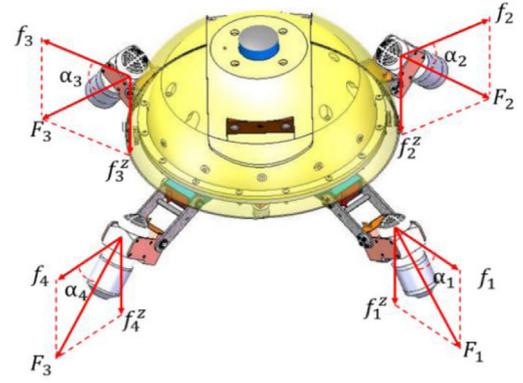


Fig. 7. Z-axis motion model. Where $F_1, F_2, F_3,$ and F_4 are the total thrust generated by the robot's four thrusters. $f_1^z, f_2^z, f_3^z,$ and f_4^z are the component forces of the total thrust generated by the robot's four thrusters in the Z-axis direction, and the thrust received by the robot in the Z-axis direction is its reaction force. $\alpha_1, \alpha_2, \alpha_3,$ and α_4 are the angles between the robot's four thrusters and the XOY plane.

and $F^{roll} = 0$. Therefore the system of Eqs. (10) can be simplified into the following system of equations:

$$\begin{cases} f_1^z + f_2^z + f_3^z + f_4^z = F^z \\ f_1^z - f_2^z - f_3^z + f_4^z = 0 \\ f_1^z + f_2^z - f_3^z - f_4^z = 0 \end{cases} \quad (11)$$

Similar to the XOY plane motion control model, it is necessary to minimize $\{f_1^z, f_2^z, f_3^z, \text{ and } f_4^z\}$ to prevent the thruster from entering the thrust saturation zone.

$$f_1^z = f_2^z = f_3^z = f_4^z = \frac{1}{4} F^z \quad (12)$$

3.4. Fuzzy PID controllers design

To realize the independent control of the robot's X, Y, and Z axes in a three-dimensional space, the robot's underwater motion control quantity must fall at a point in the vector space composed of $[F_1, F_2, F_3, F_4, \theta, \alpha_1, \alpha_2, \alpha_3, \text{ and } \alpha_4]$. From Eqs. (3) and (12), the final output control quantity of the underwater robot's three-dimensional motion control algorithm is given as follows:

$$K = \begin{bmatrix} \sqrt{f_1^2 + (f_1^z)^2} \\ \sqrt{f_2^2 + (f_2^z)^2} \\ \sqrt{f_3^2 + (f_3^z)^2} \\ \sqrt{f_4^2 + (f_4^z)^2} \\ \theta \\ \arctan \frac{f_1}{f_1^z} \\ \arctan \frac{f_2}{f_2^z} \\ \arctan \frac{f_3}{f_3^z} \\ \arctan \frac{f_4}{f_4^z} \end{bmatrix} \quad (13)$$

Fig. 8 shows the structure of a fuzzy PID controller. The control quantity is different when the four controllers are used to control the robot's motion on the X, Y, and Z axes, and the heading angle.

4. Experiments

We conducted underwater motion control experiments on the ASR-IV spherical underwater robot platform to test the performance of the

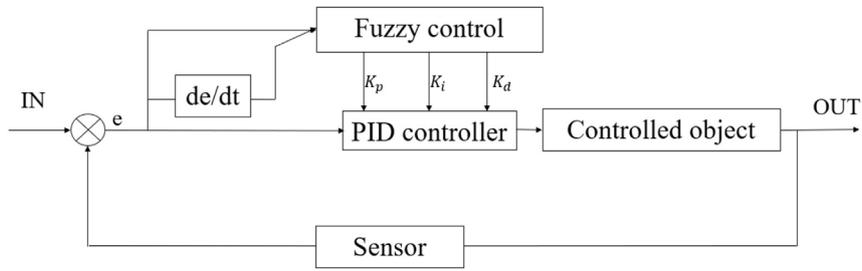


Fig. 8. Structure of the fuzzy PID controller. The control quantities are different when the controllers control the X-axis, Y-axis, Z-axis, and heading angle. k_p , k_i , and k_d represent the scale factor, integral coefficient, and differential coefficient, respectively.

