J Bionic Eng 18 (2021) 171–183
DOI: https://doi.org/10.1007/s42235-021-0008-0

Journal of Bionic Engineering
http://www.springer.com/journal/42235

Bioinspired Closed-loop CPG-based Control of a Robot Fish for Obstacle Avoidance and Direction Tracking

Jiayong Chen[1], Bo Yin[2], Chengcai Wang[3], Fengran Xie[4], Ruxu Du[1], Yong Zhong[1*]

1. *Shien-Ming Wu School of Intelligent Engineering*, *South China University of Technology*, *Guangzhou* 511442, *China*
2. *Institute of Mechanics*, *Chinese Academy of Sciences*, *Beijing* 100190, *China*
3. *China Academy of Electronics and Information Technology*, *Beijing* 100086, *China*
4. *Precision Engineering Research Center*, *Shenzhen Institutes of Advanced Technology*, *Chinese Academy of Sciences*, *Shenzhen* 518055, *China*

Abstract

This paper presents a study on bioinspired closed-loop Central Pattern Generator (CPG) based control of a robot fish for obstacle avoidance and direction tracking. The biomimetic robot fish is made of a rigid head with a pair of pectoral fins, a wire-driven active body covered with soft skin, and a compliant tail. The CPG model consists of four input parameters: the flapping amplitude, the flapping angular velocity, the flapping offset, and the time ratio between the beat phase and the restore phase in flapping. The robot fish is equipped with three infrared sensors mounted on the left, front and right of the robot fish, as well as an inertial measurement unit, from which the surrounding obstacles and moving direction can be sensed. Based on these sensor signals, the closed-loop CPG-based control can drive the robot fish to avoid obstacles and to track designated directions. Four sets of experiments are presented, including avoiding a static obstacle, avoiding a moving obstacle, tracking a designated direction and tracking a designated direction with an obstacle in the path. The experiment results indicated that the presented control strategy worked well and the robot fish can accomplish the obstacle avoidance and direction tracking effectively.

Keywords: biomimetic robot fish, closed-loop control, Central Pattern Generator (CPG), obstacle avoidance, direction tracking

1 Introduction

Over millions of years of evolution, fishes have possessed excellent swimming capabilities, such as highly efficient propulsion, high swimming speeds, and *etc*., which have inspired engineers and scientists around the world to develop biomimetic robot fish. In 1994, researchers in MIT designed the first fish-like robot, called Robo Tuna[1]. Since then, many different robot fishes have been designed and built[2–5]. In general, the propulsion modes of fish and other aquatic animals can be categorized into two types: Body and Caudal Fin (BCF) propulsion and Medium and Paired Fin (MPF) propulsion[6]. Although each propulsion type has its merits, most existing robot fishes employ BCF propulsion[7]. From the structure point of view, existing robot fish designs can be categorized into four types: single joint design, multi-joint design, smart material-based design, and wire-driven design[8]. The single joint design uses one motor[9,10] and hence, may require additional structure to mimic fish swim and increase the propulsion efficiency. The multi-joint design can mimic fish flapping motion effectively, such as the dolphin robot from Yu *et al.*[11], and autonomous multi-joint robot fish G9[12], but the increase in the number of joints increases the complexity of synchronization control. Various smart materials, such as ionic polymer-metal composites[13] and shape memory alloys[14] have been used for developing robot fishes. With suitable control, these materials can generate flapping motions, albeit low efficiency. Unlike other methods, the one with wire-driven design is better at mimicking fish flapping motions with fewer driving actuators and has been proved of high efficiency and easy implementation[15].

It is noted that in the early stage of robot fish research, the focuses were mainly on the modeling and prototyping. Robot fish control was typically done by human operator through a joystick. The next stage of

***Corresponding author:** Yong Zhong
E-mail: zhongyong@scut.edu.cn

robot fish research requires the robot being able to perceive its surrounding environment and adapt it accordingly. In this case, closed-loop control is essential. According to literatures, there have been some studies on robot fish closed-loop control. For example, Shin *et al.* developed a robot fish that used a fuzzy neural network for obstacle recognition[16]. Shin *et al*. used fuzzy systems to generate an obstacle avoidance path[17]. Deng *et al.* designed a simple 3D-printed robot fish that can detect obstacles and move away adaptively using neuro-fuzzy control[18]. Verma *et al.* used an adaptive control approach for robot fish closed-loop orientation control[19]. However, these robot fish just avoided a simple static obstacle with a stiff flapping.

Central Pattern Generator (CPG) is the natural way for animal to generate rhythmic motion control signals and has been used to control biomimetic robots[20,21]. The combination of closed-loop and CPG on the robot fish is a novel way for improving the propulsion efficiency and maneuverability. For example, Yu *et al.* developed a CPG-based multi-joint robot fish with visual feedback control for avoiding obstacles and searching for tasks[22]. Korkmaz *et al.* tried to combine CPG, fuzzy logic and sensory feedback to control a robot fish with two links and conducted yaw and pitch control experiments[23]. Wang and Xie advanced a closed-loop CPG controller for yaw and roll control of a boxfish-like robot[24]. Zhang *et al*. combined the advantages of insect wings and fish fins to achieve an agile robotic fish and designed a CPG model for an obstacle avoiding[25]. Among these robot fish, the active bodies are single-joint or multi-joint designs, which may increase control difficulties. However, there have been few studies of closed-loop CPG control on robot fish with a wire-driven active body. In addition, a systematic design of closed-loop CPG-based control for static and moving obstacles avoidance and direction tracking is still missing.

Our team has been working on robot fish for many years[15]. In 2016, Zhong *et al.* designed a wire-driven robot fish with an active body and a compliant tail, which can mimic the natural flapping movement of fish effectively[26,27]. In 2019, Xie *et al.* developed a simple CPG controller with just four parameters, namely the flapping amplitude, the flapping angular velocity, the flapping offset, and the time ratio between the beat phase and the restore phase. This CPG controller is not only simple but also effective in controlling both cruising and turning[28,29].

The objective of this paper is to develop a closed-loop CPG-based control for obstacle avoidance and direction tracking. The rest of the paper is organized as follows. Section **2** presents the design of the robot fish. Section **3** presents the closed-loop CPG-based control system. Section **4** presents the experimental results. Finally, section **5** contains conclusions and future research topics.

2 The design of the robot fish

2.1 Mechanical design

We design a biomimetic wire-driven robot fish that is similar in size and mimics the behavior of real fish. The structure of the robot fish and its major components are depicted in Fig. 1. The robot fish comprises three parts, namely a rigid head, a wire-driven active body, and a compliant tail. It is 473 mm long, 233 mm wide, and 127 mm high. The rigid head is used to install electronic components, such as circuit board, battery and sensors. The redesigned robot fish is completely waterproof for ensuring the effective protection of electronic components. The robot fish has infrared (IR) sensors. However, IR rays cannot pass through opaque objects. To resolve this problem, transparent photosensitive resin materials is used as the 3D printed material for the head shell; and the inner and outer surfaces of the shell are polished to improve transparency. The sealing rings and bearings are mounted to the shaft, which is passed through the shell to ensure water is kept out when the shaft is rotating. A pair of pectoral fins attached to the shaft control the ascent and descent of the robot fish. On the top of the robot fish, an opening blocked up by a rubber plug is reserved for charging the battery and downloading the program. The active body has four links in which two spring plates are embedded. The active body is covered with a soft watertight skin to prevent water from entering the robot fish body. The compliant tail is fixed onto the last link.

