Deep reinforcement learning-based pitch attitude control of a beaver-like underwater robot

Deep reinforcement learning-based pitch attitude control of a beaver-like underwater robot

Gang Chen [a,b,*], Zhihan Zhao [a], Yuwang Lu [a], Chenguang Yang [c,**], Huosheng Hu [b]

[a] School of Mechanical Engineering, Zhejiang Sci-Tech University, Hangzhou, 310018, China
[b] School of Computer Science and Electronic Engineering, University of Essex, Colchester, CO4 3SQ, UK
[c] Bristol Robotics Laboratory, University of the West of England, Bristol, BS16 1QY, UK

A B S T R A C T

The foot paddling of an underwater robot causes continuous changes of the water flow field, which results in the unbalanced hydrodynamic force to change the robot's posture continuously. As the water environment and robot swimming are nonlinear and strongly coupled systems, it is difficult to establish an accurate model. This paper presents an underwater robot, which adopts the synchronous and alternate swimming trajectory of a beaver. Its pitch stability control model is established by using deep reinforcement learning algorithm and its self-learning control system is constructed for stable control of pitch attitude. Experiments are conducted to show that the pitch attitude of the beaver-like underwater robot can be stabilized while maintaining a certain swimming speed. The control method does not need to establish a complex and high-order model of webbed paddling hydrody-namics, which provides a new idea for stable swimming control of underwater robots.

This work aims to find an excellent control method for underwater bionic robots. The ocean has the richest natural resources and the most diverse species on Earth. The underwater environment is complex and variable, imposing higher demands on the performance of underwater robots. Increasingly, new concept marine equip-ment is being researched for scientific exploration, and among these, underwater robots designed based on bionic principles are a growing trend. Currently, most underwater robots still use propellers as their propulsion system. Propellers have advantages such as simple control, high mechanical efficiency, and powerful propulsion, but they also have drawbacks including severe water flow disturbance during operation, high noise, poor concealment, and limited adaptability in complex water environments. Finding a propulsion system with better overall per-formance is a crucial way to enhance the motion capabilities of underwater robots. Underwater robots often have complex structures, and there are numerous factors influencing their movement in the underwater environment, making fluid dynamics modeling and optimization challenging. Reinforcement learning, as an optimization al-gorithm, can circumvent the aforementioned difficulties.

1. Introduction

As bionic underwater robots are very useful for human exploration of the ocean, Li, Shintake, Raj, Zhu et al. designed bionic fish robots to mimic fishes in nature (Li et al., 2018; Shintake et al., 2020; Raj et al., 2016; Zhu et al., 2019). However, these bionic fish robots cannot operate well in complex underwater environments, such as submarine wrecks and coral reefs. It is necessary to develop underwater crawling and swimming robots that can move and crawl on the seabed for underwater detection tasks. Recently, some underwater crawling and swimming robots have been developed. For instance, Crespi et al. conducted a bionic study on salamanders and designed a salamander-like robot that can swim in water and crawl on land using the CPG control method (Crespi et al., 2013; Ma et al., 2022; Karakasiliotis et al., 2009). Kim et al. designed the underwater multi-legged bionic robot CR200 (Kim et al., 2013), which can achieve crawling in complex seabed under the disturbance of ocean currents. Avi Cohen et al. designed a novel high-speed amphibious robot which possesses a sprawling mechanism inspired by cockroaches (Zarrouk et al., 2013, 2015, 2018; Cohen et al., 2020).

However, current underwater crawling and swimming robots generally suffer from very complex structures and difficult motion control (Chen et al., 2022a; Zhang et al., 2022), and cannot fully realize the flexible motion in the underwater environment. In this paper, we proposed a beaver-like crawling and swimming robot which swims by paddling its webbed hind legs and crawls with its limbs. The robot only uses a set of driving mechanism to achieve the swimming and crawling capabilities without deformation. Our research is focused on the swimming capability study of the beaver-like underwater robot platform.

The attitude stability is an important performance indicator for underwater robots, which determines whether the robots can adapt to complex and challenging underwater environments to accomplish corresponding tasks (Shen et al., 2020; Li et al., 2021; Chen et al., 2023a). For a beaver-like robot, two hind legs paddle back and forward during swimming, resulting back and forward of its robot body pitches. Large pitch may cause its body to swing up and down, reducing the robot's swimming efficiency and increasing its control difficulty. Therefore, it is very important to control the attitude stability of a robot in swimming, and some traditional control algorithms have been deployed, including slide mode control, PID and so on (Chen et al., 2018, 2022b, 2023b; Pu et al., 2012; Gang et al., 2016; Sun et al., 2012). These algorithms have insufficient adaptability to unknown environments or disturbances.

The reinforcement learning has good adaptability and can be directly trained to learn the swimming strategy for attitude stabilization control in water (Wang et al., 2022; B et al., 2020; Zhang et al., 2012). It does not require a complex model of attitude stabilization dynamics of the underwater robots. Currently, reinforcement learning is mainly applied to autonomous underwater robots. For instance, Zhang et al. used deep interactive reinforcement learning for robot path tracking tasks (Zhang, 2020). Li et al. implemented the heading attitude control of an AUV and the effectiveness of reinforcement learning algorithm was verified by simulation (Zhang et al., 2012). Woo et al. realized path tracking of unmanned surface robot based on reinforcement learning (Woo et al., 2019).

The D3QN algorithm, which stands for "Double Deep Q-Network with Dueling Architecture", is an extension of the Deep Q-Network (DQN) algorithm used in reinforcement learning. It combines the concepts of double DQN and dueling DQN to improve the stability and efficiency of learning. By combining double DQN and dueling DQN, the D3QN algorithm benefits from both approaches. It uses two neural networks (online and target) to mitigate overestimation bias and incorporates the dueling architecture to separately model state values and action advantages. This results in a more robust and efficient algorithm for reinforcement learning tasks, especially in environments with a large number of states and actions. The D3QN algorithm is used in robot control, but it has problems such as long training time and non-convergence.

The control object of the paper is to stabilizing the pitch attitude of a beaver-like underwater robot with reinforcement learning algorithms. The experimental results show that the pitch attitude of the robot is stable within $\pm 6^{\circ}$ in the synchronous swimming mode and within $\pm 1^{\circ}$ in the alternate swimming mode. The following statement is added in the manuscript.

The innovation points of this paper is as follows.

(1) A beaver-like underwater robot was designed, and the kinematics of the robot is conducted.
(2) Differing from the existing work above, we conducted the study on how reinforcement learning could be used for stabilizing the pitch attitude of a beaver-like underwater robot in swimming.
(3) More specifically, a D3QN-based pitch attitude stabilization control method is proposed for a beaver-like swimming underwater robot based on the swimming trajectory of its hind legs, which provides new ideas for autonomous learning methods for bionic underwater robots.

The rest content of this paper is organized as follows. Section II introduces the structure of our beaver-like underwater robot, mainly the underwater robot's joint structure and hardware composition. In Section III, a novel underwater robot's attitude stabilization control method is proposed by using the Deep Dueling Double Q-learning algorithm to implement the self-learning of attitude stabilization swimming. The experiments of pitch attitude control under synchronous and alternate swimming gait are conducted. Finally, a brief conclusion and future work are given in Section IV.

2. Deep reinforcement learning-based pitch control method

2.1. The beaver-like underwater robot and its kinematics

As shown in Fig. 1, a beaver is an amphibian mammal whose swimming structure mainly includes the hind limbs and tail. The hind limbs are multi-jointed and the foot is a webbed structure which provides strong propulsive force and reduces the resistance when swimming. It has a long and oval tail which can adjust the body attitude by swinging when swimming.

Fig. 2 shows the model of the beaver-like underwater robot. Fig. 3 shows the prototype of the beaver-like underwater robot in water.