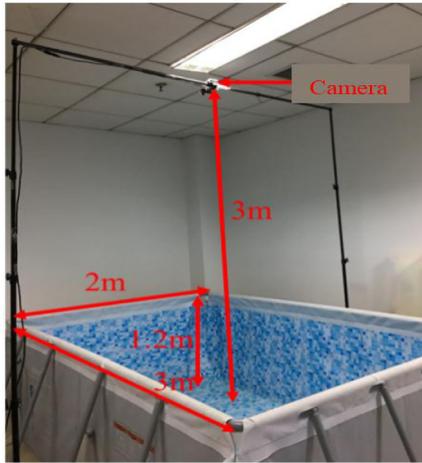


Fig. 9. Experimental environment. The pool has a water depth of 1 m and a global camera is installed 3-m directly above the center of the pool. The size of the pool is 3 m × 2 m × 1.2 m.

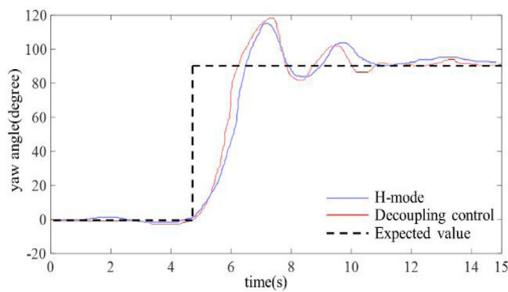


Fig. 10. Result of yaw angle control experiment. “H” mode control algorithm has a rise time of $t_{r1} = 1.12$ s, peak time of $t_{p1} = 1.68$ s, and no steady error, whereas the decoupling control algorithm has a rise time of $t_{r2} = 1.10$ s, peak time of $t_{p2} = 1.69$ s, and no steady error.

decoupling control algorithm proposed in this study, and the experimental results of the proposed algorithm were compared with the underwater multimotion mode control algorithm under the same conditions. This experiment was conducted in a pool of 3 m × 2 m × 1.2 m (Fig. 9). The robot’s submergence depth in this experiment is 1 m, where the actual design of the maximum submergence depth of the robot is 3–4 m. The robot is identified and positioned through computer vision algorithms and the coordinates of a two-dimensional plane are calculated and recorded.

4.1. Yaw angle control experiment

The ASR-IV spherical underwater robot is equipped with a 3dm-gx4-45 IMU from MicroStrain. The module has a built-in nine-axis sensor

and uses the EKF algorithm to output the robot’s attitude estimation value, which includes the robot’s heading angle. As an inertial device, the cumulative error is inevitable, however, because the filter module is preset in the selected model, the cumulative error is significantly lower than it would be without the filter module. In addition, since the total time of using IMU during the experiment process is short, the cumulative error can be ignored and no additional filters are required to minimize the error. The selected IMU model has a magnetic field sensor that can calibrate the heading angle. However, because it is installed in the robot’s geometric center and is close to the thruster, it is susceptible to the magnetic interference of the thruster motor, which may result in a heading angle measurement error. In the experiment, the calibration function of the heading-angle magnetic field is turned off and the initial value of the heading angle must be calibrated manually. The experimental process is as follows:

First, subtract the initial value of the heading angle from the current value to obtain the heading angle under the tank coordinate system. Subsequently, the thrust of the robot’s X, Y, and Z axes in the still water state is calculated using the PID algorithm. The initial expected value of the heading angle is 0°, whereas the expected value of the heading angle is set as 90° in the 5th second. Then, the final output control quantity, \vec{K} , is calculated using the decoupling motion control algorithm based on real-time attitude calculation, and the drive system is controlled to output the corresponding angle and thrust. Finally, record the experimental data and compare it with the “H” mode in the multimotion mode control algorithm.

The analysis of the experimental results in Fig. 10 shows that both methods do not have a steady error, and the “H” mode control algorithm has a rise time of $t_{r1} = 1.12$ s and a peak time of $t_{p1} = 1.68$ s, whereas the decoupling control algorithm has a rise time of $t_{r2} = 1.10$ s and a peak time of $t_{p2} = 1.69$ s. The results showed that the performance of the “H” mode control algorithm and the decoupling control algorithm alone in the heading angle control are consistent when measurement error and system noise are considered.