The design and kinematic analysis of the wire-driven active body and the compliant tail are detailed in Ref. [26]. As depicted in Fig. 2, a pair of wires

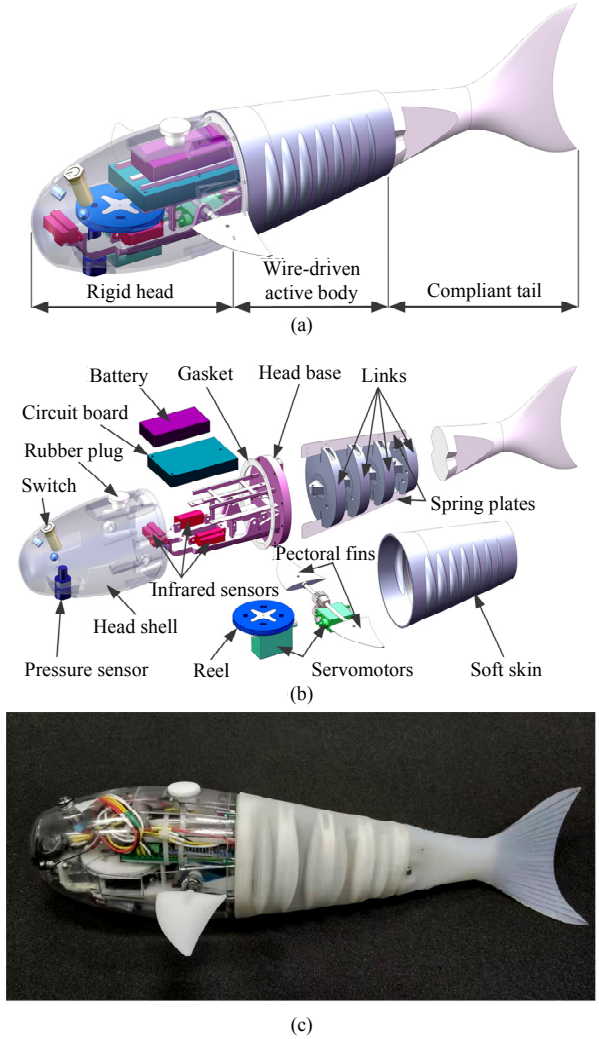

Fig. 1 Our robot fish overview. (a) The overall appearance; (b) disassembly diagram of the robot fish structure without showing the pair of wires; (c) the prototype.

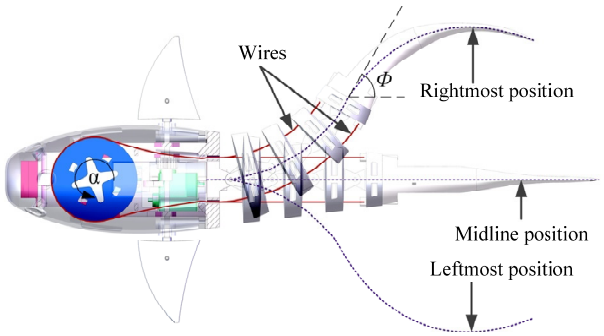

Fig. 2 Wire-driven active body and compliant tail overview with three states: midline, rightmost, and leftmost positions.

are passed through the interior of the head and the first three links and are fastened at the last link. The other end of the wires is mounted on the reel. When the reel driven by the servomotor rotates counter clockwise, the right wire is pulled, whereas the other is elongated, which drives the active body into an arc. The compliant tail deforms because of water resistance. The rotation angle of servomotor (α) corresponding to the bending angle of the active body (Φ) is depicted in Fig. 2.The relationship between the angles, α and Φ is expressed as:

$$\Phi = 2N \sin^{-1} \frac{\pi \alpha r}{180 N d}, \tag{1}$$

where N is the number of the links, r is the radius of the reel, and d is the distance between two wires.

In our design, N is 4, r is 34 mm, and d is 36 mm. The term inside the arcsine function of Eq. (1) is small; thus, Eq. (1) can be simplified as:

$$\Phi = \frac{2r}{d} \alpha. \tag{2}$$

2.2 Electronics

The electronic devices used in the robot fish are depicted in Fig. 1b. They mainly include the control system, power source, wireless transmission module, servomotors, and sensors. Using a 7.4 V 1500 mAH Ni−H battery as the power supply, it supports the robot for nearly 2 h of nonstop cruising according to our test. The STM32F103 microcontroller processes data and produces instructions. The Radio Frequency (RF) wireless module, E62-433T20D, is adopted for communication between the robot fish and the computer or joystick. The HS-5086WP servomotor drives the pectoral fin, whereas the SAVOX SW-1210SG servomotor drives the active body. A MPU6050 Inertial Measurement Unit (IMU) is used to obtain the robot fish's pitch and yaw angles.

Three GP2Y0A21 IR sensors are placed in the front, left, and right of the head base. Its range in air is 100 mm – 800 mm. However, the IR ray propagation medium in the robot is different from that in air because of the influence of water and the semitransparent head shell. Therefore, we tested the voltage output characteristic of the IR sensor in water before conducting formal experiments. As depicted in Fig. 1b, the left and right IR sensors are bilaterally symmetrical; thus, measurement of only one sensor's output characteristic is required. For

example, for the front and left IR sensors, as depicted in Fig. 3, two sets of data are tested with a certain gradient. The blue and red lines denote the front and left IR sensors' output characteristics, respectively. The results indicate that: (1) The sensors' recognition distance range in water is approximately 100 mm – 600 mm, which is suitable for obstacle avoidance. (2) The output characteristics of the front and left IR sensors are disparate. The main reasons may be the different distances between the sensors and the head shell, and the different arcs of the shell facing the sensors. These characteristics render the IR ray propagation path diverse.

