Control of Underwater Robots Based on a BP Neural Network

Miaoqing CHEN

College of Digital Technology and Engineering, Ningbo University of Finance and Economics, Ningbo, Zhejiang, China, 315175
18258778987@163.com

Abstract: As an important engineering tool, underwater robots are widely used in marine science and resource exploration. This paper proposes a BP neural network for the control of underwater robots, which could perform the initialization and online adjustment of control parameters for underwater robots based on a large amount of data related to speed control and heading control. A layout pattern featuring eight thrusters was designed and analysed in order to achieve a six-degree-of-freedom control system for underwater robots, including forward and backward translation, left and right translation and steering functions. In this context, the four vertically positioned thrusters used suction cups to offset the torque caused by the rotation of the internal spiral blades. The obtained experimental results confirmed that the S-surface controller of the BP neural network exhibited an excellent performance as regards the motion control of intelligent underwater robots. It had the capacity to autonomously initialize control parameters and adjust them online, while demonstrating a very high anti-interference ability. At a steady-state speed of 1000 rad·s^{-1}, the obtained signal was mainly composed of sinusoidal components, with a frequency distribution around 5, 25, 50, and 100 Hz. When a fault occurred, a negative sequence component appeared in the analysed signal, with a frequency distribution around 10, 30, 50, and 75 Hz, and its amplitude increased significantly.

Keywords: Underwater robots, BP neural network, Anti-disturbance performance, Sports mode, Intelligent control.

1. Introduction

Underwater Robots (UWRs) have become indispensable tools in fields such as marine science, resource development, environmental monitoring, and military applications. However, efficient control of UWR in complex and ever-changing underwater environments has always been a concern (Christensen et al., 2022). Traditional control methods are constrained by the uncertainty of underwater environments, nonlinear dynamics, and navigation challenges, therefore requiring more intelligent and adaptive measures to address these complex problems (Subad, Cross & Park, 2021). In this context, research has explored UWR control based on Back Propagation Neural Network (BPNN). Unlike traditional methods, BPNN has excellent adaptability and nonlinear modeling ability, which enable it to learn and adapt in real-time dynamic underwater environments (Hasan & Abbas, 2023). The innovation proposed in this paper lies in the combination of BPNN and UWR control, achieving high adaptability and real-time performance to meet the needs required for different underwater tasks and environments. This study also explores how to use BPNN to optimize the motion trajectory planning for UWRs, to improve their motion performance and efficiency, and better adapt to different tasks. This study aims to open up new research directions in the field of UWR, bring about broader possibilities for its application, and promote further development in areas such as deep-sea exploration and resource development.

BPNN is widely used and has been studied by many scholars. Researchers studied and solved the security issues of the Internet of Things, and designed an intrusion detection system based on LM-BPNN to detect malicious attacks. They used LM to optimize the weight threshold of traditional BPNN to improve network performance (Yang et al., 2019). Researchers extracted time-domain and frequency-domain features of rotating machinery under various operating conditions, and constructed feature vectors to describe its mechanical state. Through improved distance assessment, they selected different parameters to reconstruct low-dimensional sensitive feature samples, thereby improving fault diagnosis ability (Lu et al., 2019). Certain studies used improved algorithms to optimize the Multilayer Perceptron (MLP) so as to improve prediction performance and determine the optimal structural parameters of the MLP. They used different landslide and non-landslide data to train and test this model (Li et al., 2019). Certain studies used grey correlation analysis to establish a prediction index system to comprehensively consider multiple influencing factors. They combined BPNN with other networks to form a predictive model to overcome the low accuracy and inability to fully consider all factors of traditional models (Ding et al., 2019). Researchers investigated the

https://doi.org/10.24846/v33i1y202402

effect of process parameters on Printable Bridge Length (PBL) to minimize the use of support materials by maximizing the distance between support points. In the experiment, sample data was used to train networks to predict the nonlinear relationship between PBL and process parameters. These studies confirmed that the established network could accurately predict the longest PBL (Jiang et al., 2019).

Certain studies reviewed the current research status of biomimetic UWRs, particularly in the development of body/tail fins (BCF) and others. They summarized the different motion control methods used in biomimetic UWRs, including open-loop swimming control and closed-loop control strategies, to achieve their special maneuverability (Wang et al., 2022). An et al. (2022) constructed an acoustic communication system with excellent communication performance to improve the performance and efficiency of spherical UWRs (SURs). He developed task planning and collaborative control strategies by adjusting the motion of SURs to ensure that they maintain the desired formation position and shape during formation movement. They proposed a thrust distribution method to develop a remote-controlled submersible that supported underwater exploration and autonomous underwater operations. It could be used to effectively manage eight thrusters for precise position and attitude control. (Lee et al., 2020) designed a feedback controller that utilized sensor data to adjust and control the motion of the ROV in real-time. To achieve autonomous navigation of UWR vehicles in a confined workspace, they developed a robust nonlinear control scheme aimed at guiding UWR vehicles towards specific road points. Heshmati-Alamdari et al. (2019) considered various constraints such as input and state constraints, static obstacles, workspace boundaries, upper speed limits, and thruster saturation (Heshmati-Alamdari, Karras & Marantos, 2019). Certain studies developed a robot that simulated jellyfish, applying closed-loop fuzzy and visual control to enable soft robots to move vertically and horizontally. They imitated the movement of jellyfish and used computer-aided design and computational fluid dynamics simulations to verify the performance and behavior of the robot. These studies confirmed that robots could generate temperature gradients for monitoring in underwater environments (Cruz Ulloa et al., 2020).

In summary, although many scholars have studied BPNN and UWR control, there are still some shortcomings. Some studies tend to adopt traditional BPNN, which has certain limitations in handling complex nonlinear problems and anti-interference ability. Therefore, this research comprehensively considers the cross fusion of BPNN and UWR control in multiple fields, aiming to solve the intelligent control problem of UWR, and has a broader application prospect.

The remainder of this paper is as follows. Section 2 describes the UWR control method based on BPNN. Section 3 presents the validation of UWR control method based on BPNN and Section 4 includes the conclusion of this paper.

2. Research Method

This section focuses on BPNN-based UWR control methods. Firstly, a detailed introduction is given for S-plane control method using BPNN, which is used to achieve motion control of UWR (Afriliana & Ramadhan, 2022). Next, the motion mode and structural scheme of the UWR is presented, including a 6-degree-of-freedom control scheme designed to meet high mobility requirements, including the layout mode of four thrusters.