The mechanical structure of the beaver-like underwater robot can be divided into three parts: the body, hind legs, and tail. The body contains a sealed cabin with a hemispherical head, which can reduce fluid resistance. Each hind leg has three servo motors to generate propulsion. The tail is used to maintain balance and is controlled by a single servo motor.

Inside the sealed cabin, the control components of the beaver-like underwater robot are placed, including the control board Nvidia Jetson Nano, attitude sensor, and Wi-Fi communication module. The sealed cabin is made of acrylic material and can withstand a water pressure of up to 20 m deep.

The hind limb consists of thighs, calves, and webbed foot with a total of three degrees of freedom and high-torque motors are used to simulate the motion of the beaver joints. The webbed foot adopts a passive contraction structure. In the swimming, the actions of hind limb include kicking and recovering. In the kicking process of hind limb, the water force pushes the webbed foot open, and moves with the maximum water facing area which can increase the propulsive force. In the recovery process of hind limb, the water resistance forces the webbed foot to bend and the water facing area decreases which reduces the swimming resistance and can effectively improve the swimming speed of the robot.

The robot's tail imitates the shape of the beaver, which is made of the material with a certain degree of elasticity and hardness to effectively simulate the swimming characteristics of a beaver's tail. The tail can increase the attitude stability in the pitch direction during the robot's swimming. Table I presents the characteristics of our beaver-like underwater robot.

Beavers are good at swimming in water using their webbed feet and the common swimming modes are synchronous and alternate swimming. In synchronous swimming, its legs move backward and forward at the same time to realize high-speed movement. In alternate swimming, its legs alternately move with one phase difference, which is the basic swimming mode and can well maintain the balance of the pitch attitude (Chen et al., 2021, 2022c). According to the structure of beaver-like underwater robot, we build its kinematics model as shown in Fig. 4.

As shown in Fig. 4(b), l_1 is the length of the upper leg, l_2 is the length of the lower leg, and L is the distance between the hip joint and the ankle joint. the coordinate system OXY is established and the point of ankle joint is $\{x(u), y(u)\}$. The Angle of knee joint θ_{knee} and hip joint θ_{hip} can be obtained by solving inverse kinematics below.

$$\theta_{knee} = \arccos\left(\frac{l_1^2 + l_2^2 - L^2}{2 l_1 l_2}\right) \tag{1}$$

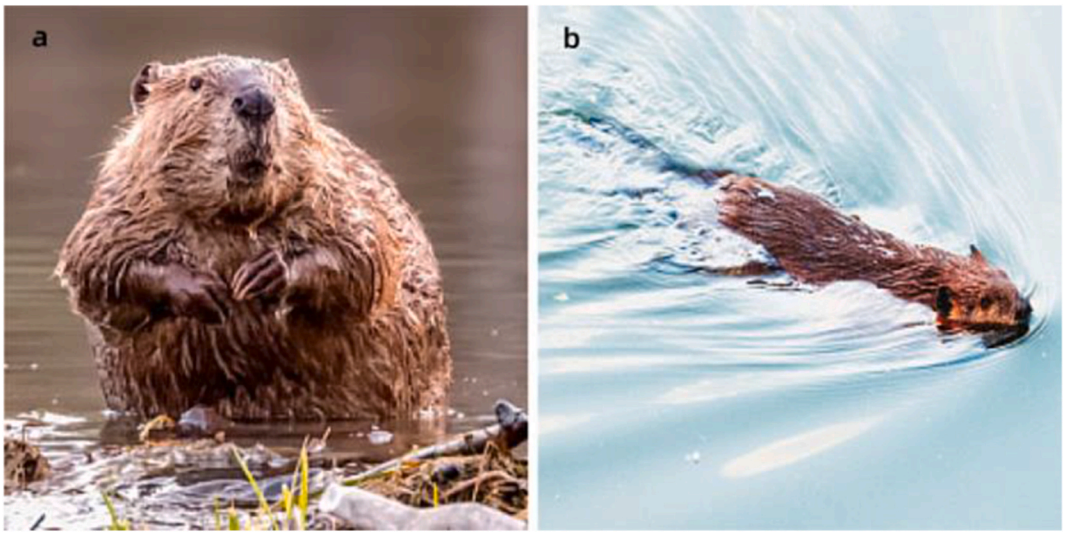

Fig. 1. A beaver in nature. (a) A beaver on the shore, (b) A beaver swims in the water (VCG.COM, 2022a; VCG.COM, 2022b).

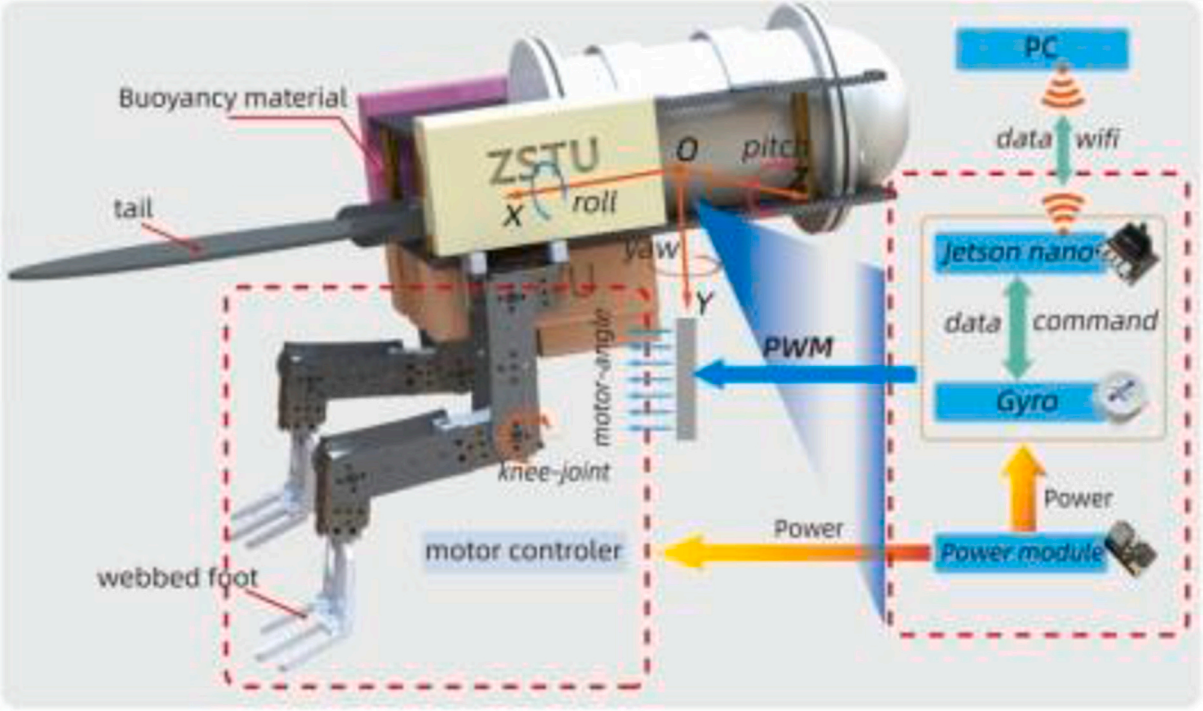

Fig. 2. System structure of the beaver-like underwater robot.

$$L = \left(x(u)^2 + y(u)^2\right)^{\frac{1}{2}} \tag{2}$$

$$\alpha = \arccos\left(\frac{l_1^2 + L^2 - l_2^2}{2 l_1 l_2}\right) \tag{3}$$

$$\beta = \arctan\left(\frac{y(u)}{x(u)}\right) \tag{4}$$

$$\theta_{hip} = 180^\circ - \alpha - \beta \tag{5}$$

where l_1 and l_2 are the length of the robot's thighs and calves, and $\{x(u), y(u)\}$ is the coordinate point of the ankle joint.

