Proceeding of the IEEE
International Conference on Robotics and Biomimetics
Dali, China, December 2019

Neural-Dynamics-Based Path Planning of a Bionic Robotic Fish*

Zhiwei Yu, Jielian Tao, Jianyu Xiong and Ao Luo
Institute of Bio-inspired Structure and Surface Engineering
Nanjing University of Aeronautics and Astronautics
Nanjing, P. R. China
yuzhiwei@nuaa.edu.cn

Simon X. Yang
The Advanced Robotics & Intelligent Systems
University of Guelph
Guelph, Ontario, Canada
syang@uoguelph.ca

***Abstract* - Without the ability of reversing, the bionic robotic fish has certain peculiarity compared with other robots, and it is more difficult to complete the specified tasks. Therefore, in this paper, a neural-dynamics-based path planning method applied to a bionic robotic fish is discussed. In this method, the target, obstacles and map are regarded as the pixel points under the coverage of the given area. Avoid the overlap of the pixel points of the target and the obstacles to realize the obstacle avoidance. The method is computationally intensive and adaptable, and the theoretical optimal path can be planned for various motion modes.**

Index Terms* - *robotic fish, neural networks, path planning, pixel points.

I. INTRODUCTION

After millions of years of selecting by nature, fish have abilities to go far beyond marine technology in many ways. As thus, more and more researchers have turned their attention to bionic robotic fish [1]. Bionic robotic fish is a new type of autonomous underwater vehicles (AUVs) with the advantages of fast movement speed, high movement efficiency and low energy consumption. In recent years, domestic and foreign research on bionic robotic fish has entered a boom, and has significant results in many areas.

Marchese et al. [2,3] has developed a flexible soft robotic fish “SoFi,” which simulates the escape ability of fish, using hydrostatic pumps for fast, continuous steering during autonomous motion. Frank Bonnet et al. [4] developed an underwater interactive robotic fish system. This system confuses and interacts with zebrafish by simulating the movement of their caudal fins. During the research, they found that this system can influence and control the swimming direction of the fish, which can be benefit for the study of animal behavior [5]. This research is of great significance in establishing the connection between biology and robotics. Using this interactive system, Polverino et al. [6] verified the ability of robots to assist freshwater fish in controlling mosquito populations, providing a new idea for solving the problem of invasive alien species. The Germany company Festo designed two classic bionic fishes “Aqua_ray” [7] and “Airacuda” [8]. Not only resembling fish in appearance, but also in sports patterns, they show the wonderful physiological structure and movement posture of fish. Korean company AIRO designed an ornamental bionic robotic fish "MIRO" [9]. There are two joints designed in the tail of the "MIRO", which can move flexibly and the motion pattern of the robot is lifelike. Several obstacle sensors are mounted on the robotic fish, through AI learning, they can prevent collision with other fishes. Wang et al. [10] are dedicated to research on robotic-fish-attitude. They developed a boxfish-like robot, using a bio-inspired controller to help the robotic fish brake, and getting strong maneuverability. At the same time, the research team used visual and inertial cues to locate the robotic fish [11]. Yu et al. [12] developed a robotic fish which can coordinate with each other. The application of the robotic fish in international competitions has greatly promoted research and education in this area. Aubin et al. [13] installed a system of biomimetic hydraulic devices in a robotic fish to simulate blood circulation, allowing the robotic fish to achieve higher energy density, providing a reference for robots to achieve efficiency and autonomy. Researchers from Imperial College London [14] have developed a flying robotic fish designed to collect water samples, which can leap out of water and fly back to base. Once airborne, it can travel for up to 26 m, by which time it would have collected water samples to determine the extent of flooding in an area or for monitoring ocean pollution. Due to the efficiency of the fish propulsion system, Zhu et al. [15] developed a tuna-like robotic fish to explore the propulsion principle of fish, which would benefit the development of underwater vehicles.