2.1 S-plane Control Method for UWR Based on BPNN

S-plane controllers are typically used to manage the movement of machines or systems. This study uses BPNN to enhance traditional S-surface controllers to ensure that they accurately reach the expected target state or trajectory. BPNN is introduced to achieve dynamic adjustment of S-plane controller parameters to adapt to different control requirements and environmental conditions. S-plane control adopts a unique control strategy, where the control edges are loose and the central part is dense, to imitate the characteristics of fuzzy control. Its main goal is to achieve precise control of small deviations (Zhao, 2023). In order to achieve this goal, the sigmoid function is used in the experiment to perform nonlinear fitting of fuzzy control rules, thereby constructing an S-surface controller as it is expressed in equation (1):

$$u=2.0/(1.0+e^{-k_1 e-k_2 \dot{e}})-1.0 \quad (1)$$

In equation (1), k_1, k_2 represent the control parameters of S-plane controller. k_1 represents the

proportional gain, which affects the performance of the control system, especially in adjusting the rise time and overshoot amplitude of the control system. k_2 represents differential gain. The coefficient of the differential term will have an impact on the performance of the control system, especially in adjusting the overshoot and stability of the control system. $e, \dot{e}$ represent deviation and its changing rate, respectively (Filaretov et al., 2021). u represents the output signal generated by the control system. When the control input is adjusted to the minimum value, the control output will also be set to its maximum negative value, within $[-1,+1]$ (Yang & Shi, 2020). For fixed interference forces such as ocean currents, the fixed deviation can be eliminated by adjusting the offset of the S-plane, and the control model function in equation (2) can be used:

$$u=2.0/(1.0+e^{-k_1 e-k_2 \dot{e}})-1.0+\Delta u \tag{2}$$

In equation (2), Δu can fix the interference force through adaptive means. Figure 1 shows S-plane control structure of BPNN.

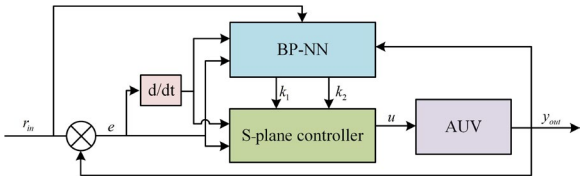


Figure 1. BP neural network S-plane control structure block diagram

As it can be seen in Figure 1, S-plane controller is usually connected to the BPNN, receives its output signal, and converts it into control actions to impact the operation of the system, and adjusts the actual physical system or process through S-plane control. S-plane control affects and controls this system through the generated signals. S-plane controller directly performs closed-loop control on the intelligent UWR and outputs control parameters k_1, k_2 through BPNN (Kabanov, Kramar & Ermakov, 2021). In forward propagation, based on the current control deviation and deviation change rate $e, \dot{e}$, the control parameters k_1, k_2 are calculated through inverse normalization. In back propagation, the weights are adjusted based on the response y_{out} of intelligent UWR and the control input r_{in}, thereby optimizing the network output and achieving online tuning of control parameters.

In Figure 2, a three-layer BPNN with a 2-5-2 structure is adopted.

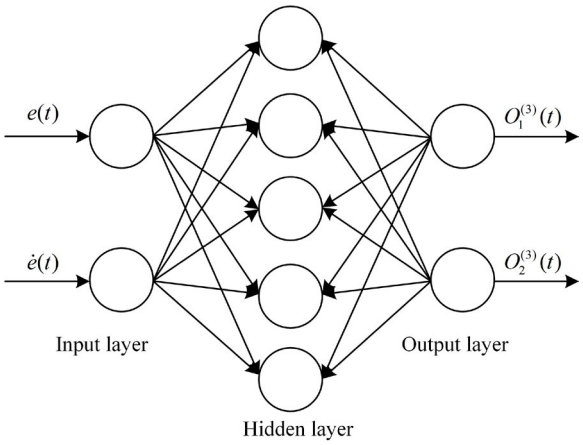


Figure 2. A three-layer BPNN

The input set at the input layer of the forward propagation network is $O_i^{(1)}(t)$, and equation (3) can be obtained:

$$\begin{cases} O_1^{(1)}(t)=e(t) \\ O_2^{(1)}(t)=\dot{e}(t) \end{cases} \tag{3}$$

In equation (3), $e, \dot{e}$ represent the deviation and its derivative at time t, respectively. Equation (4) represents the inputting and outputting of hidden layer:

$$\begin{cases} n_j^{(2)}(t)=\sum_{i=1}^{2}\omega_{ij}^{(2)}(t)O_i^{(1)} \\ O_j^{(2)}(t)=f(_j^{(2)}(t)) \end{cases} \tag{4}$$

In equation (4), $\omega_{ij}^{(2)}(t)$ represents the weight of the connection between the i-th neuron in the input layer and the j-th neuron in the hidden layer at a certain time. $n_j^{(2)}(t), O_j^{(2)}(t)$ represent the activating functions of the j-th hidden layer's neuron. The input is the weighted sum received by the neuron. The output is the result processed through the activating function. Equation (5) is the activating function of the hidden layer (Baysal & Altas, 2022):

$$f(x)=\frac{1}{1+e^{-x}} \tag{5}$$

Equation (5) is the sigmoid function. Equation (6) represents the inputting and outputting of outputting layer.

$$\begin{cases} n_k^{(3)}(t)=\sum_{j=1}^{5}\omega_{jk}^{(3)}(t)O_j^{(2)}(t) \\ O_k^{(3)}(t)=g(n_k^{(3)}(t)) \end{cases} \tag{6}$$

In equation (6), $\omega_{jk}^{(3)}(t)$ is the connection weight between the j-th hidden layer neuron and the k-th output layer neuron at time t. $n_k^{(3)}(t), O_k^{(3)}(t)$ are

the inputting and outputting of the activating functions for the *k*-th output layer (Cheng, Chen & Wang, 2020). Equation (7) is the activating function of outputting layer:

$$g(x)=\frac{1}{1+e^{-x}} \tag{7}$$

Equation (7) represents the sigmoid function. After de-normalization, S-plane control parameters are obtained in equation (8):

$$\begin{cases} k_1=10\times O_1^{(3)}(t) \\ k_2=10\times O_2^{(3)}(t) \end{cases} \tag{8}$$