4.2. Linear motion control experiment

In the linear motion control experiment, two mutually perpendicular linear paths are set for the robot. The first section is from point A (0.45 m, 0.1 m) to point B (0.45 m, 1 m), while the second section is from point B to point C (3 m, 1 m). The robot’s motion trajectory and heading angle information are recorded using the camera above the pool and the motion trajectory error is calculated. The experimental results of the decoupling control algorithm based on real-time attitude calculations are compared with those of the multimotion mode control algorithm. Fig. 11 shows the process of the two sets of experiments. Fig. 11(a) and (b) respectively show the trajectory data and experimental error data of the two methods.

The robot goes through the entire range in 15 s when using both algorithms. The robot uses the “H” motion mode in the AB section and the “X” motion mode in the BC section when using the multimotion mode control algorithm. The motion trajectory reaches the

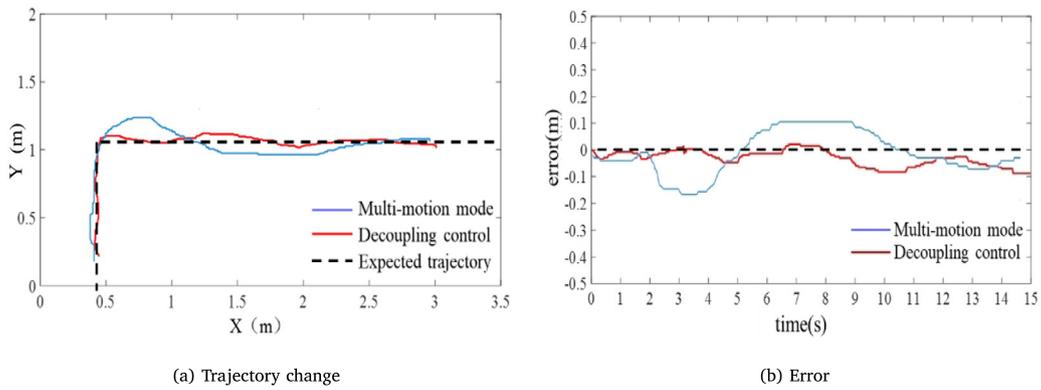


Fig. 11. Results of linear motion control experiment. The robot uses the “H” motion mode in the AB section and the “X” motion mode in the BC section when using the multimotion mode control algorithm, and the maximum error of the motion trajectory is 0.19 m. When using the decoupling control algorithm, the maximum error of the motion trajectory is 0.1 m.