Based on the motion patterns of fish, the path planning methods are also diverse.

For example, the heuristic algorithm-based method commonly used in path planning of underwater robots: A* algorithm. This global algorithm uses grid maps to generate the optimal path, but the huge amount of data could make it into an infinite loop [16]. Sun et al. [17] applied fuzzy control algorithm to the path planning of robotic fish. This algorithm does not need accurate descriptions of the AUV mathematical model, but relies on expert experience. However, the dependence on expert experience causes the difficulty of establishing fuzzy rule base, which means that the algorithm is lacking adaptability to environment. Hu et al. [18] use a simulated annealing approach in the navigation of robotic fish to improve the convergence of the reinforcement learning algorithm in unknown environment, and obtain the optimal path.

* This work is partially supported by the National Natural Science Foundation of China (Grant No. 51475230). Foundation of National Defense Youth Science and Technology in NUAA (3082018NT2018022) supported by “the Central Universities”.

978-1-7281-6321-5/19//$31.00 ©2019 IEEE

In this paper, a neural-dynamics-based path planning method is applied to a robotic fish, which uses caudal fin just in the horizontal plane. For known maps, this approach does not require a large amount of pre-learning as the reinforcement learning algorithm, or relies on the expert experience as the fuzzy control algorithm, and has a strong adaptability to the complicated and changeable underwater environment. Its amount of calculation is basically linearly related to the size of the map. So, it would never run into an infinite loop as the A* algorithm. And in theory, the planned path should be optimal.

This article is divided into the following sections. In section II, a brief introduction to the model of the robotic fish, a path planning method based on neural dynamics, and its application to the robotic fish are proposed. In section III, after simplifying the fish into rectangles, the path planning simulations are carried out and the results are analyzed. Then in section IV, we give a brief introduction to the experiment platform of the robotic fish using this method.

II. MODEL

In this section, we give a brief introduction to the model of the robotic fish, a path planning method based on neural dynamics, and its application to the robotic fish.

A. Robotic Fish Model

In this paper, a robotic fish that moves only in the horizontal plane is discussed. It consists of two parts: the body with control system, and the caudal fin that provides propulsion. The robotic fish is based on zebrasoma flavescens. Its silicone outer skin is waterproof. As shown in Fig. 1, its shape is both realistic and cute, which can arouse the interest of people. In this paper, the shape of the robotic fish is considered in the path planning, instead of simplifying the robotic fish as a point. Therefore, although the robotic fish just moves in the horizontal plane, a three-dimensional coordinate (x, y, θ) is needed to describe it. Where, x, y represents the position of the center point *Base* of the robotic fish in the coordinate system, and θ represents the angle between the direction of the robotic fish and the X-axis. We simply define the next position of the robotic fish as three possible changes, which are leftward, straight forward and rightward. At the same time, the deflection angle γ of leftward or rightward is a specified value, such as the length of each step. The bird's overhead view of the model of robotic fish in 2-D map is shown in Fig. 2.


Fig. 1 The robotic fish.

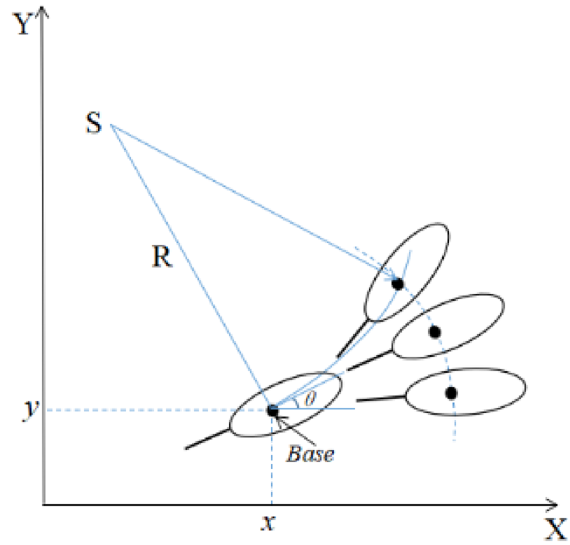


Fig. 2 Bird's overhead view of the model.