3 Closed-loop CPG-based control structure design

3.1 CPG control

It is well known that fish swims with a rhythmic locomotion. Such a rhythmic movement can be reproduced using a CPG[21]. By observing and analyzing the motion of a black carp, our team proposed an improved CPG model for wire-driven robot fish to realize flexible fish-like locomotion[28]. The improved CPG model is based on Ijspeert's CPG model, which has been successfully applied to a salamander robot to generate rhythmic motion[30,31]. The equations of the improved CPG model used in this paper is as:

$$\ddot{b} = k_b(0.25k_b(B-b)-\dot{b}), \tag{3}$$

$$\ddot{m} = k_m(0.25k_m(M-m)-\dot{m}), \tag{4}$$

$$\dot{\phi} = [\frac{(1+R)^2}{4R} - \frac{R^2-1}{4R}\text{sign}(\sin\phi)]\omega, \tag{5}$$

$$\alpha = b + m\cos(\phi), \tag{6}$$

$$\beta = b + m\sin(\phi), \tag{7}$$

$$\text{sign}(\lambda) = \begin{cases} 1, \lambda > 0 \\ 0, \lambda = 0 \\ -1, \lambda < 0 \end{cases}, \tag{8}$$

where b is the offset state, B is the high-level control command of offset, k_b is a positive constant representing how fast b converges to B; m is the amplitude state, M is the high-level control command of amplitude, k_m is a positive constant representing how fast m converges to M; ϕ is the phase state; R is the time ratio between two phases forming one turning period, restore phase (t_r), and beat phase (t_b); ω is the high-level control command of angular velocity; α is the rotation angle of the servomotor; and β is an intermediate variable. Among these, M, ω, B, and R are the four input parameters of this model, and α is the only output parameter for controlling the flapping patterns of the active body.

Through changing the input parameters, the robot fish can cruise or turn with varied accelerations, velocities, and turning radiuses. For example, the robot fish cruises when $M \neq 0$, $\omega \neq 0$, $B = 0$, and $R = 1$, whereas it turns when $M \neq 0$, $\omega \neq 0$, $B \neq 0$, and $R > 1$.

3.2 Closed-loop CPG-based control

In this section, we presented a closed-loop CPG-based control system for the wire-driven robot fish, as depicted in Fig. 4, to achieve obstacle avoidance, direction tracking, and autonomous control of the pectoral fin.

3.2.1 Closed-loop control for obstacle avoidance

The robot fish should avoid obstacles automatically under unknown circumstances to improve locomotion ability. In Fig. 3, the output voltage of the IR sensors is a continuous signal. We set up a voltage threshold to convert the continuous signal to a digital signal. When the output voltage is lower than the threshold, the obstacle is sufficiently far from the robot, and the obstacle signal is set as 0. Conversely, when the obstacle is close to the robot, the signal is set as 1. In addition, due to the

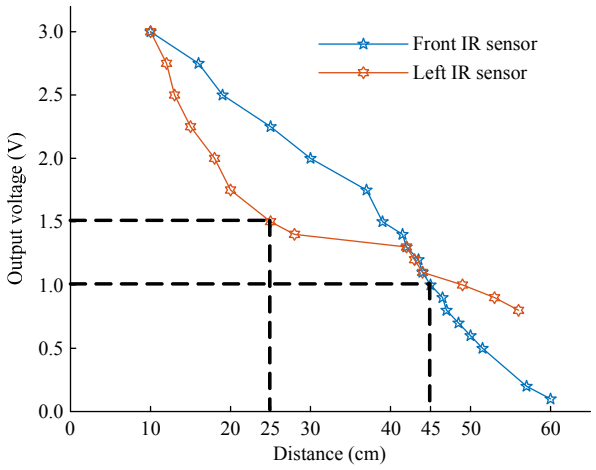

Fig. 3 The voltage output characteristics of the IR sensors in water.

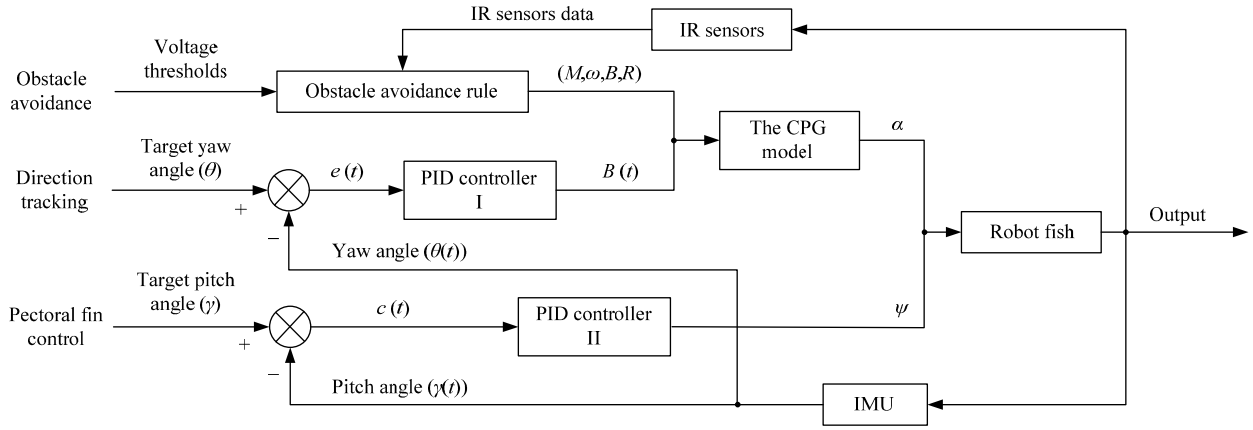

Fig. 4 The closed-loop control structure of the robot fish.

inertia of the robot fish while swimming, the recognition distance of the front IR sensor should be greater than that of the lateral sides for reserving sufficient distance and avoiding collisions. Based on these analyses, we set the voltage thresholds of the front and lateral IR sensors as 1 V and 1.5 V respectively, and the corresponding recognition distances are 450 mm and 250 mm respectively, as indicated by the dotted lines in Fig. 3.

The obstacle avoidance control design is based on the sensory feedback data and CPG model. We propose an obstacle avoidance rule (Table 1) for enabling automatic swimming of the robot fish. Here, F_{IR}, L_{IR}, and R_{IR} represent the obstacle signals on the front, left, and right sides, respectively, of the robot. The obstacle situations can be categorized into eight types, and we establish corresponding responses for each situation. For instance, when F_{IR}, L_{IR}, and R_{IR} are both set as 0, the robot fish cruises forward. Note that in situation 7, the obstacles are detected in three directions. Normally, this rarely happens in an open area of water, so we just choose to turn right.

As depicted in Fig. 4, the input signal of the first closed loop is the voltage threshold of three IR sensors, and the feedback signal is the output voltage of IR sensors, which contains obstacle information. The CPG input parameters, M, ω, B, and R, are updated in real-time by inputting the aforementioned two signals into the obstacle avoidance rule. Subsequently, the rotation angle of the servomotor (α) is changed based on the output of the CPG model, which enables the robot fish to execute different flapping patterns.