Equation (8) maps the processed and normalized output values back to the original S-plane control parameters for practical control tasks. Then, the sampling control object responds to $y_{out}(t)$, updates $e(t)=r_{in}(t)-y_{out}(t)$, and selects the performance indicator function as it is expressed in equation (9):

$$E(t)=\frac{1}{2}(r_{in}(t)-y_{out}(t))^2 \tag{9}$$

Afterwards, the gradient descent method is used to modify the weights, which searches and adjusts based on the negative gradient direction of weights to determine the weight adjustment in equation (10):

$$\Delta\omega(t)=-\eta\frac{\partial E(t)}{\partial\omega(t)} \tag{10}$$

In equation (10), η represents the learning rate. For the outputting layer, equation (11) is then used for back propagation:

$$\frac{\partial E(t)}{\partial\omega_{jk}^{(3)}(t)}=\frac{\partial E(t)\partial y_{out}(t)\partial O_k^{(3)}(t)\partial n_k^{(3)}(t)}{\partial y_{out}(t)\partial O_k^{(3)}(t)\partial n_k^{(3)}(t)\partial\omega_{jk}^{(3)}(t)} \tag{11}$$

In equation (11), $\frac{\partial y_{out}(t)}{\partial O_k^{(3)}(t)}$ is unknown and can be replaced by $\operatorname{sgn}\left(\frac{\partial y_{out}(t)}{\partial O_k^{(3)}(t)}\right)$, which can be compensated for by adjusting learning rate:

$$\frac{\partial O_k^{(3)}(t)}{\partial n_k^{(3)}(t)}=g`(n_k^{(3)}(t))=O_k^{(3)}(t)(1-O_k^{(3)}(t)) \tag{12}$$

Equation (13) represents the update of outputting layer weights:

$$\omega_{jk}^{(3)}(t+1)=\omega_{jk}^{(3)}(t)-\eta\varphi_k^{(3)}(t)O_j^{(2)}(t) \tag{13}$$

In equation (13), the weight of the outputting layer is adjusted to obtain equation (14):

$$\varphi_k^{(3)}(t)=e(t)\operatorname{sgn}\left(\frac{\partial y_{out}(t)}{\partial O_k^{(3)}(t)}\right)O_k^{(3)}(t)(1-O_k^{(3)}(t)) \tag{14}$$

Similarly to equation (14), the hidden layer weight update in equation (15) is derived:

$$\omega_{ij}^{(2)}(t+1)=\omega_{ij}^{(2)}(t)-\eta\varphi_j^{(2)}(t)O_i^{(2)}(t) \tag{15}$$

In equation (15),

$$\varphi_j^{(2)}(t)=O_j^{(2)}(t)(1-O_j^{(2)}(t))\sum_{k=1}^{2}\varphi_k^{(3)}(t)\omega_{jk}^{(3)}(t)\,.$$

2.2 UWR Motion Mode and Structural Scheme Design

UWR (Underwater Robot) is a type of autonomous or remotely controlled device designed to perform a variety of tasks in underwater environments. Its applications span across marine scientific research, seabed exploration, resource collection, and rescue missions, among others. The design of UWR's motion modes and structural schemes is of paramount importance as it directly impacts the robot's performance, stability, and task execution capabilities. To cater to the diverse requirements of different tasks, the design process must take into account multiple factors, including the type of robot, motion modes, mechanical structure, sensor systems, control systems, and power systems (Fang et al., 2022). Figure 3 illustrates the motion modes and structural scheme of UWR.

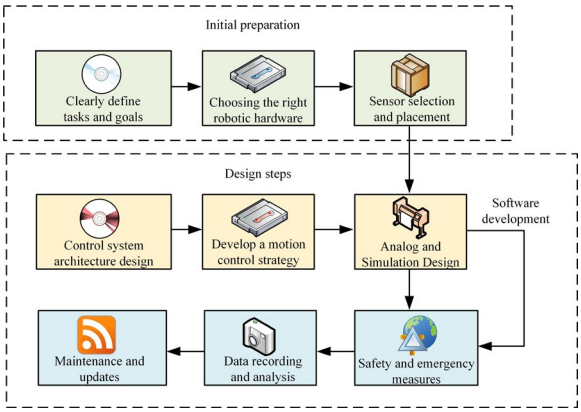


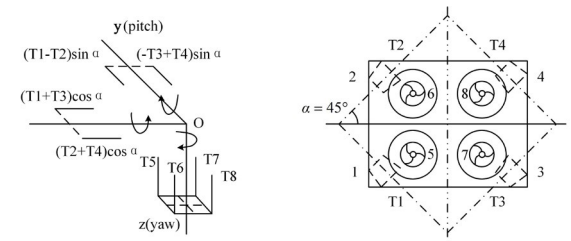
Figure 3. Underwater robot motion modes and structural design process

As it is shown in Figure 3, a systematic approach was followed to design and configure UWRs for specific tasks and application areas. The methodology began with a clear determination of the specific tasks and application domains of

UWRs, accompanied by the definition of key performance indicators tailored to each task. Subsequently, the selection of an appropriate UWR type was based on the specific requirements of each task, thereby defining the UWR's motion mode. The fundamental operations and behaviors associated with each motion mode were rigorously defined. To ensure the robot`s seamless integration with the underwater environment and task requirements, a meticulous redesign of the mechanical structure of the robot was undertaken. This redesign encompassed considerations of size and shape, allowing the UWR to adapt effectively to its operational environment (Javernik, Buchmeister & Ojstersek, 2022). The selection and arrangement of sensors were then meticulously carried out to gather essential data about the underwater environment. Attention was given to ensuring compatibility between the sensor system and the robot's mechanical structure. A robust control system was designed with a focus on intelligent control, enabling autonomous navigation and task execution. The power system, comprising batteries, motors, and electronic components, was engineered to support the robot's operation while meeting specific range requirements (He, Yang & Jiao, 2023). To guarantee the safety of the robots and establish effective emergency response measures, comprehensive simulations and testing were conducted during the design phase to verify the robot`s performance and the related safety parameters. This included thorough debugging and calibration in preparation for on-site testing within the relevant underwater environments. During task execution, sensor data was systematically recorded and analyzed to evaluate the success of each mission. Furthermore, proactive maintenance and periodic updates to the robot's motion mode and structural configuration were implemented to ensure its long-term performance and reliability. Figure 4 showcases the ROV model, which exemplifies the culmination of this comprehensive design and configuration process.