B. Proposal of Related Theory

A patch of membrane in a biological neural system, using electrical circuit elements, was proposed by Hodgkin and Huxley in 1952 [19]. The dynamics of the voltage across the membrane V_m in their membrane model is described using a state equation technique as

$$C_m \frac{dV_m}{dt} = -(U_p+V_m)G_p+(U_{Na}-V_m)G_{Na}-(U_K+V_m)G_K \qquad (1)$$

where C_m is the membrane capacitance. U_K, U_{Na} and U_p are the Nernst potentials (saturation potentials) for potassium ions, sodium ions and passive leak current in the membrane, and their corresponding conductances are G_K, G_{Na} and G_p.

Based on (1), Grossberg [20] proposed a shunting model to understand the real-time adaptive behavior of individuals to complex and dynamic environmental contingencies, by setting $C_m = 1$ and substituting $q_i = U_p+V_m$, $A = G_p$, $B = U_{Na}+U_p$, $D = U_K - U_p$, $S_i^e = G_{Na}$, and $S_i^i = G_K$ in (1). The model can be characterized by a shunting equation as

$$\frac{dq_i}{dt} = -Aq_i+(B-q_i)S_i^e(t)-(D+q_i)S_i^i(t) \qquad (2)$$

where q_i is the value of the ith neuron. A • 0, B • 0, and D • 0. Variables V_V^V and a_a^a are the excitatory and inhibitory inputs to the neuron.

Inspired by (2), Yang proposed a neural network model [21-23]. The target and its neighboring neurons determine the excitatory input e_e^e, and just the obstacles determine the inhibitory input i_i^i. Thus, the dynamics of the ith neuron in the neural network is characterized by a shunting equation as

$$\frac{dq_i}{dt} = -Aq_i + (B - q_i)\left([I_i]^+ + \sum_{j=1}^{N} \omega_{ij}[q_i]^+\right) - (D + q_i)[I_i]^- \qquad (3)$$

where N is the total number of the neurons. $[I_i]^+ + \sum_{j=1}^{N} \omega_{ij}[q_i]^+$ is the excitatory input. $[I_i]^-$ is the inhibitory input. And the input I_i can be defined as

$$I_i = \begin{cases} E, & \text{if there is a target} \\ -E, & \text{if there is an obstacle} \\ 0, & \text{otherwise} \end{cases} \qquad (4)$$

and we should notice that $E >> B$, $i = 1, 2, \ldots, N$. Function $[p]^+$ is a linear-above-threshold function defined as $[p]^+ = \max\{p, 0\}$, and the nonlinear function $[p]^-$ is defined as $[p]^- = \max\{-p, 0\}$. For ω_{ij},

$$\omega_{ij} = \begin{cases} \dfrac{\mu}{d_{ij}}, & \text{if Neurons } j \text{ and } i \text{ are neighbors} \\ 0, & \text{otherwise} \end{cases} \qquad (5)$$

where d_{ij} is the Euclidean distance between adjacent neurons, μ is a positive constant.

Yang define all the neurons having lateral connection to the ith neuron as its neighboring neurons. Owing to the symmetry of the lateral neural connection weights, i.e. $\omega_{ij} = \omega_{ji}$, the dynamics of the ith neuron can be further written as

$$\frac{dq_i}{dt} = -Aq_i + (B - q_i)\left([I_i]^+ + \sum_{j=1}^{k} \omega_{ij}[q_i]^+\right) - (D + q_i)[I_i]^- \qquad (6)$$

where k is the number of neighboring neurons of the ith neuron. According to the robotic fish model, $k = 3$ in this paper.