2.2. Attitude stabilization control algorithm based on deep reinforcement learning

The control object of the paper is to stabilizing the pitch attitude of a beaver-like underwater robot with reinforcement learning algorithms.

The state of the underwater robot in the real world is partially observable. In this paper, the real-time pitch angle $pitch_{cur}$ and motor angle $angle_i, on_{goal}, done$ of the bionic underwater robot are taken as the elements of the observation space shown in (6) which is a 10-dimension vector.

$$\boldsymbol{s} = \left[\boldsymbol{angle}, pitch_{cur}, done, on_{goal}\right] \tag{6}$$

where ***angle*** is a 7-dimension vector, which represents the angle of the robot's 7 joint motors. $pitch_{cur}$ is the real-time pitch angle to represent

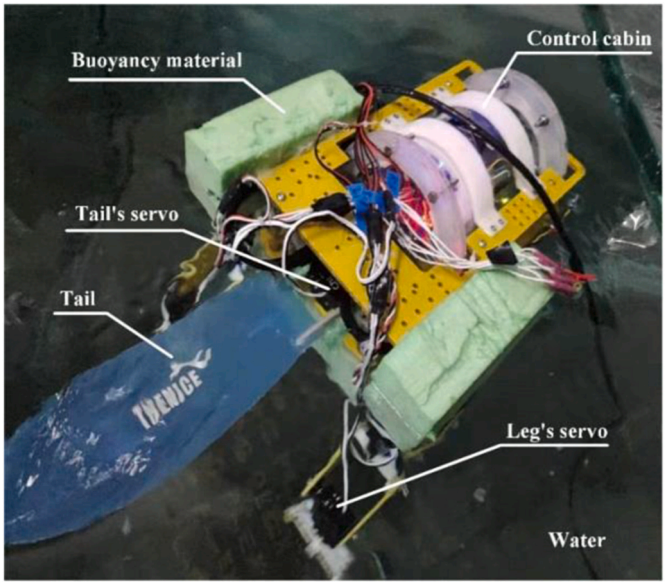

Fig. 3. Beaver-like underwater robot in water.

Table 1
CHARACTERISTICS OF THE BEAVER-LIKE UNDERWATER ROBOT.

parameters	value
Weight	3.25 kg
tail length	240 mm
length of the web	75 mm
length of the hind limb	200 mm

the real-time pitch angle change when the robot moves. *done* represents whether the robot jumps out from the current training episode. on_{goal} represents whether the robot can stay within the desired pitch angle in 5 steps. The desired pitch angle is $[-6°, 6°]$ for synchronous swimming and $[-1°, 1°]$ for alternative swimming.

Focused on the stabilization control of pitch attitude during swimming of the beaver-like underwater robot, the reward function is set to 5 parts in synchronous swimming, which is obtained by multiplying the synchronous weight matrix with the corresponding reward value matrix, as shown in (7).

$$\boldsymbol{r}_{syn} = \boldsymbol{weight}_{syn}\left[r_p, r_r, r_a, r_{er}, r_{ep}\right]^{\mathrm{T}} \tag{7}$$

In alternate swimming, the reward is set to 4 parts, which is obtained by multiplying the alternate swimming matrix with the corresponding reward value matrix, as shown in (8).

$$\boldsymbol{r}_{stagged} = \boldsymbol{weight}_{stagged}\left[r_p, r_r, r_{er}, r_{ep}\right]^{\mathrm{T}} \tag{8}$$

Equations 9 and 10 represent the weighting factor of the reward, in which w_p is the weight of the reward function r_p and indicates whether the actual pitch angle is within the desired pitch angle; w_r is the weight of the reward function r_r and indicates whether the robot completes the reach-goal target weight; w_a is the weight of the reward function r_a and indicates whether the leg moves backward faster than forward in synchronous swimming to imitate the beaver's swimming characteristics; w_{er} is the weight of the reward function r_{er} and indicates the weight of the current reward and the average value of the previous 5 rewards; w_{ep} is the weight of the reward function r_{ep}, and indicates the reward weight of the current pitch angle versus the previous 5 pitch angles.

$$\boldsymbol{weight}_{stagged} = \left[w_p, w_r, w_{er}, w_{ep}\right] \tag{9}$$

$$\boldsymbol{weight}_{syn} = \left[w_p, w_r, w_a, w_{er}, w_{ep}\right] \tag{10}$$

As shown in (11-15), r_p is the reward value that indicates the difference between the actual and desired pitch angle. r_r is the reward value that indicates whether the desired condition is satisfied. r_{er} is the reward value, which represents the difference between the current reward value and the average value of the last 5 rewards. r_{ep} is the difference between the current pitch angle and the target pitch angle and the mean value of the historical difference, which gives a more comprehensive assessment of the current movement compared to the historical reward. r_a is the reward value indicating the velocity comparison in the process that the leg moves forward and the backward during synchronous motion. The action that the velocity of the leg in the moving backward process is

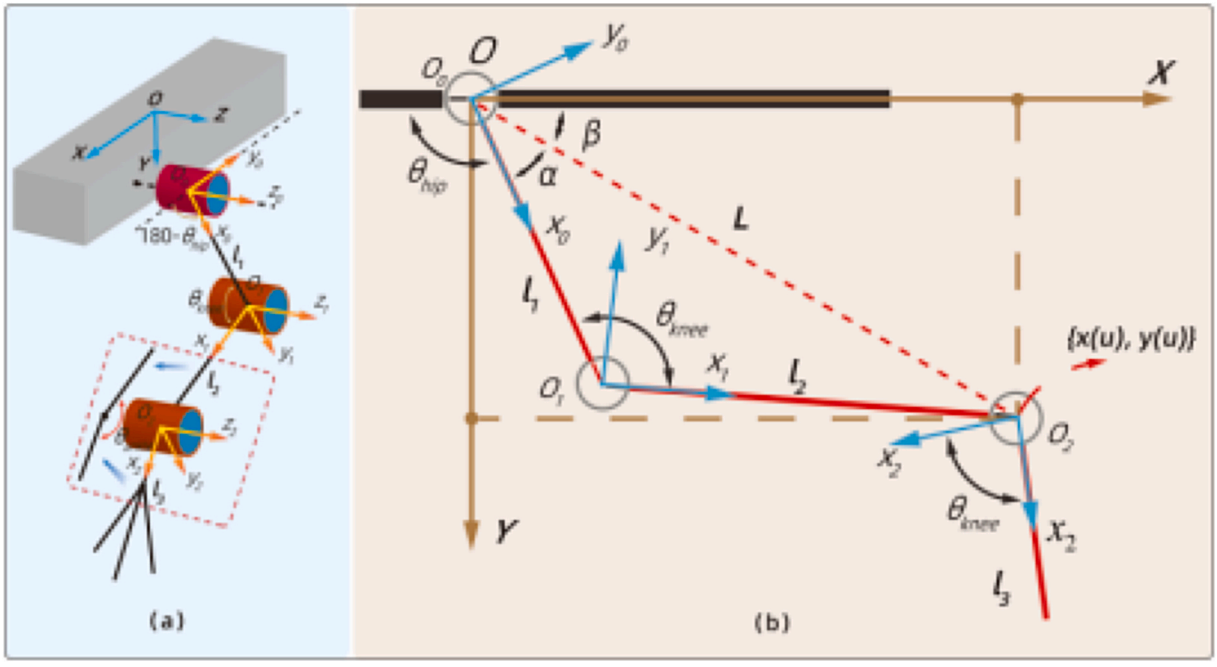

Fig. 4. Kinematics diagram of the beaver-like robot. (a) Kinematics model, (b) Geometric relation of inverse kinematics of hind limbs.

faster than that in the moving forward process is encouraged.

$$r_p = 3w_p e^{-\left(P_{cur} - P_{goal}\right)^2} \tag{11}$$

$$r_r = 2w_r \mathrm{int}\left(reach_{goal}\right) \tag{12}$$

$$r_{er} = w_{er} e^{R_{cur} - \mathrm{mean}(R_{hist})} \tag{13}$$

$$r_{ep} = w_{ep} e^{-\left|\left(P_{cur} - P_{goal}\right) - \left(\mathrm{mean}(P_{cur}) - P_{goal}\right)\right|} \tag{14}$$

$$r_a = w_a e^{back_{time} - pull_{time}} \tag{15}$$

The range of robot's joint angles is limited considering the training safety of pitch attitude stabilization control of the beaver-like underwater robot in swimming. A discrete action space is established based on the synchronous and alternate swimming trajectory of the beaver, as shown in (16).

$$\boldsymbol{action}_{space} = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9] \tag{16}$$

Note that each number in (16) represents a certain action loop, which is planned and stored into ***action*** in advance.