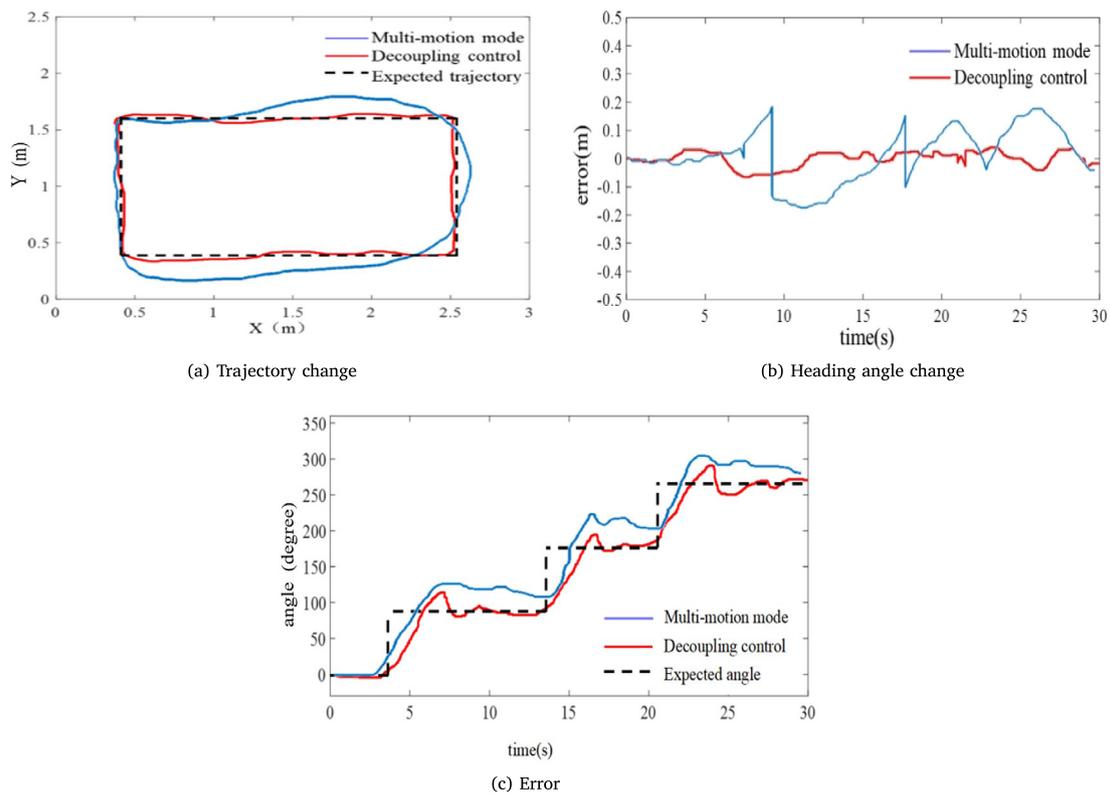


Fig. 12. Results of the rectangular motion control experiment. The maximum trajectory error is 0.23 m when using the multi-motion mode control algorithm. The maximum error of the decoupling motion control algorithm is 0.14 m and the heading-angle fluctuation is smaller than that of the multimotion mode control algorithm.

maximum at the inflection point B attachment, with a maximum error of 0.19 m. When using the decoupling control algorithm, the robot adjusts the layout and thrust of the four composite driving legs in real-time based on the current position, and the motion trajectory’s maximum error is 0.1 m. Experimental analysis shows that the spherical underwater robot’s real-time attitude calculation and decoupling control algorithm proposed in this study significantly improved control accuracy compared with the multimotion mode control algorithm.

4.3. Rectangular motion control experiment

The rectangular motion is a sequence combination of linear motion and heading-angle control. In the rectangular motion control experiment, four target points were set for the robot, such as A (0.4 m, 0.4 m), B (0.4 m, 2.6 m), C (2.6 m, 1.6 m), and D (0.4 m, 1.6 m). Let the robot

keep moving in a straight line through the four points ABCD and keep the robot facing the target direction. Record the robot’s motion trajectory and heading angle, and conduct a comparative analysis of the two algorithms. Figs. 12(a) and (b) show the motion control trajectory and motion control error, respectively. The robot completes the full range in about 30 s. The maximum trajectory error when using the multimotion mode control algorithm is 0.23 m. The position of the maximum error is consistent with the linear motion experiment because the trajectory turning point must be controlled by the heading angle to correct the lateral drift, which appears near the inflection point where the heading angle changes dramatically, as shown in Fig. 12(c). Combined with the results of the straight-line experiment, this control method can achieve a steady state during straight-line motion. However, the control quantity did not reach the expected value within 7.5 s after turning 90°. We can infer that the steady state was not precisely realized.