Figuratively, the method is to define the target as the peak, and the obstacles as the valley. Through the iteration of the neural activity, each point in the map has a value as its elevation. During the iteration, the target can affect the entire map, but the impact of obstacles can only be confined to themselves. Thus, the robotic fish can always climb to the peak along the optimal path, as long as it chooses the highest elevation of the optional points step by step.

III. Simulation Studies

The feasibility of the path planning method based on neural dynamics in a nonholonomic mobile robot has been demonstrated. In this section, we apply it to the introduced robotic fish model, changing the parameters and performing multiple simulations to obtain various results. For ease of display, the robotic fish is simplified to a rectangular. We define a point as a new obstacle when the robotic fish, that its *Base* is on this point, covers any piece of given obstacles or border. We realize obstacle avoidance by this way and achieve good results.

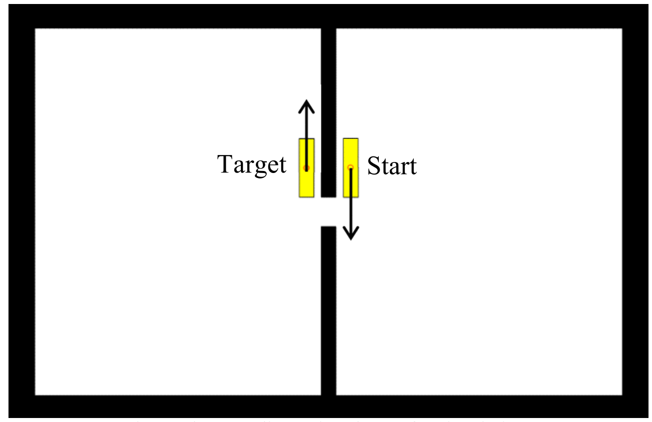


Fig. 3 The two-dimensional map for simulation.

A. *Introduction of Parameters and Map*

We select the following parameters: $E = 100$; $A = 10$; $B = 1$; $D = 1$; $\mu = 1$. In this paper, we just discuss the feasibility of this method on robotic fish, so a 60 × 90 simple two-dimensional map is designed, as shown in Fig. 3. The size of the fish is 2 × 8. In Fig. 3, the black part is the given border and obstacle; the left rectangle represents the target, which the robotic fish will eventually reach and its deflection angle with the X-axis is 90°; the right rectangle represents the starting position of the robotic fish, and the deflection angle with the X-axis is 270°. The map is open, so it can give the robotic fish enough rotation space to the subsequent comparative simulation experiments. The door for the robotic fish to pass through during the task is a tough condition for the path planning, which embodies the superiority of the method.

The neural network of the map in Fig. 4 consists of 90 × 60 × 24 neurons, which means the robotic fish can turn 15° at each step. The coordinates of the target are (40, 34, 6), and the coordinates of the starting point are (46, 34, 18) according to the given deflection angles of the robotic fish in Fig. 3. We selected four different step sizes r, and obtained the corresponding paths and the numbers of steps *Time*. The correspondence between r and *Time* is given in Table I, and the corresponding paths is shown in Fig. 4.