Figure 4. ROV model

To create Figure 4, the fluid simulation software Fluent was employed, which is widely used in scientific research and engineering fields. It incorporates validated physical modeling methods, including Reynolds-averaged Navier-Stokes equations, turbulence models, and multiphase flow models, to accurately simulate complex fluid behavior. In addition, Fluent also has powerful computing power to handle large-scale computing tasks, which enables the efficient simulation of fluid interactions in UWRs under different conditions (Banjanović-Mehmedović et al., 2021). In simulation, the first step is to establish an accurate fluid domain to ensure that all fluid conditions around the UWR are included (Shi et al., 2020). Then, grid partitioning is performed, and its precision is crucial for the accuracy of the simulation results. Subsequently, appropriate boundary conditions are established to simulate the interaction between UWR and surrounding fluids. After defining all the parameters required for simulation, Fluent software performed simulation calculations to obtain key information such as UWR motion in water and fluid damping coefficient (Chai & Xia, 2023).



(a) The function of propeller in swimming mode (b) Thruster layout in swimming mode

Figure 5. Propeller layout and function in swimming mode

As it can be seen in Figure 5, to meet the high maneuverability required for UWR in complex tasks, this design emphasizes the implementation of 6 degrees of freedom for control purposes. These six freedom degrees include the translational degrees of freedom of the robot in three directions (forward and backward, left and right, up and down), as well as the degrees of freedom related to the rotation around three axes (rolling, pitch, and yaw). To achieve this goal, a layout pattern of 8 thrusters was adopted. $Tn(n=1,2,\ldots,8)$ represents the force and direction of each thruster. The specific thruster configuration scheme in Figure 5(b)

involves four thrusters arranged horizontally, which are positioned with 45° inclination along the horizontal axis. This layout allows the robot to achieve forward and backward translation, left and right translation, and steering functions. This is a key element necessary for robots to perform posture transformation and planar motion. To that, to counteract the torque caused by the rotation of the internal spiral blades, the four thrusters in the vertical position use suction cups instead. These suction cups are arranged alternately with reverse and forward propellers to maintain the balance of the robot (Li & Zhao, 2022). This configuration not only ensures the mobility of the robot in all directions, but it also helps to reduce unnecessary torque caused by rotational motion.

3. Result Analysis

This section focused on verifying the BPNN-based UWR control. Firstly, the application effect of BPNN in S-plane control of UWR was discussed. Next, the result analysis for the motion mode and structural scheme design of UWR was presented.

3.1 The Application Effect of BPNN in S-plane Control of UWR

The initial weights of BPNN were randomly selected from within $[-0.5, +0.5]$. Learning rate was $n = 0.01$ and a classic S-plane controller was used, with control parameters $k = 3, k` = 4$. Table 1 includes the parameters of the motion model and the motion control objectives for UWR.

The target for speed control was 1.5 meters per second. The goal of directional control was to maintain the robot at 60 degrees. These two control objectives were key to performing successful UWR tasks and operations. Figure 6 shows the motion control effect.

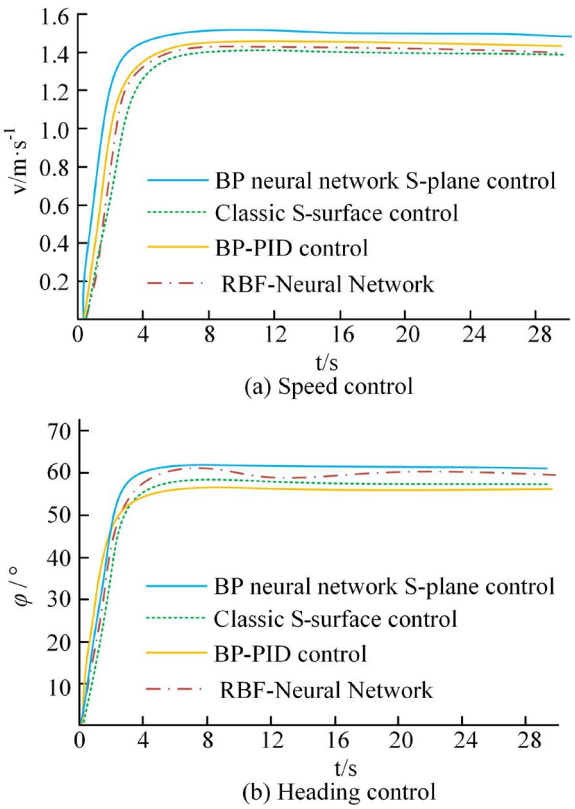


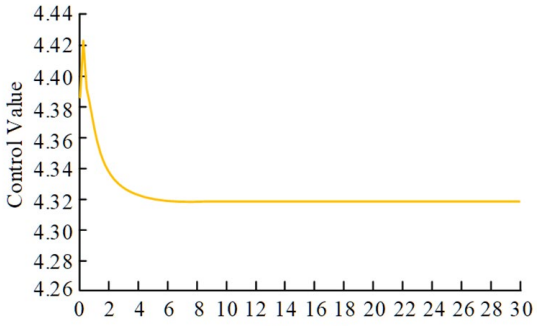
Figure 6. Motion control effect

As it is shown in Figure 6, this paper analyses the performance of the NeuroPID model, traditional model, and RBF-Neural Network model. The results showed that the proposed BP-PID model performed slightly better in many aspects. First of all, as it is illustrated in Figure 6(a), in terms of speed control, the proposed model featured a very fast speed response, hence it can quickly reach the expected speed, and no excessive oscillating or steady state error occured after the target speed was reached. This shows its excellent speed control performance. This is very important for the task that requires fast transmission. Then, as it is shown in Figure 6(b), in terms of direction control, although the model proposed by this study may have a slight super-adjustment at the initial stage, that is, the direction may be slightly deviated from the target,

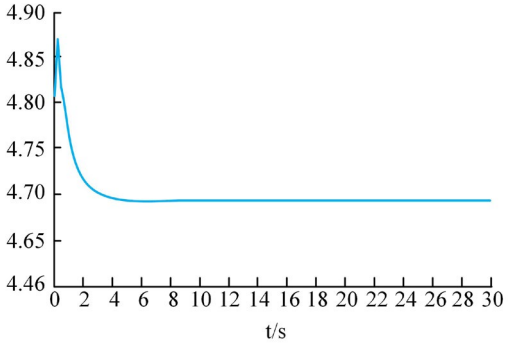
Table 1. Motion model parameters

Parameter	Value	Parameter	Value
L	2.55m	N_r	-253.08kg·m²/(s·rad)
m	183.0kg	X_u	0
I_Z	94.55kg·m2	Y_v	-149.83kg/s
$N_{\dot{r}}$	-88.48kg·m2	$N_{r\|r\|}$	-79.75kg·m²/rad
$X_{\dot{u}}$	-13.41kg	$X_{u\|u\|}$	-16.66kg/m
$Y_{\dot{v}}$	-261.34kg	$Y_{v\|v\|}$	-556.10kg/m