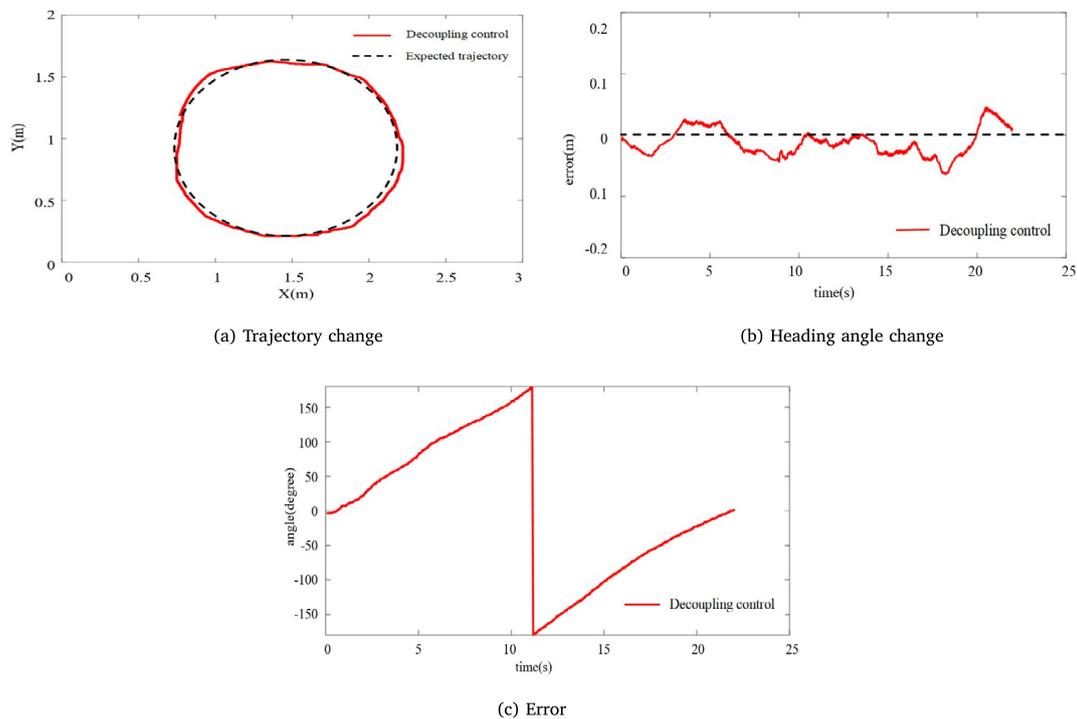


Fig. 13. Results of orientation experiment. The orientation experiment cannot be accomplished with the multimotion mode control algorithm. The experiment can be accomplished using decoupling control, with a maximum trajectory control error is 0.07 m.

The maximum error of the decoupling motion control algorithm is 0.14 m and the heading-angle fluctuation is minimal. Therefore, the experimental comparison demonstrates that the control system's steady-state time is longer and its resistance to abrupt change variables is weak. The overshoot is larger, the rise time is longer, and the steady time is longer compared with the decoupling algorithm. The experiments prove that the decoupling motion control algorithm is superior.

4.4. Orientation experiment

A circular trajectory with a center at (1.5 m, 1 m) and a radius of 0.7 m was set for the robot in the orientation experiment, and the actual trajectory and heading angle change of the robot were recorded. Fig. 13(a), (b), and (c) show the motion trajectory, control error, and heading-angle change, respectively, during the experiment. The experimental results show that the maximum trajectory control error is 0.07 m and the robot maintains a constant speed motion, while its heading angle keeps changing at a constant speed, and its head always points toward the center of the circle.

5. Conclusion

This study realizes the objective of replacing divers with spherical underwater robots to achieve autonomous operations in offshore, shallow water, and narrow terrain, as well as to improve the autonomous movement capabilities of spherical underwater robots. An ASR-IV spherical underwater robot platform is used in this study to investigate a decoupling control algorithm suitable for underwater environments and embedded platforms. First, the kinematic modeling of the spherical underwater robot is completed, and the model constructs the mathematical relationship between the speed, acceleration, and three-dimensional thrust of the propeller in the three-dimensional underwater movement process. Then, the robot's underwater X, Y, and Z axes motion, and heading-angle control are decomposed into four independent control variables based on the kinematic model using a set of control strategies for real-time calculation of the propeller

attitude layout. Consequently, the coupling relationship between each movement direction is broken. Further, four independent fuzzy PID controllers designed to realize real-time independent control of the robot's movement in four directions are achieved.

Experimental results show that the proposed algorithm achieves a linear motion control with a maximum error of 0.1 m and a control accuracy increase of 47.4% compared with the multimotion mode algorithm. In addition, it performs a rectangular motion with a maximum error of 0.14 m and increases control accuracy by 39.1% compared with the multimotion mode algorithm. Further, the proposed algorithm can control the directional target orbiting motion, which cannot be performed by the multimotion mode control algorithm, with a maximum control error of 0.07 m. The experimental results prove that the proposed algorithm has higher accuracy and flexibility when controlling the motion of spherical underwater robots.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by National Natural Science Foundation of China (Grant Nos. 61773064, 61503028).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.birob.2022.100067>.