When the deep neural network outputs an action number from $\boldsymbol{action}_{space}$, the robot calculates the trajectory and foot's paddling velocity according to the action number and (17), which can adapt to the environment changes and improve the robot's bionic swimming characteristics and training efficiency in real time. act_{max} represents the max value of ***action***.

$$\boldsymbol{act}_{target} = \begin{cases} \boldsymbol{k}_{syn}\boldsymbol{action} + act_{max} \\ \boldsymbol{k}_{alter}\boldsymbol{action} + act_{max} \end{cases} \tag{17}$$

Equations (18) and (19) are the slopes of the chosen mapping functions based on the body structure of the beaver-like robot. As the robot has 7 joints, there are 7 elements in the equation.

$$\boldsymbol{k}_{syn} = [-3.2, +1.1, +3.4, -3.2, +1.1, +3.4, -0.8] \tag{18}$$

$$\boldsymbol{k}_{alter} = [-3.2, +1.1, +3.4, +3.2, -1.1, -3.4, -0.8] \tag{19}$$

Double Deep Q-learning is a reinforcement learning algorithm based on value learning, and its goal is to obtain the true value of the action value function Q by iterations shown below.

$$Q(s_t, a_t; \theta) \leftarrow Q(s_t, a_t; \theta) + \alpha\left[r + \gamma Q\left(s_{t+1}\underset{a_{t+1}}{\mathrm{argmax}} Q(s_{t+1}, a_{t+1}; \theta); w\right) - Q(s_t, a_t; \theta)\right] \tag{20}$$

where θ is the parameter of the current Q-network; w is the parameter of the target Q-network; s_t and s_{t+1} represent the state of the robot at the current moment and the next moment; a_t and a_{t+1} represent its actions at the two moments, α and γ are the learning rate and the decay factor respectively.

The neural network loss function update equation is given in (21). Its purpose is to reduce the difference between the predicted value and the real value to make the neural network convergence.

$$Loss = \left[\left(r + \gamma Q\left(s_{t+1}\underset{a_{t+1}}{\mathrm{argmax}} Q(s_{t+1}, a_{t+1}; \theta); w\right) - Q(s_t, a_t; \theta)\right)\right]^2 \tag{21}$$

Double Deep Q-learning combines dueling network structure to form dueling Double Deep Q-learning algorithm (D3QN). The D3QN algorithm used in this paper adopts the Dueling network structure composed of two fully connected layers (Van Hasselt and G. A and S. D., 2016; Z and S. T and H. M., 2016; Mnih et al., 2015; Kröse, 1995; Mnih et al., 2013). The network used to output the predicted Q value is called the current Q network, which is updated in real time during training. The target network used to output the target value has the same structure as the current network, and the update is delayed during training.

In the current Q network, its input is the current state vector and its output layer can be described in three aspects: the dominant function $A(s_t, a_t; \boldsymbol{\theta})$ to represent the dominant value of the current action relative to the state and is used to select the action, and the state value function $V(s; \boldsymbol{\theta})$ to represent the value of the current state. The two outputs are added together to obtain the action value function $Q(s_t, a_t; \boldsymbol{\theta})$ of the current state, which can efficiently estimate the action value and speed up training.

Based on the above analysis, the pitch attitude stabilization control of our robot is achieved by combining the synchronous and alternate swimming of the beaver with the reinforcement learning algorithm. Fig. 5 shows the proposed attitude stabilization control system consisting of two parts. The first part is the motion decision layer of the robot. It outputs the action index a_t according to the current input state s_t of the robot using the D3QN algorithm. The robot selects the trajectory with reinforcement learning, and then obtains the joint angles with inverse kinematics. The second part is the motion control layer of the robot. It maps the action sequence number to the bionic synchronous and alternate swimming gait according to the decision action and the bionic trajectory mapping function, as shown in (16), and the robot joints execute actions to swim in the pool and generate the next robot state.

The detailed information of the neural network is described in Table II.

3. Experiments and results

Our beaver-like underwater swimming robot is trained in a pool for reinforcement learning bionic swimming. The learning rate of the network is set to 10^{-4}; Mini-batch size is 32; the soft update parameter is 0.05; the initial value of the reward value decay factor is set to 0.95 and decreases by 0.005 with each training step; and the delayed update step is 80 to update the target network parameters. The minimum decay factor is 0.05 to ensure the robot's exploration of the environment. The robot is trained for a total of 100 episodes with 100 steps in each one.

3.1. .Experiments on pitch attitude control in swimming

We set up the synchronous and alternate swimming experiments to verify the stability of the pitch attitude control of the underwater robot. The control board of the robot is Nvidia Jetson NANO, and the operating system of the board is Linux Ubuntu 64. Fig. 6 shows the experiment on pitch stabilization control under synchronous swimming and Fig. 7 shows the experiment of pitch stabilization control under alternate swimming. Because the robot's environment is partially observable and the robot swimming environment is highly random, the loss value of deep network fluctuates greatly but is generally convergent.

We use the sliding average method to smooth the loss function curve with a sliding factor of 0.9 as shown in Fig. 8(a) and (b), and the loss function is stable after about 500th training. In terms of the effect of stable swimming in pitch attitude, reinforcement learning enables both synchronous and alternate swimming to achieve stable swimming within the desired pitch angle. However, alternate swimming has a significantly better learning effect due to its alternate swinging mode.

Fig. 9 shows the pitch angle for different training episodes, in which (a) is the pitch angle under synchronous swimming and (b) is the pitch angle under alternate swimming. Fig. 10 shows turn reward curves for training with synchronous and alternate swimming modes and average reward curve of the test in synchronous and alternate swimming modes.