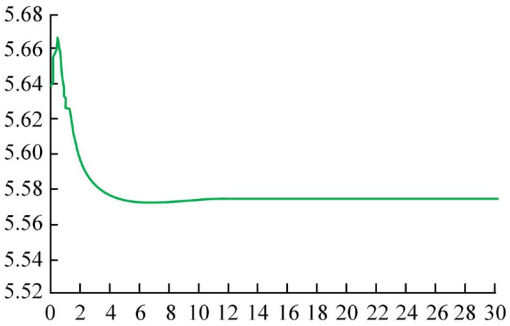
still the gap is controlled within 5%. The proposed model can still respond quickly and stabilize the gap to reach the expected direction, and there is no steady state error. By contrast, the NeuroPID model, traditional model, and RBF-Neural Network model are slightly inferior in these regards. The S-plane control effect is shown in Figure 7.



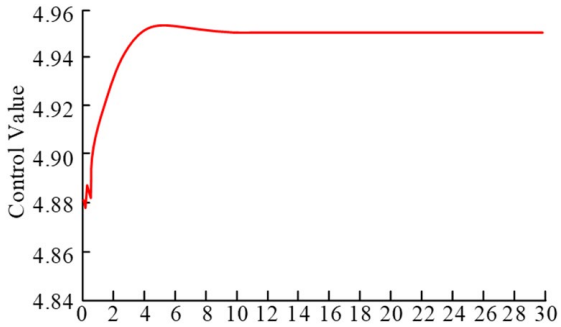
(a) k_1 in speed control



(b) $k_2$in speed control



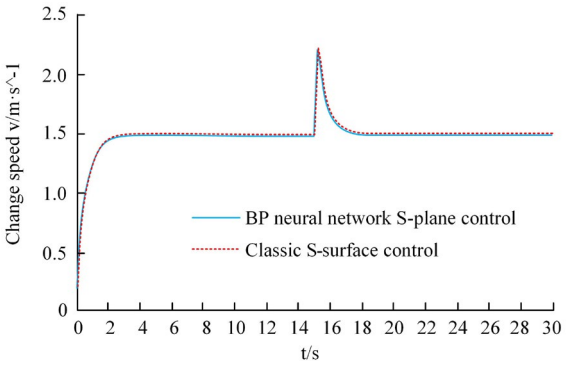
(c) $k_1$in heading control



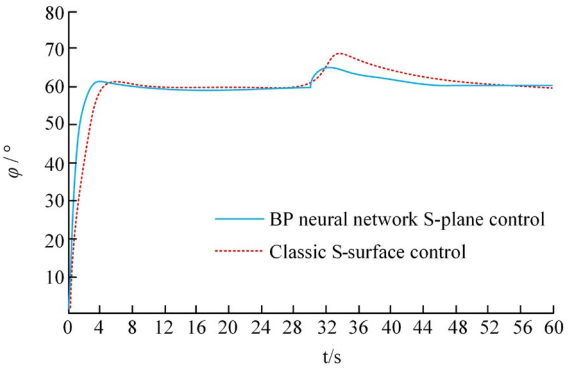
(d) k_2 in heading control

Figure 7. S-surface control effect

S-plane controller of BPNN exhibited an excellent performance in intelligent UWR motion control, thanks to its data-driven characteristics. Firstly, it could autonomously initialize and adjust control parameters online, which was the foundation of its excellent performance. By collecting a large amount of data related to speed control and heading control, the S-surface controller could dynamically optimize control parameters in real-time operation to adapt to different underwater environments and task requirements. Secondly, S-plane controller exhibited excellent anti-interference ability when facing disturbances. This feature was particularly evident in heading control. Through real-time data analysis and feedback, the controller could effectively suppress the fluctuation amplitude and ensure that the UWR could maintain the required direction stably. The S-plane control results are shown in Figure 8.



(a) Speed control



(b) Heading control

Figure 8. S-surface control result

The performance of BPNN's S-plane controller benefitted from a large amount of data. By collecting speed control data from UWR, after initializing the control parameters, the controller could increase the robot's speed to 1.5 meters per second in less than 2 seconds without obvious overshoot. This reflected its fast response and excellent speed control ability. In terms of heading control, when facing the interference term $[d\dot{u}, d\dot{r}, d\dot{i} = 500, 0, 100]$, S-plane

controller could quickly adjust the direction of the UWR, and the fluctuation amplitude was relatively small, not exceeding 5 degrees. This showed the anti-interference characteristics of the controller against disturbances.

3.2 Results Analysis for UWR Motion Mode and Structural Scheme

In the design and motion control of UWRs, the motion modes and structural schemes were crucial. Through a reasonable design and selection, the efficient movement and operation of UWR could be achieved for different tasks and environments. This article aimed to conduct in-depth research on the motion mode and structural scheme of UWR, to analyse its performance. Time-frequency plots, as a key analytical tool, would be used to analyze the motion characteristics of UWRs, including the working status of thrusters and the impact of underwater environments. A complex propulsion motion simulation model was developed using Matlab/Simulink software, which adopted a speed closed-loop control strategy. In the speed closed-loop control system, a Proportional Integral (PI) controller was used for research, whose main task was to manage the intensity of bus current, thereby adjusting the output voltage of the inverter to achieve precise speed control of UWR. The simulation covered two key scenarios: the normal operation of the thruster and the short-circuit of the B and C phases for the motor. Figure 9 illustrates the obtained stator time-frequency diagram. It presents the deep analysis of the current signal of the UWR thruster. Firstly, at a steady-state speed of 1000 rad·s^{-1}, Figure 9(a) shows the time domain current signal, a notable feature being that it was mainly composed of sinusoidal components. Furthermore, in the frequency domain current signal in Figure 9(b), the frequency was mainly distributed around 5, 25, 50, and 100 Hz, with amplitudes around 4.5, 1.0, 0.5, and 0.2 dB·Hz^{-1}, respectively. In addition, the performance of current signals under fault conditions was also studied. With regard to the A-phase short circuit which was introduced at 0.1 seconds at a steady-state speed of 1000 rad·s^{-1}, Figure 9(c) shows the evolution of time domain current signal, where a significant negative sequence component appeared. For the frequency domain current signal in Figure 9(d), the frequency components were mainly distributed around 10, 30, 50, and 75 Hz, with amplitudes of around 60, 4, 18, and 8 dB·Hz^{-1}, respectively.