Table 1 Robot fish obstacle avoidance rule

Situations	F_{IR}	L_{IR}	R_{IR}	Response	(M, ω, B, R)
0	0	0	0	Cruise	(20, 3π, 0, 1)
1	0	0	1	Turn left	(30, 4π, −30, 1.5)
2	0	1	0	Turn right	(30, 4π, 30, 1.5)
3	1	0	0	Turn left	(30, 4π, −30, 1.5)
4	0	1	1	Cruise	(20, 3π, 0, 1)
5	1	0	1	Turn left	(30, 4π, −30, 1.5)
6	1	1	0	Turn right	(30, 4π, 30, 1.5)
7	1	1	1	Turn right	(30, 4π, 30, 1.5)

3.2.2 Closed-loop control for direction tracking

In our design, an IMU is fit parallel to the robot fish on the circuit board for obtaining yaw and pitch angles, which contributes to direction tracking and the control of pectoral fin. In Fig. 4, the input signal of the second closed loop is the target yaw angle (θ), which is the forward orientation of the robot fish as it cruises steadily. The angle reference coordinate system (0°) is the direction of the robot fish's head when the robot is energized, and the counterclockwise direction is positive. The feedback signal is the yaw angle ($\theta(t)$), that is the swimming direction of the robot fish. However, the real-time yaw angle sampled from the IMU is the direction of the head, and the head is constantly swinging along with the approximate sinusoidal oscillation of the active body; thus, it is difficult to determine which real-time yaw angle is the swimming direction of the robot fish. To address this problem, we set the CPG parameters M, ω, and R to be 20, 2π, and 1, respectively, which ensures that the robot fish's swing is a real sine

curve. Then, the yaw angle of the robot fish ($\theta(t)$) is obtained by calculating the angles sampled from the IMU at the same time interval and calculating the mean value in a flapping period as:

$$\theta(t) = \frac{1}{N}\sum_{i=1}^{N}\theta_i(t), \tag{9}$$

$$N = \frac{2\pi}{\omega}f, \tag{10}$$

where f is the sampling frequency of the IMU, and its value is 100 Hz, N is the total number of data sampling in one flapping period, and $\theta_i(t)$ (i = 1, 2, …, N) is the yaw angle of the ith sampling at the same time interval.

The deviation signal ($e(t)$), which is the input of PID controller I, is calculated as:

$$e(t) = \theta - \theta(t). \tag{11}$$

Through PID controller I, $B(t)$, one of the CPG model input parameters, is obtained using the following expression:

$$B(t) = K_p \times e(t) + K_i \times \int_0^t e(\tau)\mathrm{d}\tau + K_d \times \frac{\mathrm{d}e(t)}{\mathrm{d}t}, \tag{12}$$

$$B(t) = \begin{cases} 40, & B(t) \geq 40 \\ B(t), & -40 < B(t) < 40, \\ -40, & B(t) \leq -40 \end{cases} \tag{13}$$

where $B(t)$ is the offset of the CPG model, which is the only altered parameter when the robot fish tracks a designated direction. Here, K_p, K_i, and K_d denote the proportional, integral and differential coefficients, respectively, of the PID controller I. Before conducting the experiments, we performed tests to observe the swimming performance of the robot under different PID parameters. According to the analysis of results, we adopted proportion and differential control to improve the performance of the robot fish, and K_p, K_i, and K_d are set to 0.7, 0, and 0.05, respectively. Due to the limitation of the flapping amplitude of the active body, the range of $B(t)$ is limited to [−40, 40]. The $B(t)$ value is adjusted in real time to reduce deviation $e(t)$ by changing the rotation angle of the servomotor. Therefore, the robot fish continually adjusts its own course and swims steadily at the target yaw angle.

The third closed loop is pectoral fin control (Fig. 4), keeping the robot fish from pitching up and down while it is cruising. The input signal is the target pitch angle (γ), and the feedback signal is the real-time pitch angle ($\gamma(t)$). Through a comparison of these two signals, the deviation signal ($c(t)$) is calculated as:

$$c(t) = \gamma - \gamma(t). \tag{14}$$

Then, the deviation signal is transmitted to PID controller II, which is similar to PID controller I, and the output signal (ψ) is the rotation angle of the servomotor that controls the pectoral fin:

$$\psi = K_{p'} \times c(t) + K_{i'} \times \int_0^t c(\tau)\mathrm{d}\tau + K_{d'} \times \frac{\mathrm{d}c(t)}{\mathrm{d}t}, \tag{15}$$

where $K_{p'}$, $K_{i'}$, and $K_{d'}$ are the proportional, integral and differential coefficients, respectively, of the PID controller II. After testing, we set $K_{p'}$, $K_{i'}$, and $K_{d'}$ to be 1, 0, 0, respectively, which is suitable for our robot fish. This closed-loop structure functions when the target pitch angle is set as 0° until the ascent or descent command is provided. The robot fish body can be stable at the same level through the pectoral fin rotating opposite to the pitch angle of the robot.

It shall be mentioned that while the presented control strategy is derived for the robot fish that has a wire-driven active body and a complaint tail, it is also applicable to other robot fish, such as a multi-joint robot fish. In this case, the kinetic model will be different while the sensing can be the same. Because of the change of kinetics, the CPG connected to the controller I will be different.

4 Experiments

Several experiments are conducted to investigate the closed-loop CPG-based control design on an experimental platform. For satisfying various experimental requirements, two pools are used: one is a 3000 mm (L) × 2000 mm (W) × 660 mm (H) rectangular pool and the other is a circular pool with a diameter and height of 3050 mm and 760 mm, respectively. A camera is installed above the pool to shoot the robot fish. The communication between PC and the robot occurs through the RF wireless module, and the user interface is programmed using LabVIEW.

4.1 Obstacle avoidance

Obstacle avoidance experiment is conducted using

the aforementioned closed-loop CPG control. The experimental parameters are as follows. The CPG model parameters M, ω, B, and R are 20, 3π, 0, and 1, respectively, when the robot cruises straight, whereas the parameters M, ω, B, and R are 30, 4π, ± 30, and 1.5, respectively, when the robot makes a turn, as depicted in Table 1. The voltage thresholds of the front, left, and right IR sensors are 1 V, 1.5 V, and 1.5 V, respectively, as discussed in section **3**.

The experiment includes two parts: simulation and physics. The IR sensor sampling frequency is 100 Hz, which is adequate for updating obstacle information. However, it also means that the obstacle can be detected when the active body flaps to any position within a flapping period. Then, the robot fish starts to react according to the changed CPG parameters. The response varies depending on where the active body bends to. For instance, the front IR sensor detects an obstacle when the body bends to the rightmost position, leftmost position, or somewhere between them, then the robot fish starts turning left from these states, resulting in the imparities of initial responses. Fig. 5 depicts the simulation result of the initial responses at three special states. At first, the robot fish cruises forward, and the flapping of the active body is symmetrical. An obstacle signal of the front IR sensor is received when the active body bends to the midline position, rightmost position, and leftmost position, then the CPG parameters M, ω, B, and R change from 20, 3π, 0, and 1 to 30, 4π, −30, and 1.5, respectively. The rightmost position, midline position, and leftmost position are depicted in Fig. 2. At the initial response, the active body flaps following the trend of the previous moment and then moves to the next pattern gradually instead of switching to the next pattern immediately. These different responses help the robot fish avoid obstacles more effectively.