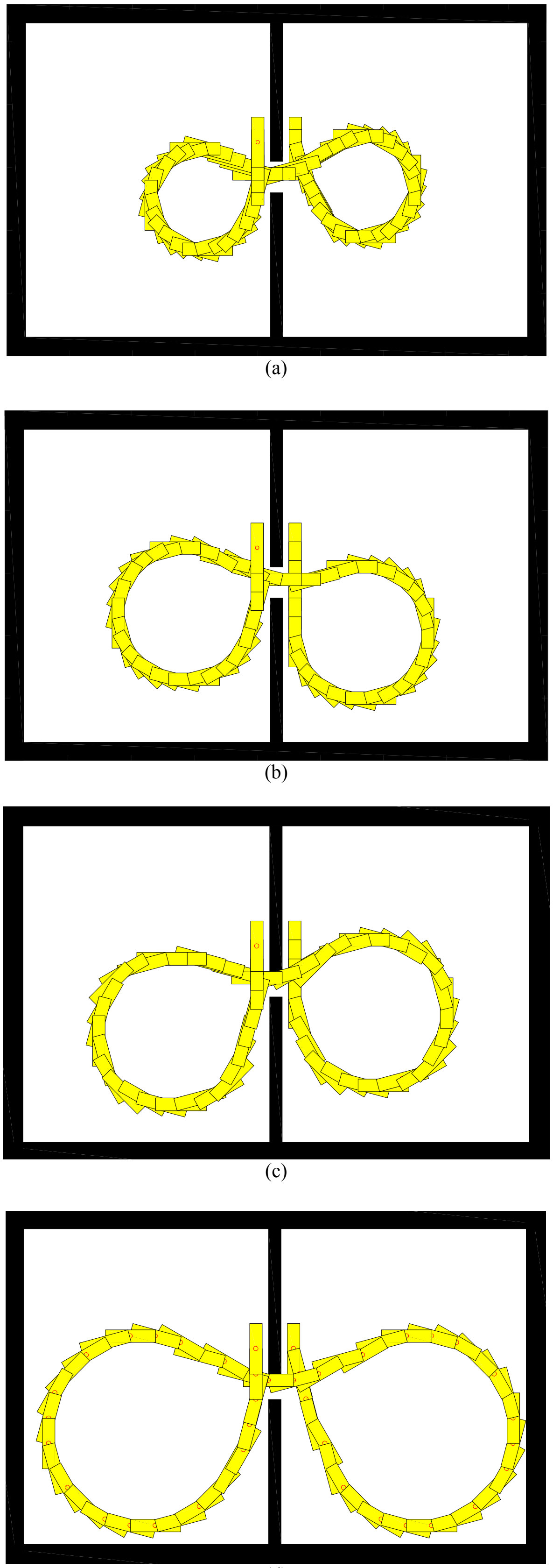
The neural network of the map in Fig. 5 consists of 90 × 60 × 12 neurons, which means the robotic fish can turn 30° at each step. The coordinates of the target are (40, 34, 3), and the coordinates of the starting point are (46, 34, 9) according to the given deflection angles of the robotic fish in Fig. 3. We selected four different step sizes r, and obtained the corresponding paths and the numbers of steps *Time*. The correspondence between r and *Time* is given in Table II, and the corresponding paths is shown in Fig. 5.

TABLE I
Selections of R and the Corresponding Numbers of Steps *Time*

r	2	2.61	3	4
Time	57	54	53	55

TABLE II
Selections of R and the Corresponding Numbers of Steps *Time*

r	1.1	3.5	6..5	7.1
Time	47	31	30	56



(a)

(b)

(c)

(d)

Fig. 4 The simulation track in MATLAB when the deflection angle is 15°. (a) The step sizes $r = 2$. (b) The step sizes $r = 2.61$. (c) The step sizes $r = 3$. (d) The step sizes $r = 4$.

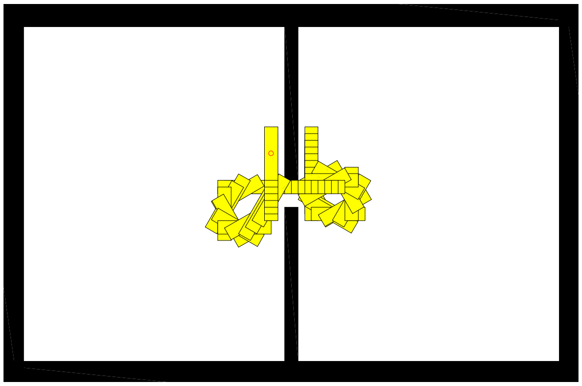
B. Discussion

In the map, as shown in Fig. 3, we set a quiet narrow door for the robotic fish, which is necessary to go through to complete the task, and set the start position and the target position of the robotic fish close to the obstacle wall, where the door is located. In this case, to achieve the target without the ability of reversing, the robotic fish must adjust its position by winding. Different step sizes and deflection angles will both affect the path of the robotic fish. Therefore, we carefully observed Fig. 4 and Fig. 5 and discussed to draw some conclusions.