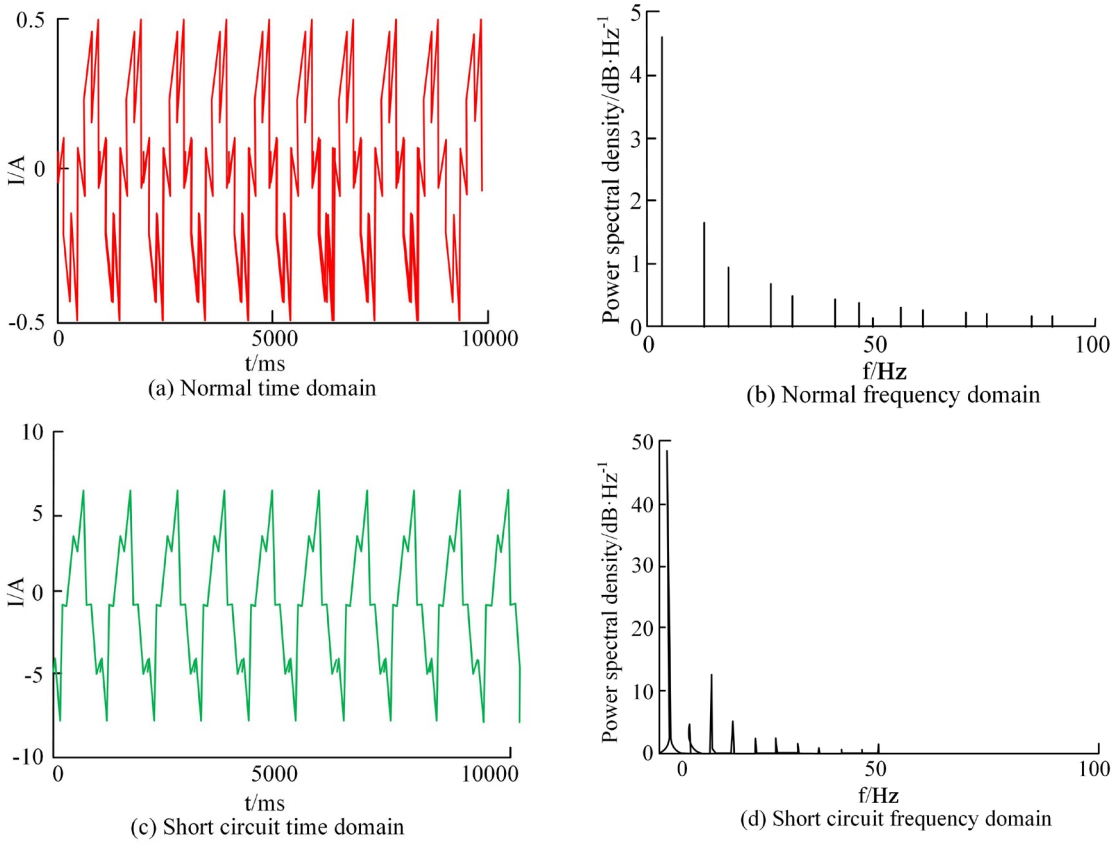


Figure 9. Stator current time-frequency diagram

It is worth noting that in comparison with normal signals, the spectrum of the fault signal showed significant enhancement of low-frequency and high-frequency components, as well as a significant increase in amplitude. Vertical thruster output is shown in Figure 10.

After 10 seconds of Z-direction control, the output control values for thrusters 5 to 8 exhibited excellent stability. Specifically, the output control of thruster 5 in the Z direction remained stable at about -2 Newton, for thruster 6 it remained stable at 5 Newton, for thruster 7 at -12 Newton, and thruster 8 maintained an output control of about 13 Newton. These data results indicated that each thruster could reliably meet the control requirements of UWR in all directions according to the design requirements. Figure 11 shows the classification results. Classification results are shown in Figure 11.