References

- [1] W.S. Chu, K.T. Lee, S.H. Song, M.W. Han, J.Y. Lee, H.S. Kim, M.S. Kim, Y.J. Park, K.J. Cho, S.H. Ahn, Review of biomimetic underwater robots using smart actuators, *Int. J. Precis. Eng. Manuf.* 13 (7) (2012) 1281–1292, <http://dx.doi.org/10.1007/s12541-012-0171-7>.
- [2] S. Oshiro, Y. Sakuma, R. Chibana, A. Kinjo, T. Wada, A prototype wireless underwater robot control system using a 32 kHz bandwidth Underwater Small Area acoustic network (USAAN), in: *OCEANS 2019 MTS/IEEE SEATTLE*, 2020.
- [3] J.M. Maquire, H. Peng, S. Bai, *Dynamic Analysis and Control System Design of Automatic Transmissions*, SAE International, 2013.
- [4] M. Hubbard, F.M. Davis, K. Malcolm, S.J. Mitchell, Decompression illness and other injuries in a recreational dive charter operation, *J. South Pac. Underw. Med. Soc.* 48 (4) (2018) 218–223, <http://dx.doi.org/10.28920/dhm48.4.218-223>.
- [5] J.M. Longphre, P.J. Denoble, R.E. Moon, R.D. Vann, J.J. Freiburger, First aid normobaric oxygen for the treatment of recreational diving injuries., *Undersea Hyperb. Med.* 34 (1) (2007) 43–49, [http://dx.doi.org/10.1016/S0020-1693\(01\)00567-9](http://dx.doi.org/10.1016/S0020-1693(01)00567-9).
- [6] P. Beale, L. Kitchen, W.R. Graf, M.E. Fenton, Abdominal decompression illness following repetitive diving: a case report and review of the literature, *Undersea Hyperb. Med.: J. Undersea Hyperb. Med. Soc., Inc* 46 (2) (2019) 211–215, <http://dx.doi.org/10.22462/04.06.2019.16>.
- [7] P. Wadhams, J.P. Wilkinson, A. Kaletsky, Sidescan sonar imagery of the winter marginal ice zone obtained from an AUV, *J. Atmos. Ocean. Technol.* 21 (9) (2010) 1462–1470, [http://dx.doi.org/10.1175/1520-0426\(2004\)021.0.CO;2](http://dx.doi.org/10.1175/1520-0426(2004)021.0.CO;2).
- [8] L. Cai, Q. Sun, T. Xu, Y. Ma, Z. Chen, Multi-AUV collaborative target recognition based on transfer-reinforcement learning, *IEEE Access PP* (99) (2020) 1, <http://dx.doi.org/10.1109/ACCESS.2020.2976121>.
- [9] L. Liu, Z. Xi, J. Wu, Strengthening the achilles' heel: An AUV-aided message ferry approach against dissemination vulnerability in UASNs, *IEEE Internet Things J. PP* (99) (2021) 1, <http://dx.doi.org/10.1109/JIOT.2021.3072920>.
- [10] Q. Zhan, T. Zhou, M. Chen, S. Cai, Dynamic trajectory planning of a spherical mobile robot, in: *2006 IEEE Conference on Robotics, Automation and Mechatronics, IEEE*, 2006, pp. 1–6.
- [11] A. Halme, T. Schonberg, Y. Wang, Motion control of a spherical mobile robot, in: *Proceedings of 4th IEEE International Workshop on Advanced Motion Control-AMC'96-MIE*, Vol. 1, IEEE, 1996, pp. 259–264.
- [12] P. Mojabi, et al., Introducing august: a novel strategy for an omnidirectional spherical rolling robot, in: *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, Vol. 4, IEEE, 2002, pp. 3527–3533.
- [13] M. Li, S. Guo, H. Hirata, H. Ishihara, Design and performance evaluation of an amphibious spherical robot, *Robot. Auton. Syst.* (2015) <http://dx.doi.org/10.1016/j.robot.2014.11.007>.
- [14] Shuxiang, GUO, Study on biometrics moth-son types of multi-functional amphibious spherical robot systems, *J. Soc. Instrum. Control Eng.* 54 (1) (2015) 62–63, <http://dx.doi.org/10.11499/sicej.54.62>.
- [15] G. Shuxiang, P. Shaowu, S. Liwei, G. Ping, H. Yanlin, T. Kun, Visual detection and tracking system for a spherical amphibious robot, *Sensors* 17 (4) (2017) 870, <http://dx.doi.org/10.3390/s17040870>.