Fig. 6 presents images of an obstacle avoidance experiment. The white rubber plug on the top of the robot fish is used as a marker to depict the swimming track. Initially, the robot fish cruises from the top left corner of the pool, with several obstacles placed in the pool. At 1.51 s, the robot fish swims close to the blue obstacle on its left until the left IR sensor detects the obstacle. Then, the robot fish changes the CPG parameters to turn slightly to the right until it swims away from the obstacle. After a very short time, the robot fish detects the front wall, and starts to turn left. The fourth figure illustrates that there is none of collision between the robot fish and wall at 3 s. From 1.51 s to 2.21 s, the robot fish successfully detects and avoids obstacles twice within a second, which certifies that the robot fish responds fast enough. After avoiding the wall, the robot fish continues to cruise forward. At 7.18 s, it again detects a white obstacle in front, and begins to turn left. The sixth figure shows even more clearly that there is still some space between the robot and the obstacle. Similarly, the robot detects the green obstacle at 9.15 s, and successfully avoids it at 10.31 s. At 12.31 s, the robot fish completes one lap. In this experiment, the closed-loop control of static obstacle avoidance is verified to be effective.

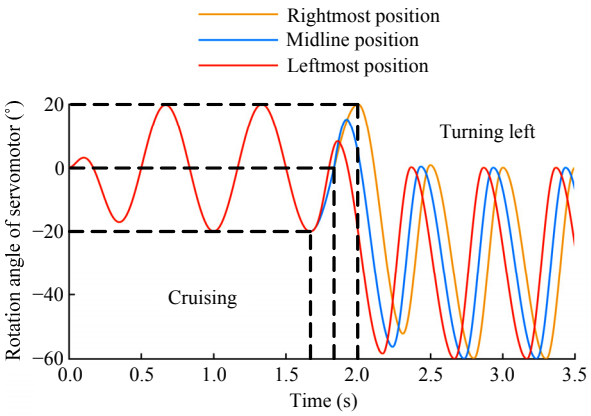

Fig. 5 The initial response of the robot fish when the obstacle is detected by the front IR sensor at different flapping states: midline position, rightmost position, and leftmost position.

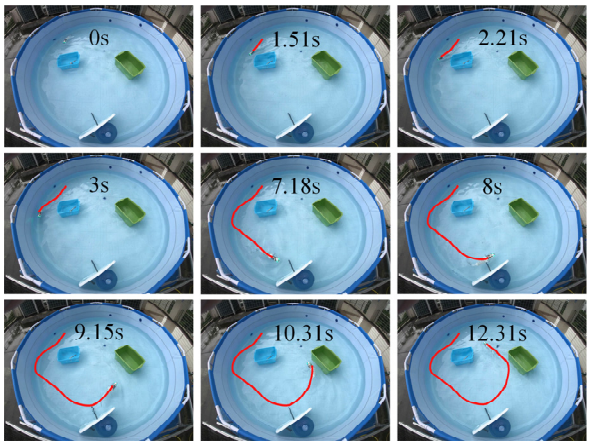

Fig. 6 Images of the obstacle avoidance experiment.

4.2 Moving obstacle avoidance

In nature, fish often encounter moving obstacles such as other swimming fish. However, they usually do not bump into each other. Our robot fish can do the same. In this experiment, two wire-driven robot fish are placed in a circular pool, as depicted in Fig. 7. The marker for painting the swimming track is also the white rubber plug. The red line shows the track of fish I, while the blue line displays fish II. At first, fish I cruises towards the upper right of the pool, whereas fish II cruises to the right of the pool, and they keep getting closer. At 1.45 s, according to the feedback signal of the front IR sensor, fish I detects that fish II is passing in front, and then turns left to avoid collision with it. Almost at the same time, fish II detects the pool wall in front and starts to turn left. The third figure shows that, at 2.14 s, both of them are in the state of turning. After swimming for a while, two fish get closer again. This time fish I cruises towards the upper left of the pool, whereas fish II cruises to the left of the pool. Fish I detects fish II and starts to turn left at 5.74 s. At 6.50 s, fish I and fish II are parallel to each other. Although the distance between them is minimized, there is still no collision here. The right IR sensor of fish I detects the fish II, while the left IR sensor of fish II detects the fish I. Then, fish I turns left; rather, fish II turns right. They keep away from each other and continue to swim freely in the pool without any collision. In this experiment, the obstacle is not only the static pool wall, but also the swimming fish. This increases the difficulty of avoiding obstacles, but the result shows that the robot fish can successfully avoid the moving obstacle, which further verifies the closed-loop CPG-based control of obstacle avoidance to be effective.

4.3 Direction tracking

In this experiment, the robot fish is controlled to track a designated direction. The direction is determined by two directional parameters: the yaw and the pitch angles. The former is achieved through controlling the flapping of the robot fish while the latter is achieved through controlling the pectoral fin. According to the analysis of the closed-loop control in section **3**, the robot fish can stabilize at the target yaw angle by reducing the deviation between the yaw angle and the target yaw angle. We set three groups of experiments, including three target yaw angles, namely 30˚, 60˚, and 90˚, and each group experiment is repeated three times. Before the robot fish starts swimming, some parameters are set to be constant as follows. Three of the CPG parameters M, ω, and R are set to be 20, 2π, and 1, respectively. The coefficients of the PID controller I and II, K_p, K_i, and K_d, are set as 0.7, 0, and 0.05, respectively, and $K_{p'}$, $K_{i'}$, and $K_{d'}$ are set to be 1, 0, and 0, respectively, based on the analysis in section **3**.

Fig. 8 shows the direction tracking experiment at three different target yaw angles. The swimming tracks are depicted by using the white rubber plug as a marker. Initially, the robot fish starts at the corner of the rectangular pool, parallel to the pool wall. After setting the target yaw angle, it begins to cruise and adjust its swimming orientation toward the target direction. A comparison of the track and the reference angle at the bottom right corner of the figure reveals that the swimming direction of the robot is close to the reference angle when it reaches a stable swimming state. The real-time yaw angle of the robot fish and the rotation angle of the servomotor driving the active body, which are sent back from the robot fish, are depicted in Figs. 9 and 10, respectively. The blue, red, and yellow lines, whose

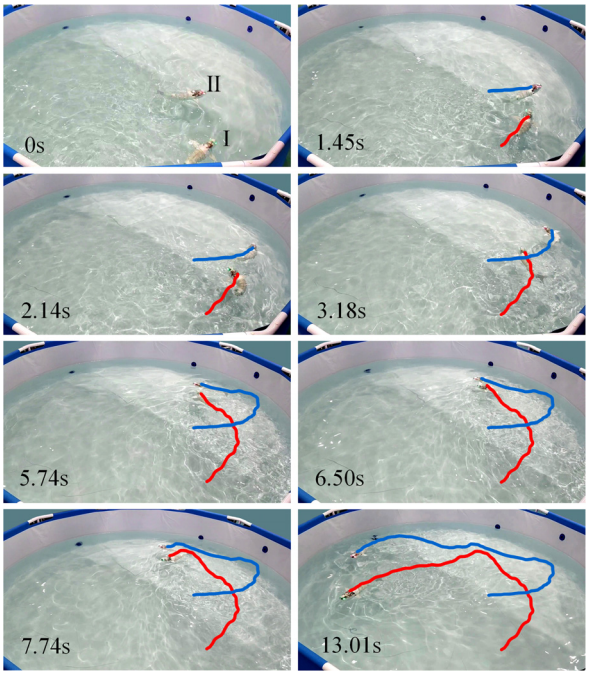

Fig. 7 Moving obstacle avoidance experiment. The robot in front is the obstacle, and the two robot fishes are obstacles to each other based on the relative position.