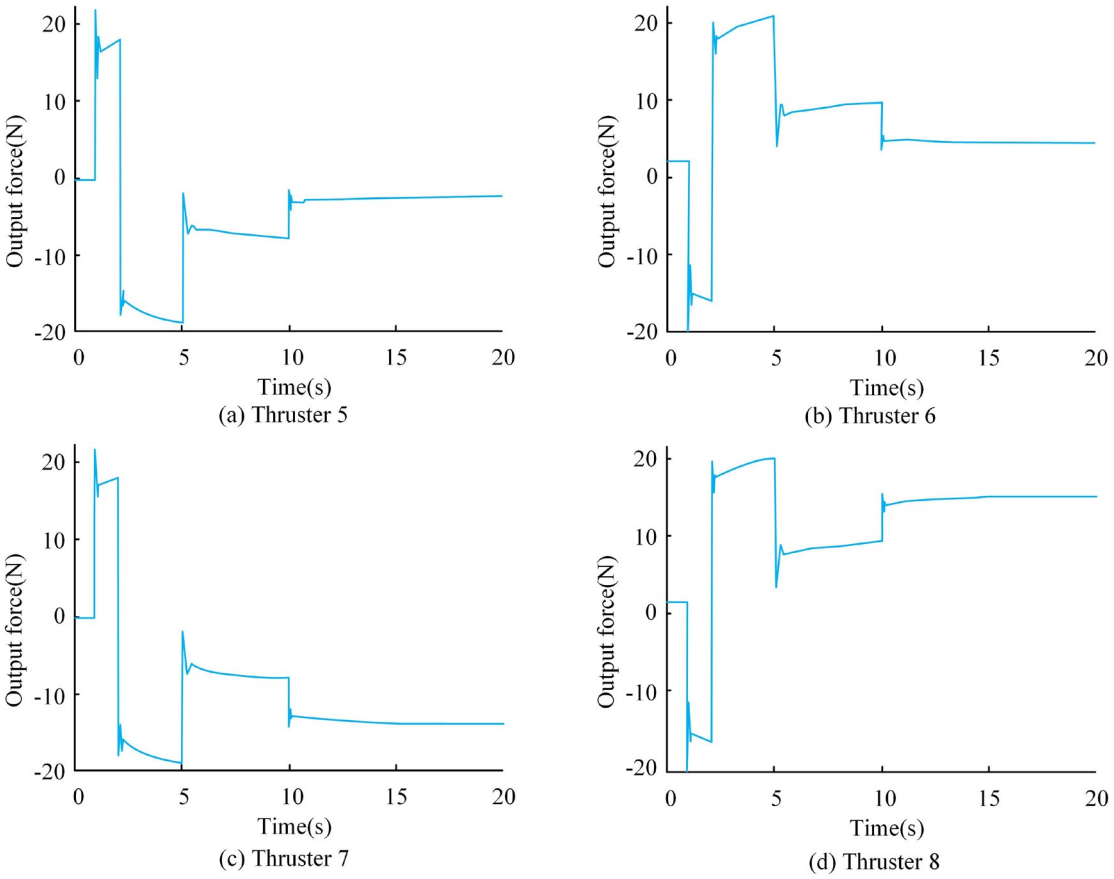


Figure 10. Vertical thruster output

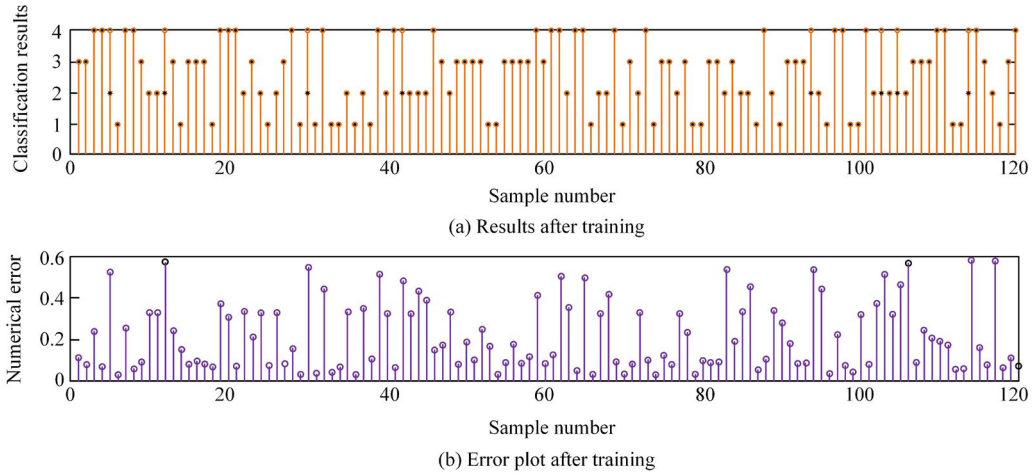


Figure 11. Classification results

The training results show that out of a total of 120 samples, only 8 samples were misclassified. These misclassified samples were labeled with circles, while the remaining samples were represented by black dots, indicating that they were correctly classified. In addition, as it was illustrated by the error plots after training, there were certain differences in numerical errors among these misclassified samples, but overall they exhibited high accuracy.

The application of research results in real-life scenarios has improved. First of all, the movement control for underwater robots has been significantly improved, and now they are capable of completing underwater survey and resource exploration tasks with a high efficiency. This not only means that underwater data and resource samples can be obtained faster, but it can also lead to a deep understanding of the marine environment and resources. Scientific research has also benefitted a lot. The research robots show higher accuracy and stability in the underwater environment, which can help better explore many fields such as marine ecosystems, climate change, and underwater geology. In addition, the stability and fault detection ability of robots improve the safety of underwater operations, reduce potential risks, and ensure the safety of operators. The multifunctional characteristics of robots have also been improved, which enables them to adapt to different types of underwater tasks and turns them into multi-purpose tools, which can be used in many contexts, including underwater maintenance, deep-sea exploration, underwater archeology and marine biological research. This series of improvements reduces operating costs, extends the service life of the robot, and also enhances task autonomy and reduces the demand for human intervention.

4. Conclusion

As technology advances, the UWR is becoming increasingly important in areas such as ocean exploration, seabed resource development, and marine environmental monitoring. This paper provides a detailed introduction to the application of BPNN in UWR control and verifies its effectiveness through experiments. The experiments confirmed that the speed control response of UWR was fast, and it reached the required speed quickly, without overshoot or steady-state error. The direction control performance was good, with a slight overshoot which did not exceed 5%. It responded quickly and stabilized to the desired direction without steady-state error. S-plane controller of BPNN exhibited excellent performance in intelligent UWR motion control, as it independently completed control parameter initialization and online adjustment, and featured an excellent anti-interference ability. The signal analysis which was carried out showed the thruster performance under normal and fault conditions. At a steady-state speed of 1000 rad·s-1, the generated signal was mainly composed of sinusoidal components, with a frequency distribution around 5, 25, 50, and 100 Hz. When a fault occured, the generated signal exhibited a negative sequence component, with a frequency distribution of around 10, 30, 50, and 75 Hz, and a significant increase in amplitude. The output control values for thrusters 5 to 8 exhibited excellent stability after Z-direction control, and the output control values for each thruster reliably met the control requirements of the analysed UWR. After real-life tests, it was noticed that the optimized robot can more effectively avoid obstacles, adapt to varying underwater terrains, and achieve a more precise positioning. Additionally, the underwater robot's operational accuracy has been enhanced, thereby reducing errors and repetitive work. By better predicting and responding to environmental changes, the safety and reliability of underwater operations have been improved. The proposed automated and intelligent control system reduces the need for manual intervention, thereby saving time and costs.

The paper mainly focused on the short-term performance of the UWR, but the long-term stability and maintenance requirements for UWRs are also important topics that would require further research and experimental verification.

Acknowledgements

This work was supported by Zhejiang Provincial Natural Science Foundation (LQ21E010004), the Key R&D Program of Shaanxi Province (2023-YBNY-202) and Ningbo Science and Technology Plan Project (2021S022).

REFERENCES

Afriliana, N. & Ramadhan, A. (2022) The trends and roles of robotic process automation technology in digital transformation: a literature review. *Journal of System and Management Sciences*. 12(3), 51-73.

An, R., Guo, S., Yu, Y., Li, C. & Awa, T. (2022) Task planning and collaboration of jellyfish-inspired multiple spherical underwater robots. *Journal of Bionic Engineering*. 19(3), 643-656. doi: 10.1007/s42235-022-00164-6.

Banjanović-Mehmedović, L., Karabegović, I., Jahić, J. H., & Omercic, M. (2021). Optimal path planning of a disinfection mobile robot against COVID-19 in a ROS-based research platform. *Advances in Production Engineering & Management*. 16(4), 405–417. doi: 10.14743/apem2021.4.409.