- [16] S. Guo, Y. He, L. Shi, S. Pan, X. Rui, K. Tang, G. Ping, Modeling and experimental evaluation of an improved amphibious robot with compact structure, *Robot. Comput.-Integr. Manuf.* 51 (jun.) (2018) 37–52, <http://dx.doi.org/10.1016/j.rcim.2017.11.009>.
- [17] X. Hou, S. Guo, L. Shi, H. Xing, Y. Liu, H. Liu, Y. Hu, D. Xia, Z. Li, Hydrodynamic analysis-based modeling and experimental verification of a new water-jet thruster for an amphibious spherical robot, *Sensors* 19 (2) (2019) <http://dx.doi.org/10.3390/s19020259>.
- [18] L.S.A. B, S.G.A. B, S.M. C, C.Y. C, M.L. C, K.A. D, Development of an amphibious turtle-inspired spherical mother robot, *J. Bionic Eng.* 10 (4) (2013) 446–455, [http://dx.doi.org/10.1016/S1672-6529\(13\)60248-6](http://dx.doi.org/10.1016/S1672-6529(13)60248-6).
- [19] S. Pan, L. Shi, S. Guo, A kinect-based real-time compressive tracking prototype system for amphibious spherical robots, *Sensors (Basel, Switzerland)* 15 (4) (2015) <http://dx.doi.org/10.3390/s150408232>.
- [20] H. Xing, S. Guo, L. Shi, Y. He, S. Su, C. Zhan, X. Hou, Hybrid locomotion evaluation for a novel amphibious spherical robot, *Appl. Ences* 8 (2) (2018) 156, <http://dx.doi.org/10.3390/app8020156>.
- [21] L. Shi, Y. Hu, S. Su, S. Guo, D. Xia, A fuzzy PID algorithm for a novel miniature spherical robots with three-dimensional underwater motion control, *J. Bionic Eng.* 17 (5) (2020) <http://dx.doi.org/10.1007/s42235-020-0087-3>.
- [22] H. Xing, L. Shi, K. Tang, S. Guo, X. Hou, Y. Liu, H. Liu, Y. Hu, Robust RGB-D camera and IMU fusion-based cooperative and relative close-range localization for multiple turtle-inspired amphibious spherical robots, *J. Bionic Eng.* 16 (003) (2019) 442–454, <http://dx.doi.org/10.1007/s42235-019-0036-1>.
- [23] Y. Zhou, B. Zhong, T. Fang, J. Liu, X. Zhou, S. Zhang, Application of bio-inspired control of Amphihex-I in detection of oncomelania hupensis, the amphibious snail intermediate host of schistosoma japonicum, *Ind. Robot: Int. J.* (2017) <http://dx.doi.org/10.1108/ir-05-2016-0151>.
- [24] V. Kaznov, M. Seeman, Outdoor navigation with a spherical amphibious robot, in: *IEEE/RSJ International Conference on Intelligent Robots & Systems*, 2010.
- [25] S. Satria, J.W. Lee, S. Chan, Portable amphibious spherical rolling robot with live-streaming capability for ground and aquatic deployment, 2015.
- [26] X.J. Lan, H.X. Sun, Q.X. Jia, H.Y. Li, Design and optimization of a spherical underwater vehicle BYSQ-2's structure, *J. Beijing Univ. Posts Telecommun.* (2012) <http://dx.doi.org/10.13190/jbupt.201204.11.lanxj>.
- [27] S. Xie, J. Chen, J. Luo, H. Li, J. Gu, Dynamic analysis and control system of spherical robot for polar region scientific research, in: *2013 IEEE International Conference on Robotics and Biomimetics, ROBIO*, 2013.
- [28] V. Kaznov, M. Seeman, Outdoor navigation with a spherical amphibious robot, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.
- [29] S.M. Zanoli, G. Conte, Remotely operated vehicle depth control, *Control Eng. Pract.* 11 (4) (2003) 453–459, [http://dx.doi.org/10.1016/S0967-0661\(02\)00013-8](http://dx.doi.org/10.1016/S0967-0661(02)00013-8).