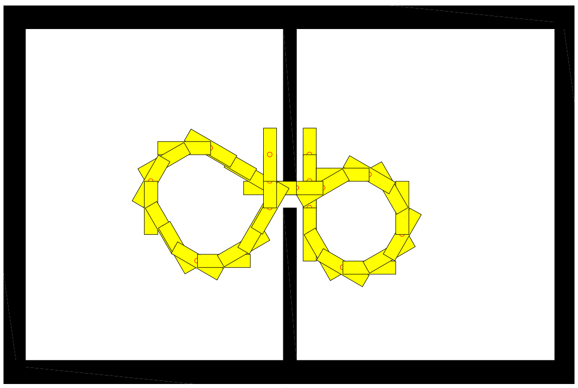
Fig. 4 shows the corresponding paths according to different r when $\gamma = 15°$. As the step size r increases, the turning radius R increases too. But the shape of the paths is similar that the basic figures are two circles. The four paths differ in the relationship between the two circles: in Fig. 4(a) and Fig. 4(c), the right circle is slightly higher than the left circle, which is on the contrary to the condition in Fig. 4(b), and the two circles in Fig. 4(d) are accordant. This is affected by the number of steps that the robotic fish goes straight forward at the beginning. At the same time, we found that the numbers of steps *Time* are similar, because the number of steps to make a circle is determined when γ is given. When r is small, the number of steps to reach the rotatable position, and to go through the door is large, so at the beginning, *Time* decreases as r increases (r = 2, 2.61, 3). When r continues to increase ($r = 4$), the robotic fish will not go straight forward at the beginning, but adjust its position during subsequent rotations. Cause its flexibility decreases, the *Time* becomes slightly larger. Fig. 5 shows the corresponding paths according to different r when $\gamma = 30°$. At the beginning, the path shape is similar to that in Fig. 4 when r increases (r = 1.1, 3.5, 6.5). The conclusions in Fig. 5(a)-(c) are similar to Fig. 4 as well. When r continues to increase ($r = 7.1$), as shown in Fig. 5(d), due to the size limitation of the map, the robotic fish needs to increase the number of turning circles, taking the opportunity to adjust its position, which means the step size increases but the efficiency reduces. Throughout Fig. 4 and Fig. 5, we can achieve the following two conclusions.

1) The smaller the γ, the more natural and beautiful the rotation. Since γ becomes larger, the circles in Fig. 5 do not have the smoothness of the circles in Fig. 4. It can be considered that the steering dispersion of the robotic fish is enhanced, but the steering speed will be greatly accelerated. In the simulation, we found that $\sin\gamma$ would be too small with a small γ, then a large r would be required for rotation. So, it will be difficult to obtain the path. When the steering becomes too discrete, the flexibility is too poor and it is difficult to obtain a good path as well. Therefore, the selection of γ should be appropriate, and we recommend 15° and 30° as two good values.

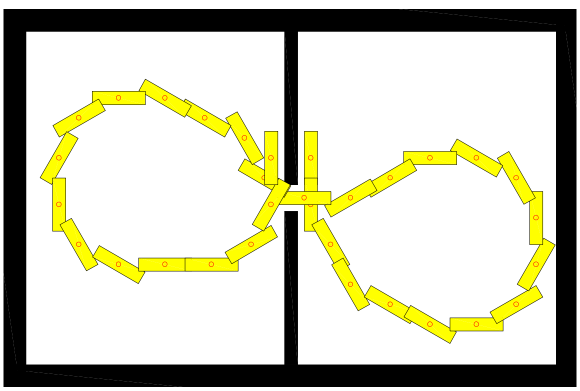
2) We define that the smaller *Time*, the higher the efficiency. Then the efficiency will increase first and then decrease as r increases, because when r is small, more steps are required when the robotic fish needs to go straight; when r increases to a certain degree, the flexibility of the robotic fish decreases. Therefore, the step size r can be selected according to actual needs, such as the map size, the complexity of the task, and the balance between time and energy consumption.