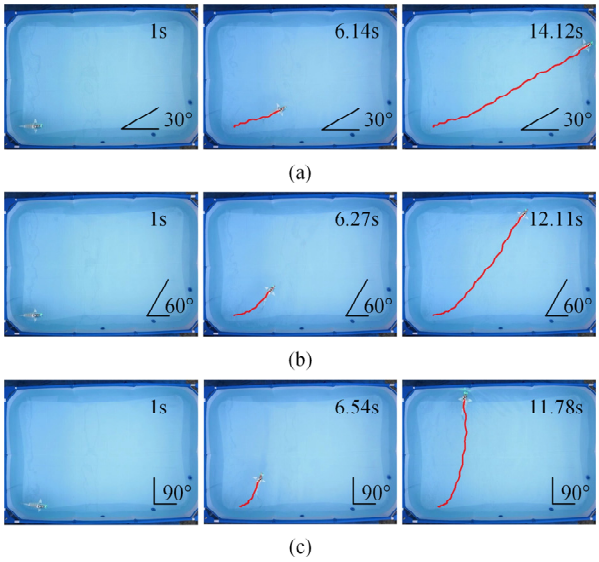

Fig. 8 Direction tracking at different target yaw angles. (a) 30˚; (b) 60˚; (c) 90˚.

initial angles are 0˚, are the target yaw angles of 30˚, 60˚, and 90˚, respectively. The following observations can be made from these two figures:

(1) In Fig. 9, the blue line first reaches 30˚ at 2 s approximately. However, by comparing the blue line in Fig. 10, the rotation angle of the servomotor is still negative at 2 s, which indicates that the robot is turning left. At this moment, the robot head reaches the target yaw angle, whereas the active body is flapping on the left side of the midline position. Then, the absolute value of $B(t)$ gradually decreases, causing the yaw angle to deviate slightly by 30˚. The robot fish adjusts its direction toward 30˚ again. The aforementioned process is repeated at several periods until the yaw angle stabilizes at 30˚. Therefore, some deviations occur from 30˚ in the first few periods after it first reaches 30˚, but the robot can adjust back and stabilize at 30˚ when $B(t)$ is stable at approximately 0˚. Similarly, the red and yellow lines first reach 60˚ and 90˚ at approximately 4.5 s and 6 s, respectively (Fig. 9), and some deviations occur in the first few periods. Then, both the lines gradually stabilize at the target yaw angle.

(2) The trend of three lines in Fig. 10 reveals that the absolute value of $B(t)$ reaches its maximum at first and then decreases gradually. The trend corresponds to the deviation signal of the closed-loop CPG control, which verifies the feasibility of the closed loop. Moreover, the offset values of 60˚ and 90˚ are not considerably different in the first period because of the range of $B(t)$, as expressed in Eq. (13). However, the robot can also achieve a greater target yaw angle by reducing the change rate of $B(t)$.

(3) The further the initial yaw angle is from the target yaw angle, the longer the time to reach a stable swimming state is. As detailed in Fig. 10, the offset values $B(t)$ of three target yaw angles start to stabilize at 0˚ at approximately 4 s, 5.5 s, and 7.1 s.

The yaw angle of the robot fish swings up and down in sine form at the target yaw angle due to the swing of the head. In each experiment, we select three relatively stable periods to obtain the mean values of yaw angles and subtract the target yaw angles to obtain

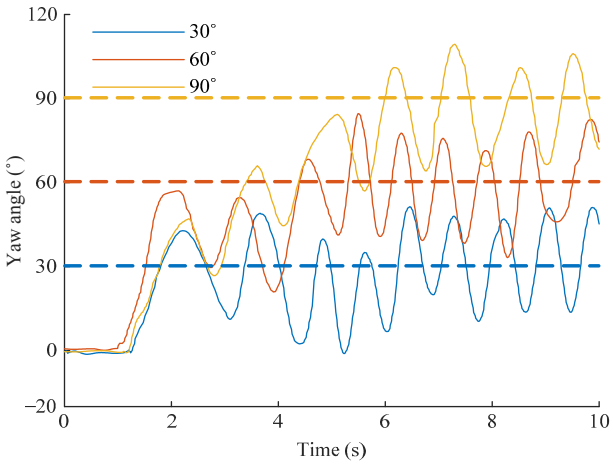

Fig. 9 The real-time yaw angle of the robot fish at different target yaw angles.

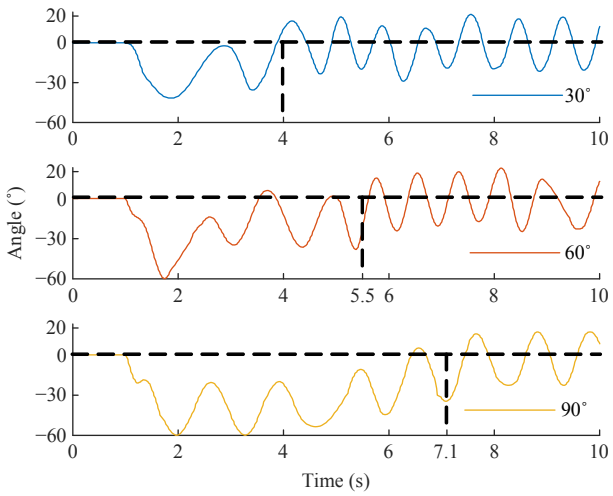

Fig. 10 The rotation angle of the servomotor (α) at different target yaw angles.

the errors, as shown in Table 2. The data show that the errors are small, with the average errors of −1.275°, 1.939°, and 1.930° for three target yaw angels of 30°, 60°, and 90°, respectively. The corresponding standard deviations are 2.728°, 2.699°, and 0.668°, respectively. The errors are acceptable, because the experiment process is dynamically adjusted. The average errors are less than 2°, which demonstrates the feasibility of our design.

The last closed-loop control is the pectoral fin control, corresponding to the pitch angle of the robot. In this experiment, the buoyancy of the robot fish is slightly larger than gravity, but the robot may dive without the pectoral fin control. We set the target pitch angle to be 0° throughout the direction tracking. Fig. 11 illustrates the pitch angle when the robot fish cruises at different target yaw angles from 6 s to 10 s, at which time the robot reaches a relatively stable state. The robot adjusts the rotation of its pectoral fin automatically to keep the robot fish body near the level. All of the pitch angles fluctuate around 0°, and the range of the fluctuation is within ± 4°. This shows that the robot fish is relatively stable during swimming.