- [30] I.S. Akkizidis, G.N. Roberts, P. Ridao, J. Batlle, Designing a fuzzy-like PD controller for an underwater robot, *Control Eng. Pract.* 11 (4) (2003) 471–480, [http://dx.doi.org/10.1016/S0967-0661\(02\)00055-2](http://dx.doi.org/10.1016/S0967-0661(02)00055-2).
- [31] G. Istitireni, O.B. Akan, Three-dimensional underwater target tracking with acoustic sensor networks, *IEEE Trans. Veh. Technol.* 60 (8) (2011) 3897–3906, <http://dx.doi.org/10.1109/TVT.2011.2163538>.
- [32] Z.Z. Chu, Adaptive sliding mode control based on local recurrent neural networks for underwater robot, *Ocean Eng.* (2012) <http://dx.doi.org/10.1016/j.oceaneng.2012.02.004>.
- [33] L. Paull, S. Saeedi, M. Seto, H. Li, AUV navigation and localization: A review, *IEEE J. Ocean. Eng.* 39 (1) (2014) 131–149, <http://dx.doi.org/10.1109/JOE.2013.2278891>.
- [34] J. Delcourt, M. Ylieff, V. Bolliet, P. Poncin, A. Bardouet, Video tracking in the extreme: A new possibility for tracking nocturnal underwater transparent animals with fluorescent elastomer tags, *Behav. Res. Methods* 43 (2) (2011) 590–600, <http://dx.doi.org/10.3758/s13428-011-0060-5>.
- [35] T.M. Grothues, J. Dobarro, J. Eiler, Collecting, interpreting, and merging fish telemetry data from an AUV: Remote sensing from an already remote platform, in: *Autonomous Underwater Vehicles*, 2010.
- [36] J. Bosch, N. Gracias, P. Ridao, K. Isteni, D. Ribas, Close-range tracking of underwater vehicles using light beacons, *Sensors* 16 (4) (2016) <http://dx.doi.org/10.3390/s16040429>.
- [37] O.K. Soni, J.S. Kumare, A survey on underwater images enhancement techniques, in: *2020 IEEE 9th International Conference on Communication Systems and Network Technologies, CSNT*, 2020.
- [38] S. Pan, L. Shi, S. Guo, A kinect-based real-time compressive tracking prototype system for amphibious spherical robots, *Sensors (Basel, Switzerland)* 15 (4) (2015) <http://dx.doi.org/10.3390/s150408232>.
- [39] R.L. Baines, J.W. Booth, F.E. Fish, R. Kramer-Bottiglio, Toward a bio-inspired variable-stiffness morphing limb for amphibious robot locomotion, in: *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, 2019.
- [40] L. Zhang, B. He, Y. Song, T. Yan, Consistent target tracking via multiple underwater cameras, in: *OCEANS 2016 - Shanghai*, 2016.
- [41] D. Lee, G. Kim, D. Kim, H. Myung, H.T. Choi, Vision-based object detection and tracking for autonomous navigation of underwater robots, *Ocean Eng.* 48 (none) (2012) 59–68, <http://dx.doi.org/10.1016/j.oceaneng.2012.04.006>.
- [42] M.F. Yahya, M.R. Arshad, Tracking of multiple light sources using computer vision for underwater docking, *Procedia Comput. Sci.* 76 (2015) 192–197, <http://dx.doi.org/10.1016/j.procs.2015.12.339>.
- [43] T. Elamsy, A. Habel, B. Boufama, A new method for linear affine self-calibration of stationary zooming stereo cameras, in: *IEEE International Conference on Image Processing*, 2013.
- [44] R.L. Hartley, Self-calibration from multiple views with a rotating camera, in: *European Conference on Computer Vision*, 1994.
- [45] Q.T. Luong, O.D. Faugeras, Self-calibration of a moving camera from point correspondences and fundamental matrices, *Int'l J. Comput. Vis.* 22 (3) (1997) 261–289, <http://dx.doi.org/10.1023/A:1007982716991>.
- [46] H. Di, S. Qi, S. Wang, A virtual binocular vision range finding method of remote object based on single rotating angle indexing camera, in: *International Conference on Electronic Measurement & Instruments*.