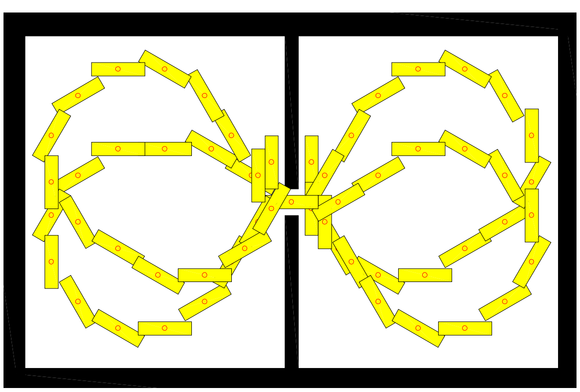
(a)



(b)



(c)



(d)

Fig. 5 The simulation track in MATLAB when the deflection angle is 30°. (a) The step sizes $r = 1.1$. (b) The step sizes $r = 3.5$. (c) The step sizes $r = 6.5$. (d) The step sizes $r = 7.1$.

IV. Experiment Platform

We built a physical experiment platform, which consists of two parts. The first part is a $2.4m \times 1.5m \times 0.6m$ pool, where the robotic fish can move freely. The planar shape of this pool is geometrically similar to the map in simulations, which is conducive to the verification of simulation experiments. The other part is the camera hanging above the pool and the computer for processing data and transmitting commands. The camera can transmit the position of the robotic fish to the computer in real time. Fig. 6 is the pool screen taken by the camera. Fig. 7 is the image of Fig. 6 after subjected to threshold processing.

In future experiments, we will first study the motion characteristics of the robotic fish and master the commands to control its forward distance and deflection angle. Then carry out the verification experiment, which translates the planned path into commands and transmit them to the robotic fish. The camera records the motion of the robotic fish. Then we subjected the images to threshold processing, and superimposed them to obtain the trajectory of the robotic fish. The obtained trajectory can be compared with the planned path, given by the simulation experiment.

Fig. 6 The scene that the robotic fish moves in the water.

Fig. 7 The scene that the robotic fish moves in the water after being subjected to threshold processing.

V. Conclusions

For a known map, the path planning method proposed in this paper can theoretically plan the optimal path of the robotic fish that moves in the horizontal plane by the caudal fin, and has been applied to the self-designed bionic robotic fish platform. This method requires neither extensive pre-learning nor expert experience, and is highly adaptable to complex and variable underwater environments. The amount of calculation is basically linearly related to the size of the map. Simulation studies show that even under the rigorous control condition, without reversing ability, the robotic fish can obtain a reasonable path by winding to reach the target position. The parameters r, γ involved in the method have a significant impact on the efficiency and shape of the path. So, they should be reasonably selected according to the demand. In the future research, we will discuss the twist of fish and how to the real-time changing of r in the path planning.

Acknowledgment

Thanks to Professor Zhendong Dai for providing a good laboratory space. And thanks all the reviewers for their suggestion.

References

[1] Scaradozzi D, Palmieri G, Costa D, et al. BCF swimming locomotion for autonomous underwater robots: a review and a novel solution to improve control and efficiency[J]. Ocean Engineering, 2017, 130: 437-453.

[2] Marchese A D, Onal C D, Rus D. Autonomous soft robotic fish capable of escape maneuvers using fluidic elastomer actuators[J]. Soft Robotics, 2014, 1(1): 75-87.