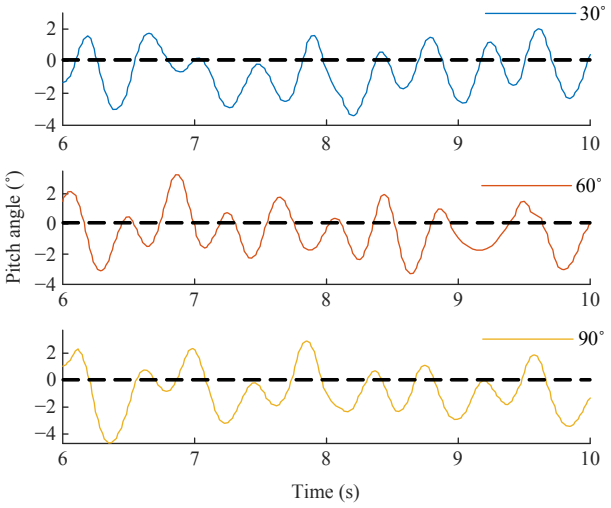

Fig. 11 The pitch angle of the robot fish at different target yaw angles.

Table 2 Error analysis of three stable periods

Angle (°)	Error (°)	Average error (°)	Standard deviation (°)
30	−0.886 1.237 −4.177	−1.275	2.728
60	−1.080 2.778 4.119	1.939	2.699
90	1.239 2.573 1.976	1.930	0.668

4.4 Direction tracking with obstacle avoidance

To enrich the closed-loop CPG control, an obstacle avoidance experiment under a fixed cruising orientation is conducted by integrating the aforementioned closed-loop control. The target yaw angle is set as 0°, and other parameters are the same as the obstacle avoidance experiment. A red obstacle is placed in the robot fish's path to test if the robot could avoid it and adjust to the initial orientation. Fig. 12 depicts the experimental result. The reference angle 0° is parallel to the rectangular pool wall. First, the robot fish starts cruising from the right side of the pool at a yaw angle of 0°. At 3.07 s, the front IR sensor detects the obstacle and the robot fish starts turning left. At this time, obstacle avoidance has a higher priority than direction tracking. After about 2 s, the robot successfully avoids the obstacle. At 5.27 s, the robot is almost perpendicular to the original direction and begins to conduct direction tracking. It takes about two seconds to reach the original direction based on the feedback signal of IMU. Similar to the direction tracking experiment, some deviations occur in the first few flapping periods. At 10.98 s, the robot fish basically returns to its original direction. Dotted lines I and II denote the cruising directions of the robot fish during several stable periods before and after encountering the obstacle. These two lines are nearly parallel, indicating that the robot fish performs both obstacle avoidance and direction tracking well.

Table 3 shows the obstacle avoidance and direction tracking of our robot fish in comparison with other closed-loop CPG-based robot fishes in the literatures. First, our robot fish is the only one with a wire-driven design, and only needs one driving servomotor for cruising. It is easy to control and low-cost. Second, few robot fishes adopted closed-loop CPG-based control in researching obstacle avoidance and direction tracking, even no other robot fish conducted moving obstacle avoidance. Third, in the reported angular error of

Table 3 Comparison of closed-loop CPG-based robot fishes in the literatures

Prototype	Structure design	Number of driving servomotors	Static obstacle avoidance	Moving obstacle avoidance	Direction tracking	Angular error of direction tracking
Our robot fish	Wire-driven	1	Yes	Yes	Yes	Less than 2°
Robotic fish (*i*-RoF)[23,32]	Multi-joint	2	Yes	Not reported	Yes	Not reported
Robotic fish[22,33,34]	Multi-joint	4	Yes	Not reported	Yes	Less than 3°
Boxfish-like robot[24]	Single joint	1	Not reported	Not reported	Yes	About 5°
Agile robotic fish[25]	Two caudal fins	2	Yes	Not reported	Not reported	Not reported
Robotic shark[35]	Multi-joint	2	Yes	Not reported	Not reported	Not reported

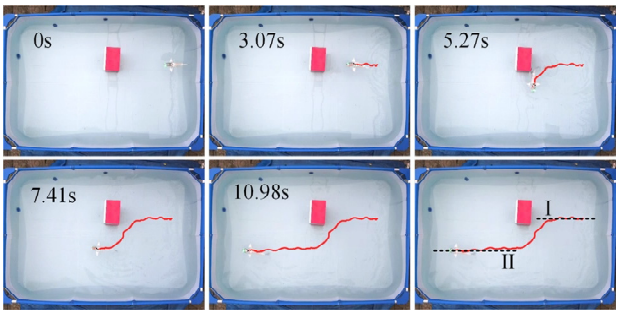

Fig. 12 Obstacle avoidance under a target yaw angle of 0°.

direction tracking, our robot fish has the minimum error, less than 2°.

5 Conclusions and future work

This paper presents a bioinspired closed-loop CPG-based control of a robot fish for obstacle avoidance and direction tracking. Based on the discussions above, the following conclusions can be drawn. The control strategy can effectively control the robot fish to avoid obstacles and track designated directions with IR sensors and IMU through proper calibration. The experiments demonstrated that the robot fish can avoid static and moving obstacle, and track designated direction even with an obstacle in the path. The average tracking error is less than 2°. Comparing to the existing robot fishes, our robot fish has two advantages: First, it can detect and avoid not only static obstacles but also moving obstacles. Second, it has less direction tracking errors. Additionally, the presented control strategy is developed for our robot fish. Though, it can also be used for other robot fish, such as the multi-joint robot fish, with some modifications in the CPG control part. In the future, several issues shall be further investigated such as target tracking or three-dimensional obstacle avoidance. In particular, we plan to add a GPS navigation system so that the robot fish can swimming across rivers.

Acknowledgment

The authors would like to thank Mr. Yudong Chen, and Mr. Binghuan Yu for their help in building the robot fish and conducting experiments. Research was supported by the Strategic Priority Research Program of the Chinese Academy of Sciences (class A) (Grant No. XDA22040203), the Fundamental Research Funds for the Central Universities (Grant No. 2019XX01), GDNRC[2020]031, and the Natural Science Foundation of Guangdong Province (Grant No. 2020A1515010621).

References

[1] Triantafyllou M S, Triantafyllou G S. An efficient swimming machine. *Scientific American*, 1995, **272**, 64–70.

[2] Yu J Z, Hu Y H, Fan R F, Wang L, Huo J Y. Mechanical design and motion control of a biomimetic robot dolphin. *Advanced Robotics*, 2007, **21**, 499–513.