In synchronous swimming mode, the robot can swim steadily within 40–50 steps after longer training episodes but will be difficult to return to the stable state within 5 steps if it encounters a large disturbance. This indicates that it needs longer training time to get more robust effect. In the alternate swimming mode, the robot can complete an episode in a stable pitch angle $[-1°, 1°]$ after 60 episodes of training, and its pitch

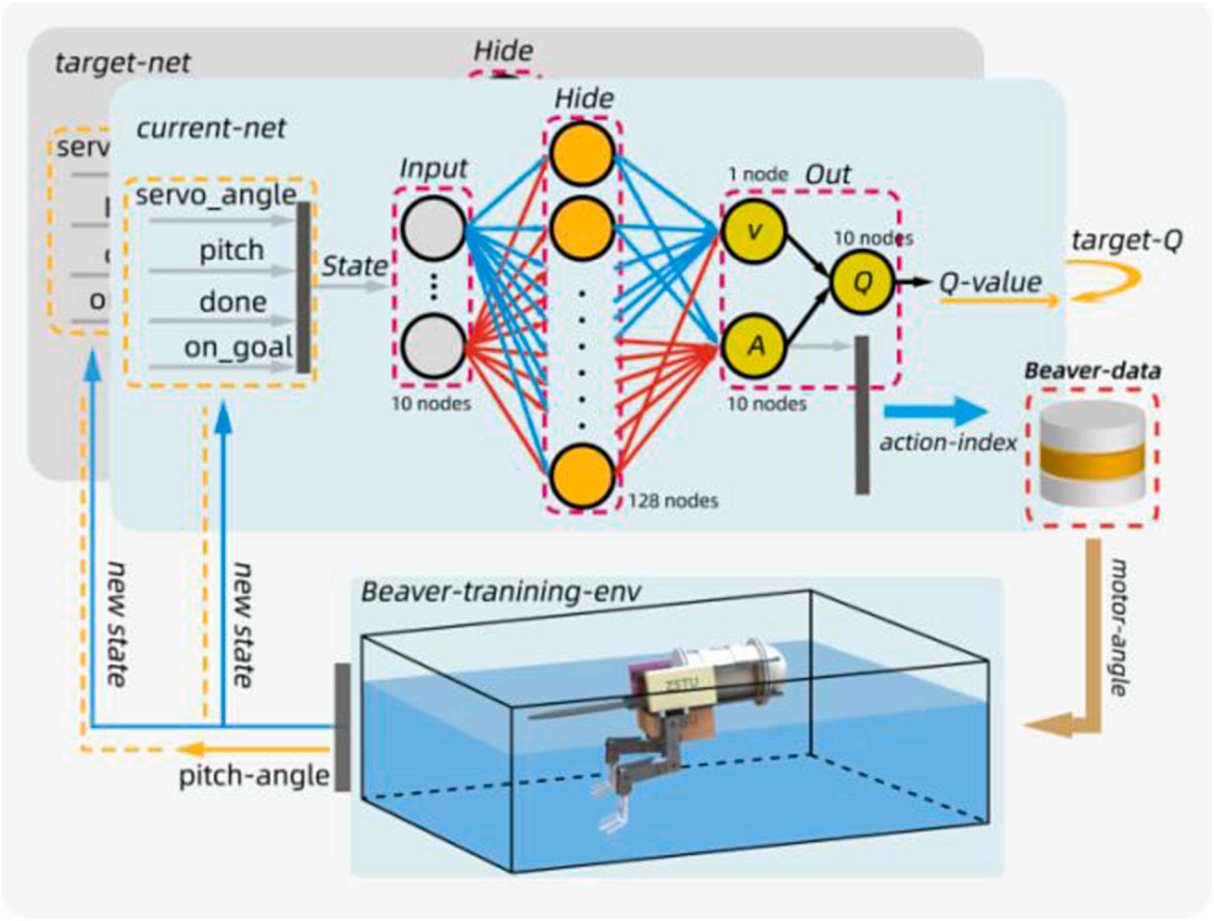

Fig. 5. Pitch attitude stabilization control system of beaver-like underwater robot.

Table 2
Detailed information of the neural network.

	Number of neurons	Activation function
Input layer	10	relu
Hidden layer	128	relu
Output layer	10	relu

angle is very stable. The episode reward values and average reward values in both modes start from lower values and rise to stable values. This indicates that the robot has achieved the knowledge of environment after training and is able to adapt to it.

3.2. Verification of pitch attitude control algorithm in the swimming

After training the beaver-like underwater robot, the pitch attitude control method using the trained deep network model was compared with that using only bionic swimming trajectory. In the synchronous and alternate swimming experiments, the pitch attitude angle of the robot was tested using 100 test steps. The pitch attitude change of the robot without reinforcement learning can reach to $[-3°, 16°]$ and $[-1°, 16°]$ in the synchronous and alternate swimming mode, which has a great impact on the swimming velocity of the robot.

However, with the reinforcement learning training, it can swim stably after 30 steps and the pitch angle can maintain $[-6°, 6°]$ and $[-1°, 1°]$, as shown in Fig. 11. The experiments demonstrate that the pitch

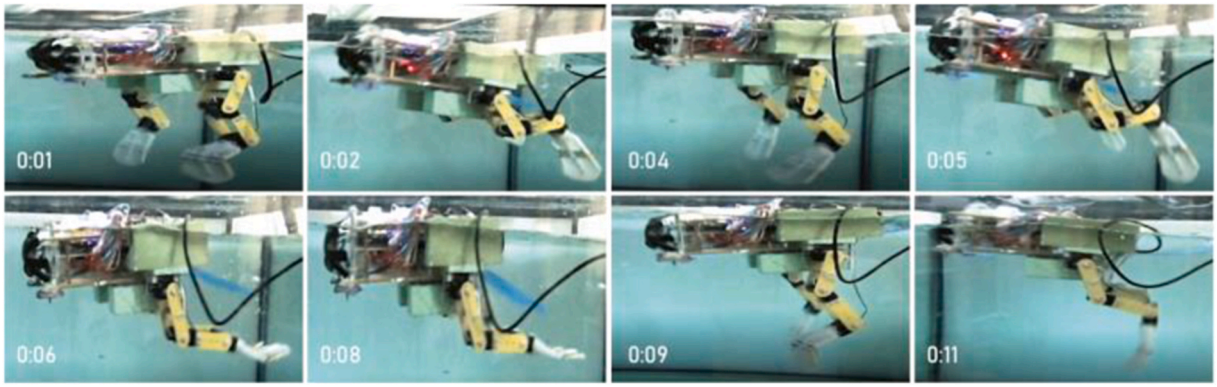

Fig. 6. Experiment on pitch stabilization control under synchronous swimming.

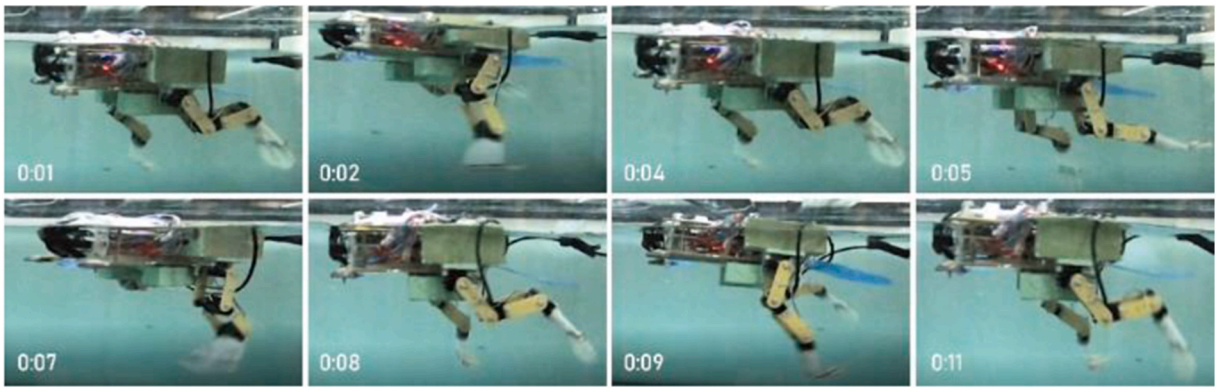

Fig. 7. Experiment of pitch stabilization control under alternate swimming.