[3] Katzschmann R K, Marchese A D, Rus D. Hydraulic autonomous soft robotic fish for 3D swimming[C]//Experimental Robotics. Springer, Cham, 2016: 405-420.

[4] Bonnet F, Cazenille L, Séguret A, et al. Design of a modular robotic system that mimics small fish locomotion and body movements for ethological studies[J]. International Journal of Advanced Robotic Systems, 2017, 14(3): 1729881417706628.

[5] Bonnet F, Gribovskiy A, Halloy J, et al. Closed-loop interactions between a shoal of zebrafish and a group of robotic fish in a circular corridor[J]. Swarm Intelligence, 2018: 1-18.

[6] Giovanni Polverino, Mert Karakaya, Chiara Spinello, Vrishin R. Soman, Maurizio Porfiri. Behavioural and life-history responses of mosquitofish to biologically inspired and interactive robotic predators. Journal of The Royal Society Interface, 2019; 16 (158).

[7] W. Stoll, Aquaray: Water-hydraulic manta ray with flapping-wing drive, 2009.

[8] Festo. Airacuda, 2006. Available at: www.festo.com/cms/en_corp/9761.htm (accessed Dec. 31, 2013).

[9] https://www.sohu.com/a/202948419_309591

[10] Wang W, Xie G. CPG-based locomotion controller design for a boxfish-like robot[J]. International Journal of Advanced Robotic Systems, 2014, 11(6): 87.

[11] Wang W, Xie G. Online high-precision probabilistic localization of robotic fish using visual and inertial cues[J]. IEEE Transactions on Industrial Electronics, 2015, 62(2): 1113-1124.

[12] Yu J, Wang C, Xie G. Coordination of multiple robotic fish with applications to underwater robot competition[J]. IEEE Transactions on Industrial Electronics, 2016, 63(2): 1280-1288.

[13] Aubin C A, Choudhury S, Jerch R, et al. Electrolytic vascular systems for energy-dense robots[J]. Nature, 2019: 1.

[14] Zufferey R, Ancel A O, Farinha A, et al. Consecutive aquatic jump-gliding with water-reactive fuel[J]. Science Robotics, 2019, 4(34): 7330.

[15] Zhu J, White C, Wainwright D K, et al. Tuna robotics: A high-frequency experimental platform exploring the performance space of swimming fishes[J]. Science Robotics, 2019, 4(34): 4615.

[16] Wang H, Jiang Y L. Robotic Fish Path Planning Based on an Improved A* Algorithm[C]//Applied Mechanics and Materials. Trans Tech Publications, 2013, 336: 968-972.

[17] Sun B, Zhu D, Yang S X. An optimized fuzzy control algorithm for three-dimensional AUV path planning[J]. International Journal of Fuzzy Systems, 2018, 20(2): 597-610.

[18] Hu J, Mei J, Chen D, et al. Path Planning of Robotic Fish in Unknown Environment with Improved Reinforcement Learning Algorithm[C]//International Conference on Internet and Distributed Computing Systems. Springer, Cham, 2018: 248-257.

[19] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," J. Physiol. Lond., vol. 117, pp. 500–544, 1952.

[20] S. Grossberg, "Nonlinear neural networks: Principles, mechanisms, and architecture," *Neural Networks*, vol. 1, pp. 17–61, Sep. 1988.

[21] Yang X, Meng M. An efficient neural network model for path planning of car-like robots in dynamic environment[C]//Engineering Solutions for the Next Millennium. 1999 IEEE Canadian Conference on Electrical and Computer Engineering (Cat. No. 99TH8411). IEEE, 1999, 3: 1374-1379.

[22] Tu J, Yang S X. Genetic algorithm based path planning for a mobile robot[C]//2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422). IEEE, 2003, 1: 1221-1226.

[23] Yang S X, Meng M Q H. Real-time collision-free motion planning of a mobile robot using a neural dynamics-based approach[J]. IEEE Transactions on Neural Networks, 2003, 14(6): 1541-1552.