[3] Yu J Z, Ding R, Yang Q H, Tan M, Wang W B, Zhang J W. On a bio-inspired amphibious robot capable of multimodal motion. *IEEE/ASME Transactions on Mechatronics*, 2012, **17**, 847–856.

[4] Wen L, Wang T M, Wu G H, Liang J H. Quantitative thrust efficiency of a self-propulsive robot fish: Experimental method and hydrodynamic investigation. *IEEE/ASME Transactions on Mechatronics*, 2013, **18**, 1027–1038.

[5] Katzschmann R K, DelPreto J, MacCurdy R, Rus D. Exploration of underwater life with an acoustically controlled soft robot fish. *Science Robotics*, 2018, **3**, eaar3449.

[6] Sfakiotakis M, Lane D M, Davies J B C. Review of fish swimming modes for aquatic locomotion. *IEEE Journal of Oceanic Engineering*, 1999, **24**, 237–252.

[7] Yu J Z, Wang M, Dong H F, Zhang Y L, Wu Z X. Motion control and motion coordination of bionic robot fish: A review. *Journal of Bionic Engineering*, 2018, **15**, 579–598.

[8] Du R, Li Z, Youcef-Toumi K, y Alvarado P V. *Robot Fish: Bio-Inspired Fishlike Underwater Robots*, Springer, New York, USA, 2015.

[9] Hu Y H, Zhao W, Wang L. Vision-based target tracking and collision avoidance for two autonomous robot fish. *IEEE Transactions on Industrial Electronics*, 2009, **56**, 1401–1410.

[10] Wang W, Xie G M. Online high-precision probabilistic localization of robot fish using visual and inertial cues. *IEEE Transactions on Industrial Electronics*, 2015, **62**, 1113–1124.

[11] Yu J Z, Su Z S, Wang M, Tan M, Zhang J W. Control of yaw and pitch maneuvers of a multilink dolphin robot. *IEEE Transactions on Robotics*, 2012, **28**, 318–329.

[12] Liu J D, Hu H S. Biological inspiration: From carangiform fish to multi-joint robot fish. *Journal of Bionic Engineering*, 2010, **7**, 35–48.

[13] Chen Z, Shatara S, Tan X B. Modeling of biomimetic robot fish propelled by an Ionic polymer-metal composite caudal fin. *IEEE/ASME Transactions on Mechatronics*, 2010, **15**, 448–459.

[14] Rossi C, Colorado J, Coral W, Barrientos A. Bending continuous structures with SMAs: A novel robot fish design. *Bioinspiration & Biomimetics*, 2011, **6**, 045005.

[15] Li Z, Du R X. Design and analysis of a biomimetic wire-driven flapping propeller. 4*th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*(*BioRob*), Rome, Italy, 2012, 276–281.

[16] Shin D J, Na S Y, Kim J Y, Baek S J. Fuzzy neural networks for obstacle pattern recognition and collision avoidance of fish robots. *Soft Computing*, 2008, **12**, 715–720.

[17] Shin D J, Na SY, Kim J Y, Baek S J. Fuzzy neural networks for obstacle pattern recognition and collision avoidance of fish robots. *Soft Computing*, 2008, **12**, 715–720.

[18] Deng X, Jiang D L, Wang J, Li M X, Chen Q S. Study on the 3D printed robot fish with autonomous obstacle avoidance behavior based on the adaptive neuro-fuzzy control. *IECON* 41*st Annual Conference of the IEEE Industrial Electronics Society*, Yokohama, Japan, 2015, 000007–000012.

[19] Verma S, Shen D, Xu J. Motion control of robot fish under dynamic environmental conditions using adaptive control approach. *IEEE Journal of Oceanic Engineering*, 2018, **43**, 381–390.

[20] Ijspeert A J. Central pattern generators for locomotion control in animals and robots: A review. *Neural Networks*, 2008, **21**, 642–653.

[21] Marder E, Bucher D. Central pattern generators and the control of rhythmic movements. *Current Biology*, 2001, **11**, R986–R996.

[22] Yu J Z, Wang K, Tan M, Zhang J W. Design and control of an embedded vision guided robot fish with multiple control surfaces. *Scientific World Journal*, 2014, **2014**, 631296.

[23] Korkmaz D, Koca G O, Li G Y, Bal C, Ay M, Akpolat Z H. Locomotion control of a biomimetic robot fish based on closed loop sensory feedback CPG model. *Journal of Marine Engineering & Technology*, 2019, 1638703.

[24] Wang W, Xie G M. CPG-based locomotion controller design for a boxfish-like robot. *International Journal of Advanced Robotic Systems*, 2014, **11**, 87–97.

[25] Zhang S W, Qian Y, Liao P, Qin F H, Yang J M. Design and control of an agile robotic fish with integrative biomimetic mechanisms. *IEEE/ASME Transactions on Mechatronics*, 2016, **21**, 1846–1857.

[26] Zhong Y, Li Z, Du R X. A novel robot fish with wire-driven active body and compliant tail. *IEEE/ASME Transactions on Mechatronics*, 2017, **22**, 1633–1643.

[27] Zhong Y, Song J L, Yu H Y, Du R X. Toward a transform method from lighthill fish swimming model to biomimetic robot fish. *IEEE Robots and Automation Letters*, 2018, **3**, 2632–2639.

[28] Xie F R, Zhong Y, Du R X, Li Z. Central pattern generator (CPG) control of a biomimetic robot fish for multimodal swimming. *Journal of Bionic Engineering*, 2019, **16**, 222–234.

[29] Xie F R, Li Z, Ding Y, Zhong Y, Du R X. An experimental study on the fish body flapping patterns by using a biomimetic robot fish. *IEEE Robotics and Automation Letters*, 2020, **5**, 64–71.

[30] Ijspeert A J, Crespi A, Ryczko D, Cabelguen J M. From swimming to walking with a salamander robot driven by a spinal cord model. *Science*, 2007, **315**, 1416–1420.

[31] Ijspeert A J, Crespi A. Online trajectory generation in an amphibious snake robot using a lamprey-like central pattern generator model. *Proceedings IEEE International Conference on Robotics and Automation*, Roma, Italy, 2007, 262–268.

[32] Bal C, Koca G O, Korkmaz D, Akpolat Z H, Ay M. CPG-based autonomous swimming control for multi-tasks of a biomimetic robotic fish. *Ocean Engineering*, 2019, **189**, 106334.

[33] Wang M, Yu J Z, Tan M. CPG-based sensory feedback control for bio-inspired multimodal swimming. *International Journal of Advanced Robotic Systems*, 2014, **11**, 170–180.

[34] Wu Z X, Yu J Z, Tan M, Zhang J W. Kinematic comparison of forward and backward swimming and maneuvering in a self-propelled sub-carangiform robotic fish. *Journal of Bionic Engineering*, 2014, **11**, 199–212.

[35] Yu J Z, Chen S F, Wu Z X, Wang W B. On a miniature free-swimming robotic fish with multiple sensors. *International Journal of Advanced Robotic Systems*, 2016, **13**, 62887.