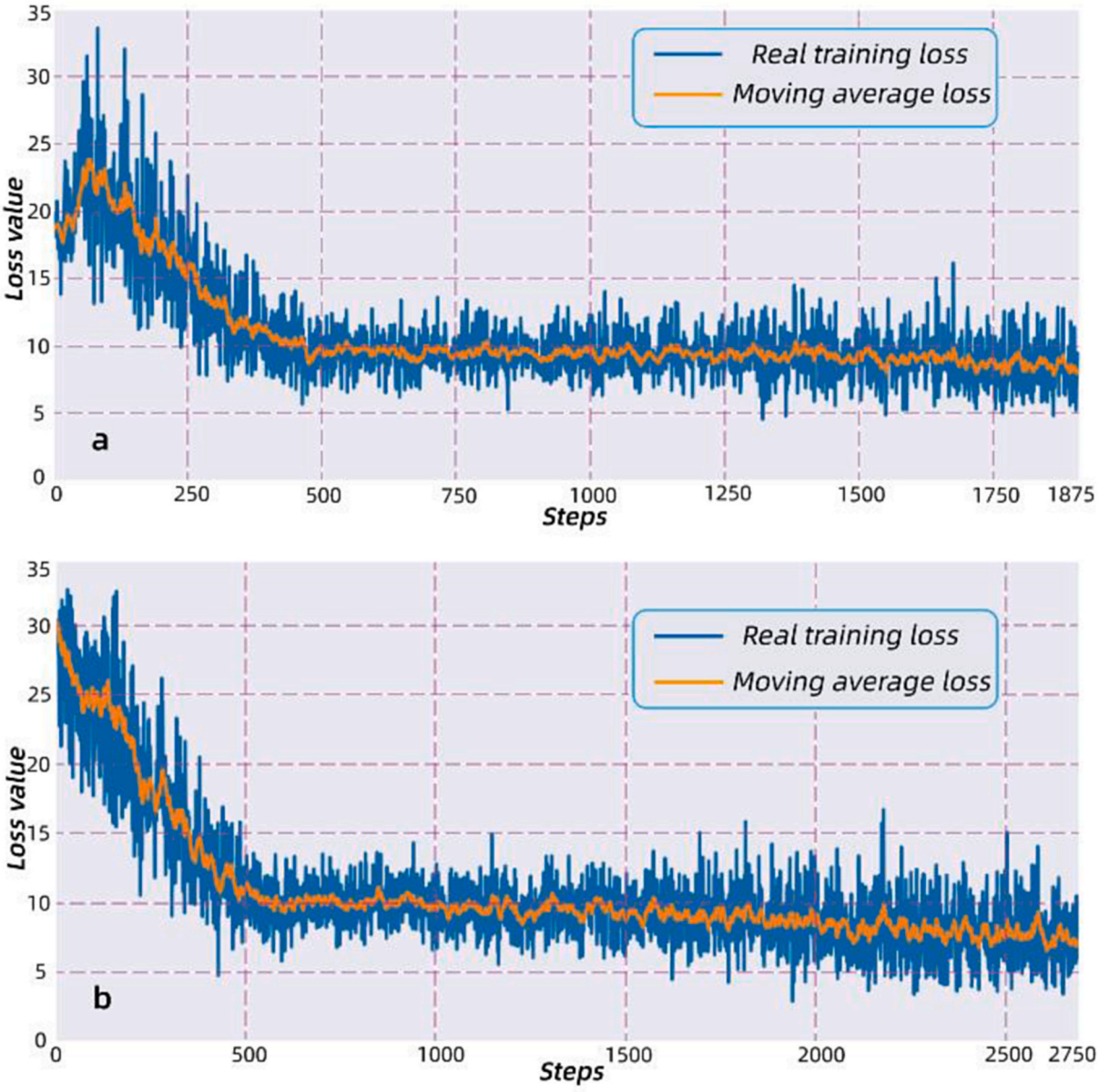

Fig. 8. Neural network loss curves. (a) Loss curves under synchronous swimming. (b) Loss curves under alternate swimming.

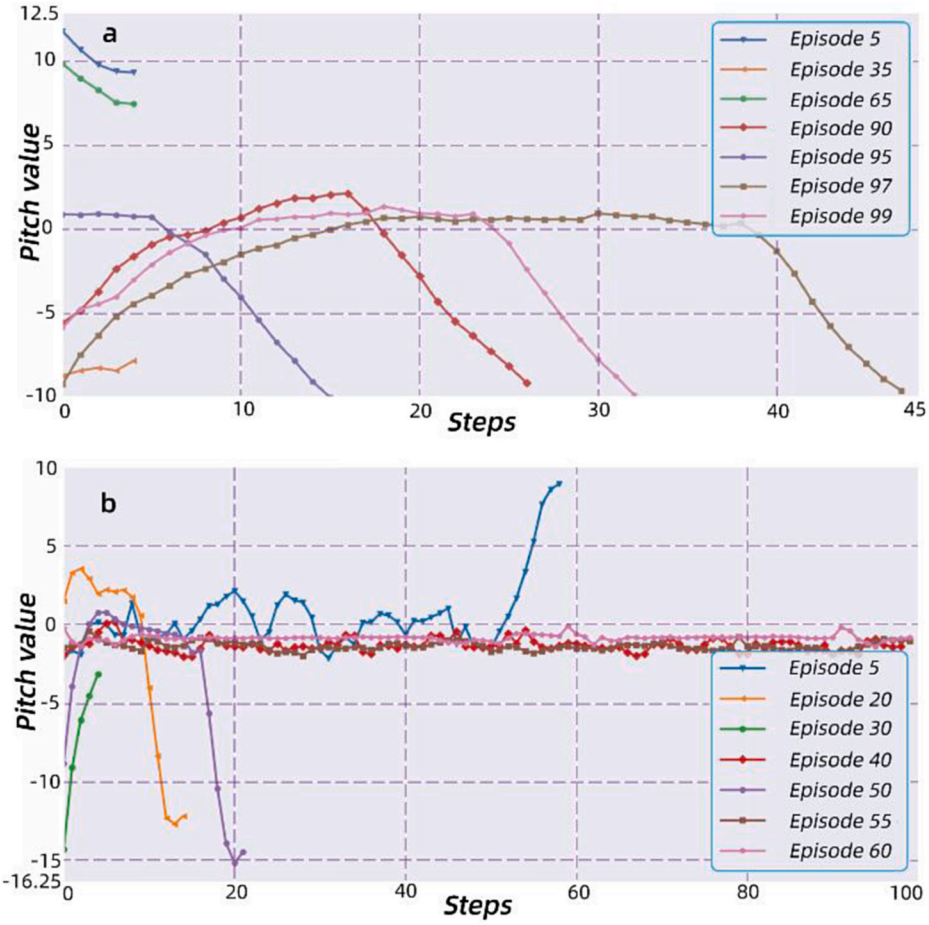

Fig. 9. Pitch angle for different training episodes. (a) Pitch angle under synchronous swimming. (b) Pitch angle under alternate swimming.

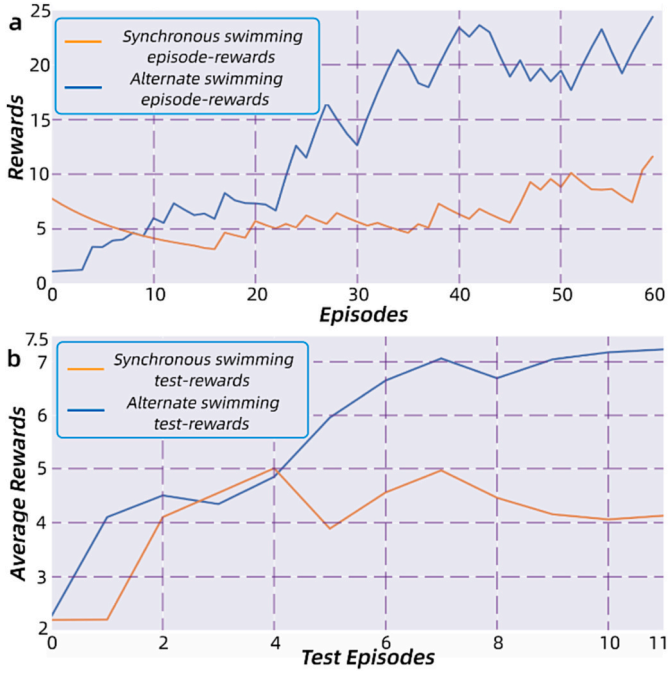

Fig. 10. Reward curves for training with synchronous and alternate swimming modes and average reward curve of the test in synchronous and alternate swimming mode.

attitude control method with reinforcement learning can well implement the pitch attitude stabilization control of our beaver-like underwater robot and has good adaptation to the environment. Therefore, this study provides a novel idea and method for the attitude control of underwater robots with disturbances.