Baysal, Y. A. & Altas, I. H. (2022) A Fast Non-dominated Sorting Multi-objective Symbiotic Organism Search Algorithm for Energy Efficient Locomotion of Snake Robot. *Computer Science and Information Systems*. 19(1), 353-378. doi: 10.2298/csis210222067b.

Chai, G. F., & Xia, Y. Z. (2023) Multi-Robot Path Optimization and Simulation for Multi-Route Inspection in Manufacturing. *International Journal of Simulation Modelling*. 22(1), 121-132. doi: 10.2507/IJSIMM22-1-CO1

Cheng, P., Chen, D. & Wang, J. (2020) Clustering of the body shape of the adult male by using principal component analysis and genetic algorithm–BP neural network. *Soft Computing*. 24(17), 13219-13237. doi: 10.1007/s00500-020-04735-9.

Christensen, L., de Gea Fernández, J., Hildebrandt, M., Koch, C. E. S. & Wehbe, B. (2022) Recent advances in AI for navigation and control of underwater robots. *Current Robotics Reports*. 3(4), 165-175. doi: 10.1007/s43154-022-00088-3.

Cruz Ulloa, C., Terrile, S. & Barrientos, A. (2020) Soft underwater robot actuated by shape-memory alloys "Jellyrobcib" for path tracking through fuzzy visual control. *Applied Sciences*. 10(20), 7160-7181. doi: 10.3390/app10207160.

Ding, M., Zhang, S., Zhong, H., Wu, Y. & Zhang, L. (2019) A prediction model of the sum of container based on combined BP neural network and SVM. *Journal of Information Processing Systems*. 15(2), 305-319. doi: 10.3745/JIPS.04.0107.

Fang, Y., Luo, B., Zhao, T., He, D., Jiang, B. & Liu, Q. (2022) ST-SIGMA: Spatio-temporal semantics and interaction graph aggregation for multi-agent perception and trajectory forecasting. *CAAI Transactions on Intelligence Technology*. 7(4), 744-757. doi: 10.1049/cit2.12145.

Filaretov, V. F., Konoplin, A. Y., Zuev, A. V. & Krasavin, N. A. (2021) A Method to Synthesize High-Precision Motion Control Systems For Underwater Manipulator. *International Journal of Simulation Modelling*. 20(4), 625-636. doi: 10.2507/IJSIMM20-4-571.

Hasan, M. W. & Abbas, N. H. (2023) Disturbance rejection based on adaptive neural network controller design for underwater robotic vehicle. *International Journal of Dynamics and Control*. 11, 717-737. doi: 10.1007/s40435-022-00995-5.

He, Z., Yang, Y. & Jiao, P. (2023) Copebot: underwater soft robot with copepod-like locomotion. *Soft Robotics*. 10(2), 314-325. doi: 10.1089/soro.2021.0158.

Heshmati-Alamdari, S., Karras, G. C., Marantos, P & Kyriakopoulos, K. J. (2019) A robust predictive control approach for underwater robotic vehicles. *IEEE Transactions on Control Systems Technology*. 28(6), 2352-2363. doi: 10.1109/TCST.2019.2939248.

Javernik, A., Buchmeister, B. & Ojstersek, R. (2022). Impact of Cobot parameters on the worker productivity: Optimization challenge, *Advances in Production Engineering & Management*. 17(4), 494-504. doi: 10.14743/apem2022.4.451.

Jiang, J., Hu, G., Li, X., Xu, X., Zheng, P. & Stringer, J. (2019) Analysis and prediction of printable bridge length in fused deposition modelling based on back propagation neural network. *Virtual and physical prototyping*. 14(3), 253-266. doi: 10.1080/17452759.2019.1576010.

Kabanov, A., Kramar, V. & Ermakov, I. (2021) Design and modeling of an experimental ROV with six degrees of freedom. *Drones*. 5(4), 113-134. doi: 10.3390/drones5040113.

Lee, Y., Lee, Y., Chae, J., Choi, H. & Yeu, T. (2020) A study on the development of underwater robot control system for autonomous grasping. *The Journal of Korea Robotics Society*. 15(1), 39-47.

Li, D., Huang, F., Yan, L., Cao, Z., Chen, J. & Ye, Z. (2019) Landslide susceptibility prediction using particle-swarm-optimized multilayer perceptron: Comparisons with multilayer-perceptron-only, BP neural network, and information value models. *Applied Sciences*. 9(18), 3664-3681. doi: 10.3390/app9183664.

Li, P. & Zhao, L. (2022) A Novel Art Gesture Recognition Model Based on Two Channel Region-Based Convolution Neural Network for Explainable Human-computer Interaction Understanding. *Computer Science and Information Systems*. 19(3), 1371-1388. doi: 10.2298/CSIS220322037L.

Lu, Q., Yang, R., Zhong, M. & Wang, Y. (2019) An improved fault diagnosis method of rotating machinery using sensitive features and RLS-BP neural network. *IEEE Transactions on Instrumentation and Measurement*. 69(4), 1585-1593. doi: 10.1109/TIM.2019.2913057.

Shi, J., Chengchao, S., Lei, H. & Mengxi, X. (2020) Smart grid short-term load estimation model based on BP neural network. *International Journal of Computing Science and Mathematics*. 11(2), 123-136. doi: 10.1504/IJCSM.2020.106390.

Subad, R. A. S. I., Cross, L. B. & Park, K. (2021) Soft robotic hands and tactile sensors for underwater robotics. *Applied Mechanics*. 2(2), 356-382. doi: 10.3390/applmech2020021.

Wang, R., Wang, S., Wang, Y., Cheng, L. & Tan, M. (2022) Development and motion control of biomimetic underwater robots: A survey. *IEEE Transactions on Systems, Man and Cybernetics: Systems*. 52(2), 833-844. doi: 10.1109/TSMC.2020.3004862.

Yang, A., Zhuansun, Y., Liu, C., Li, J. & Zhang, C. (2019) Design of intrusion detection system for internet of things based on improved BP neural network. *IEEE Access*. 7, 106043-106052. doi: 10.1109/ACCESS.2019.2929919.

Yang, Y. H. & Shi, Y. (2020) Application of improved BP neural network in information fusion Kalman filter. *Circuits, Systems and Signal Processing*. 39(10), 4890-4902. doi: 10.1007/s00034-020-01393-y.

Zhao, X. (2023) Implementation of English ICAI MOOC system based on BP neural network. *Journal of Ambient Intelligence and Humanized Computing*. 14(4), 3177-3186. doi: 10.1007/s12652-021-03446-9.