4. Conclusions

This study focused on the problem about the unsteady attitude of underwater robot in swimming. The trajectory of the beaver's hind limbs was analyzed using deep reinforcement learning algorithm. It can establish an attitude stabilization control model of the beaver-like underwater robot based on the bionic trajectory of the beaver. The underwater robot was constructed and trained with the proposed method to realize the pitch attitude stabilization control. The experimental results show that the pitch attitude of the robot is stable within $\pm 6^{\circ}$ in the synchronous swimming mode and within $\pm 1^{\circ}$ in the alternate swimming mode.

This study has validated the correctness and efficacy of the proposed attitude stabilization control method, leveraging deep reinforcement learning. Through continuous learning, the robot has achieved self-learning of swimming and attitude stabilization, thereby eliminating the need for model construction. Furthermore, the proposed control method is rooted in bionic swimming trajectories, endowing the robot with swimming experience. This approach avoids the exploration of ineffective swimming actions, significantly enhancing learning efficiency and effectiveness. This study offers a novel perspective for non-model-based motion control of underwater robots.

During the reinforcement learning task, a discrete action space was

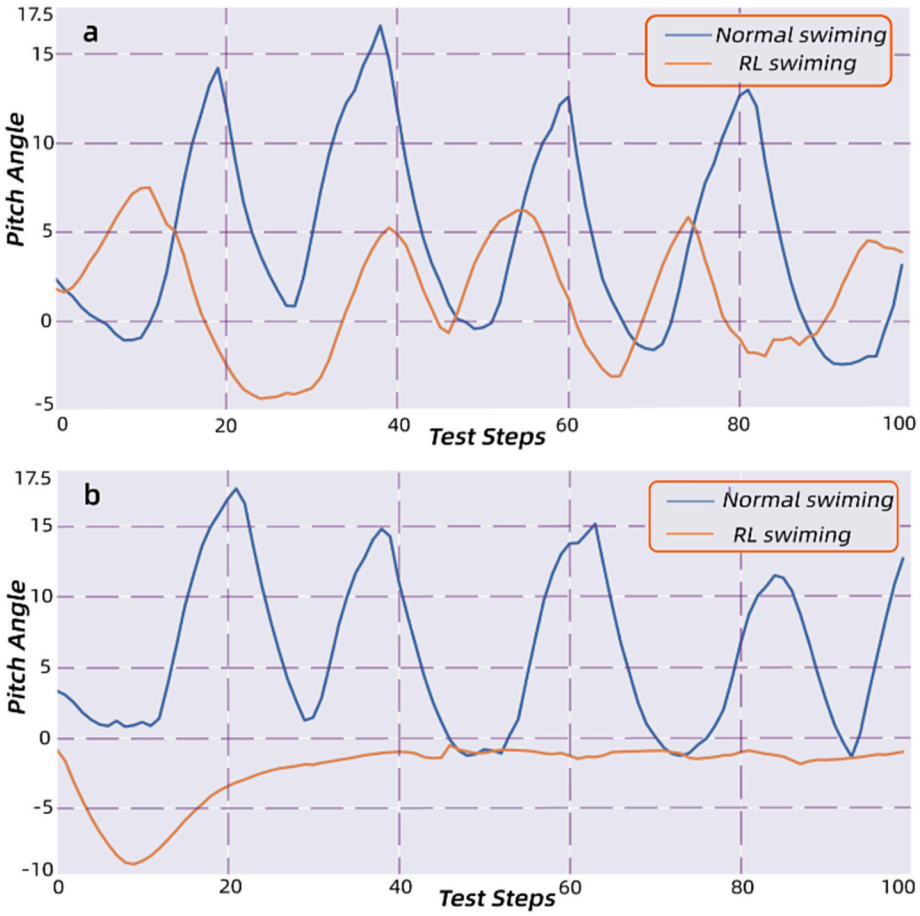

Fig. 11. Pitch angle of the robot under synchronous and alternate gait.

employed, resulting in limited robot performance. Additionally, the control strategy was developed in a calm aquatic environment, raising concerns about its performance in the presence of water currents. Future work will focus on integrating force sensors, vision sensors, and flow velocity sensors into the underwater robot to enhance its environmental perception capabilities. This integration aims to achieve robust control and improved learning efficiency in natural environments.

Notably, there are distinct differences in stability between alternating swimming and synchronized swimming. The larger waves generated by synchronized swimming have a significant impact on the surrounding environment. This study recognized that there are inherent differences between laboratory and natural environments, with the latter posing unique challenges for reinforcement learning. Future research directions will explore reinforcement learning in natural settings, aiming to address the complexities and dynamics introduced by real-world conditions.

A discrete action space is applied in the reinforcement learning task, and the performance of the robot is limited as a result. The control strategy is obtained in a calm aquatic environment, and there is no guarantee that its performance will remain good in the presence of water current. Our future work will be focused on how to integrate force sensors, vision sensors, and flow velocity sensor into our underwater robot to improve its environmental perception ability so that it can achieve highly control robustness and learning efficiency.

Funding

This work is financially supported by National Natural Science Foundation of China (Nos. 52275037, 51875528, and 41506116), Zhejiang Provincial Natural Science Foundation of China (No. LR24E050002), the Key Research and Development Project of Zhejiang Province (No. 2023C03015), the Emergency Management Research and Development Project of Zhejiang Province (No. 2024YJ026), the Key Research and Development Project of Ningxia Hui Autonomous Region (No. 2023BDE03002), and the Fundamental Research Funds of Zhejiang Sci-Tech University (24242088-Y).

CRediT authorship contribution statement

Gang Chen: Methodology, Funding acquisition, Conceptualization. **Zhihan Zhao:** Writing – review & editing, Investigation, Formal analysis. **Yuwang Lu:** Writing – original draft, Validation, Methodology, Investigation, Formal analysis, Data curation. **Chenguang Yang:** Writing – review & editing. **Huosheng Hu:** Writing – review & editing.

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.

Data availability

No data was used for the research described in the article.

References

B, W., Z, L., Q, L., 2020. Mobile robot path planning in dynamic environments through Globally guided reinforcement learning. IEEE Rob. Autom. Lett. 4 (5), 6932–6939. https://doi.org/10.1109/LRA.2020.3026638.

Chen, G., Jin, B., Chen, Y., 2018. Nonsingular fast terminal sliding mode posture control for six-legged walking robots with redundant actuation. Mechatronics 50, 1–15. https://doi.org/10.1016/j.mechatronics.2018.01.011.

Chen, G., et al., 2021. Hydrodynamic model of the beaver-like bendable webbed foot and paddling characteristics under different flow velocities. Ocean Eng. 234, 109179 https://doi.org/10.1016/j.oceaneng.2021.109179.

Chen, B., et al., 2022a. Fully body visual self-modeling of robot morphologies. Sci. Robot. 7 (68), eabn1944. https://doi.org/10.1126/scirobotics.abn1944.

Chen, G., et al., 2022b. Reinforcement learning control for the swimming motions of a beaver-like, single-legged robot based on biological inspiration. Robot. Autonom. Syst. 154, 104116 https://doi.org/10.1016/j.robot.2022.104116.

Chen, G., et al., 2022c. Design of beaver-like hind limb and analysis of two swimming gaits for underwater narrow space exploration. J. Intell. Rob. Syst. 104 (4), 65. https://doi.org/10.1007/s10846-022-01610-7.

Chen, G., et al., 2023a. Design and control of a novel bionic Mantis shrimp robot. IEEE/ASME Trans. Mechatron. 1–10. https://doi.org/10.1109/TMECH.2023.3266778.

Chen, G., et al., 2023b. Swimming modeling and performance optimization of a fish-inspired underwater vehicle (FIUV). Ocean Eng. 271, 113748 https://doi.org/10.1016/j.oceaneng.2023.113748.

Cohen, A., Zarrouk, D., 2020. The AmphiSTAR high speed amphibious sprawl tuned robot: design and experiments. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, pp. 6411–6418.

Crespi, A., et al., 2013. Salamandra robotica II: an amphibious robot to study salamander-like swimming and walking gaits. IEEE Trans. Robot. 29 (2), 308–320. https://doi.org/10.1109/TRO.2012.2234311.

Gang, C., Bo, J., 2016. Methods to resist water current disturbances for underwater walking robots. Mar. Technol. Soc. J. 1 (50), 73–87. https://doi.org/10.4031/MTSJ.50.1.5.

Karakasiliotis, K., Ijspeert, A.J., 2009. Analysis of the terrestrial locomotion of a salamander robot. In: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, pp. 5015–5020.

Kim, B., et al., 2013. Operating software for a multi-legged subsea robot CR200. In: 2013 MTS/IEEE OCEANS-Bergen, Bergen, Norway, pp. 1–5.

Kröse, B.J.A., 1995. Learning from delayed rewards. Robot. Autonom. Syst. 15 (4), 233–235. https://doi.org/10.1016/0921-8890(95)00026-C.

Li, Z., et al., 2018. Turning characteristics of biomimetic robotic fish driven by two degrees of freedom of pectoral fins and flexible body/caudal fin. Int. J. Adv. Rob. Syst. 15 (1), 1729881417749950 https://doi.org/10.1177/1729881417749950.

Li, Y., Sato, H., Li, B., 2021. Feedback altitude control of a flying insect-computer hybrid robot. IEEE Trans. Robot. 37 (6), 2041–2051. https://doi.org/10.1109/TRO.2021.3070983.

Ma, X., Wang, G., Liu, K., 2022. Design and optimization of a multimode amphibious robot with propeller-leg. IEEE Trans. Robot. 38 (6), 1–14. https://doi.org/10.1109/TRO.2022.3182880.

Mnih, V., et al., 2013. Playing atari with deep reinforcement learning. arXiv preprint 5602, 1312. https://doi.org/10.48550/arXiv.1312.5602.

Mnih, V., et al., 2015. Human-level control through deep reinforcement learning. Nature (London) 518 (7540), 529–533. https://doi.org/10.1038/nature14236.

Pu, H., et al., 2012. Experimental study on oscillating paddling gait of an eccentric paddle mechanism. In: 2012 IEEE International Conference on Robotics and Biomimetics, Guangzhou, China, pp. 187–192.

Raj, A., Thakur, A., 2016. Fish-inspired robots: design, sensing, actuation, and autonomy—a review of research. Bioinspiration Biomimetics 11 (3), 031001. https://doi.org/10.1088/1748-3190/11/3/031001.

Shen, X., Zheng, Y., Zhang, R., 2020. A hybrid forecasting model for the velocity of hybrid robotic fish based on back-propagation neural network with genetic algorithm optimization. IEEE Access 8, 111731–111741. https://doi.org/10.1109/ACCESS.2020.3002928.

Shintake, J., et al., 2020. Bio-inspired tensegrity fish robot. In: 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, pp. 2887–2892.

Sun, Y., et al., 2012. Modeling the rotational paddling of an ePaddle-based amphibious robot. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, pp. 610–615.

Van Hasselt, H., A, G., D, S., 2016. Deep reinforcement learning with double Q-learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, California, USA.

VCG.COM, 2022a. A beaver on the shore [Online], Available: https://www.vcg.com/creative/1224424789.

VCG.COM, 2022b. A beaver swims in the water [Online], Available: https://www.vcg.com/creative/1300783494.

Wang, Z., et al., 2022. Hybrid bipedal locomotion based on reinforcement learning and heuristics. Micromachines 13 (10), 1688. https://doi.org/10.3390/mi13101688.

Woo, J., Yu, C., Kim, N., 2019. Deep reinforcement learning-based controller for path following of an unmanned surface vehicle. Ocean Eng. 183, 155–166. https://doi.org/10.1016/j.oceaneng.2019.04.099.

Z, W., S. T and H. M., 2016. Dueling network architectures for deep reinforcement learning. In: International Conference on Machine Learning. PMLR, pp. 1995–2003.

Zarrouk, D., et al., 2013. STAR, a sprawl tuned autonomous robot. In: 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, pp. 20–25.

Zarrouk, D., Fearing, R.S., 2015. Controlled in-plane locomotion of a hexapod using a single actuator. IEEE Trans. Robot. 31 (1), 157–167. https://doi.org/10.1109/TRO.2014.2382981.

Zarrouk, D., Yehezkel, L., 2018. Rising star: a highly reconfigurable sprawl tuned robot. IEEE Rob. Autom. Lett. 3 (3), 1888–1895. https://doi.org/10.1109/LRA.2018.2805165.

Zhang, Q., 2020. Deep interactive reinforcement learning for path following of autonomous underwater vehicle. IEEE Access 8 (2020), 24258–24268. https://doi.org/10.1109/ACCESS.2020.2970433.

Zhang, W., Lin, L., 2012. Reinforcement learning controller based attitude stabilization for bionic underwater robots. Comput. Meas. Control 20 (11), 3063–3065. https://doi.org/10.13374/j.issn1001-053x.2012.01.014.shu.

Zhang, X., et al., 2022. Mechanism analysis of cheetah's high-speed locomotion based on digital reconstruction. Biomimetic Intellig. Robot. 2 (1), 100033 https://doi.org/10.1016/j.birob.2021.100033.

Zhu, J., et al., 2019. Tuna robotics: a high-frequency experimental platform exploring the performance space of swimming fishes. Sci. Robot. 4 (34), eaax4615. https://doi.org/10.1126/scirobotics.aax4615.

Gang Chen (Member, IEEE) received the Ph.D. degree in mechatronic engineering from Zhejiang University, Hangzhou, China, in 2014. He is currently a professor at the School of Mechanical Engineering, Zhejiang Sci-Tech University, working on biomimetic underwater robots.

Zhihan Zhao (Student Member, IEEE) received the BSc degree in mechatronic engineering from Zhejiang Sci-Tech University, Hangzhou, China, in 2021. He is currently a postgraduate student in the School of Mechanical Engineering, Zhejiang Sci-Tech University, working on biomimetic underwater robots and reinforcement learning.

Yuwang Lu (Student Member, IEEE) received the BSc degree in mechatronic engineering from Zhejiang Sci-tech University, Hangzhou, China, in 2019. He received the MSc degree in mechatronic engineering from Zhejiang Sci-tech University, Hangzhou, China, in 2022. His research interests are focused on biomimetic underwater robots and reinforcement learning.

Chenguang Yang (Senior Member, IEEE) received the Ph.D. degree in control engineering from the National University of Singapore, Singapore, in 2010. He received the postdoctoral training in human robotics from the Imperial College London, London, U.K. His research interest lies in human robot interaction and intelligent system design.

Huosheng Hu (Life Senior Member, IEEE) received the Ph.D. degree in robotics from the University of Oxford in U.K, in 1992. He is currently a professor in the School of Computer Science & Electronic Engineering in University of Essex in the United Kingdom. His research interests include mobile robotics, human-robot interaction, embedded systems, data fusion, learning algorithms, mechatronics, and pervasive computing.