Proceedings of the 42nd IEEE
Conference on Decision and Control
Maui, Hawaii USA, December 2003

TuP08-5

Basic Motion Control of a Free-swimming Biomimetic Robot Fish

Junzhi Yu Shuo Wang Min Tan
Labatory of Complex Systems and intelligence science
Institute of Automation, Chinese Academy Sciences
P. O. Box 2728, Beijing 100080, China
E-mail: jzyu@compsys.ia.ac.cn

***Abstract*: We develop a practical motion control strategy for a radio-controlled, 4-link and free-swimming biomimetic robot fish that uses a flexible posterior body and an oscillating foil as propulsor. Because motion control of robot fish involves hydrodynamics of the fluid medium and dynamics of the robot, it is hard to establish a mathematic model employing purely analytical methods. The fish's motion control task, based on control performance of the fish, is decomposed into on-line speed control and orientation control. The speed control algorithm is then implemented by using step control, and orientation control is realized by fuzzy logic. Combining with step control and fuzzy control, a point-to-point control algorithm is implemented and applied to the closed-loop experimental system that using a vision-based position sensing subsystem to provide feedback. The running experiments confirm the reliability and effectiveness of the presented algorithms.**

I. INTRODUCTION

As is known to all, a fish in nature propels itself by the coordinate motion of its body, fins and tail, achieving tremendous propulsive efficiency or excellent maneuverability which win the advantage over conventional marine vehicles powered by rotary propellers with the same power consumption. Nature selection has ensured that the mechanical systems evolved in fish, although not necessarily, are very efficient and fitted for their living environments. In a sense of engineering, the fish is a distinguished AUV prototype. An overview of fish swimming and the analytical methods which have been applied to some of their propulsive mechanisms were given by Sfakiotakis et al [1] and Tong [2]. In recent years, research in propulsion and maneuvering mechanisms used by fish has demonstrated a variety of prospective utility in undersea vehicles [3-6]. In 1994, MIT successfully developed an 8-link, fish-like machine RoboTuna, which may be the first free-swimming robot fish in the world. RoboTuna and subsequent RoboPike projects attempted to create AUVs with increased energy savings and longer mission duration by utilizing a flexible posterior body and a flapping foil (tail fin) that exploits external fluid forces to produce thrust. In the mean time, another motivation was to answer Gray's paradox, which is that fishes don't seem to have enough power to propel themselves at the speed they do. Since then, based on progresses in robotics, hydrodynamics of fishlike swimming, new materials, actuators and control technology, more and more research has focused on the development of novel fish-like vehicles.

Here our goal is to design a radio-controlled, 4-link and free-swimming biomimetic robot fish that uses a flexible posterior body and an oscillating foil as propulsor, and to develop preliminary motion control strategy of robot fish systems that use visual feedback of robot position for path planning. The PTP (Point To Point) control, which means how to make robot fish move continuously and steadily from an initial point to a destination point, is one of basic problems concerning robot fish's controllability. Many complex motions of the fish in the future such as obstacle avoidance and formation control can be reduced to a series of PTP controls. So we focus on some basic motion control algorithms for the designed robot fish. Inspired by pet-style robots like AIBO [7], a remote controller that can be used for remote operation is also designed in the fish system.

Because robot fish's swimming involves hydrodynamics of the fluid environment and dynamics of the robot, precise mathematical model is difficult to establish by purely analytical methods. Considering that the speed of fish's swimming is adjusted by modulating joint's oscillating frequency, and its orientation is tuned by different joint's deflection, consequently, the overall motion control is decomposed into on-line speed control and orientation control. Recently there is an increasing tendency to build up Fuzzy Logic controllers (FLC) for uncertain control issues without precise mathematic model. Fuzzy logic was first introduced by Lofti A. Zadeh, a Professor at the University of California at Berkely in 1965. The mechanism of a FLC is that the uncertainty is represented by fuzzy sets and an action is generated cooperatively by several rules that are triggered to some degrees, and produce smooth and robust control outputs. The detailed FLC for robot fish's orientation is demonstrated in this paper. At present, without necessary positional sensor or telemetry, a CCD camera hanged over swimming pond acts as sensor that is in charge of capturing robot fish's moving and surrounding information. The position and orientation of fishes are then tracked employing a real-time tracking algorithm. Also the information of fish's position and orientation are input as basis of decision-making, hence a closed-loop swimming control is accomplished.

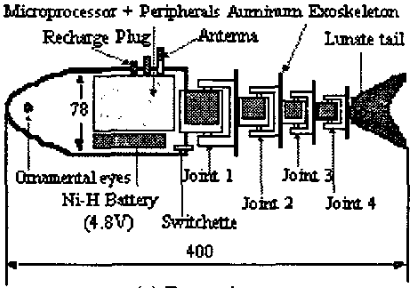
The paper is organized as follows: the prototype robot fish and its control performance are described in Section II. A speed control algorithm is presented in section III. Then a fuzzy logic controller for orientation control is designed in section IV. Based on the former, a detailed PTP control algorithm is given. Finally, corresponding experimental results are demonstrated and discussed.

II. REVIEW OF PROTOTYPE ROBOT FISH

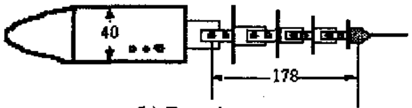
It is common in literature [1] that the passage of a wave underlies fish's propulsive structure, and the propulsive wave traverses the fish body in a direction opposite to the overall movement and at a speed greater than the overall swimming. A swimming model for RoboTuna was presented by Barrett et al [8], which consists of two basic components: the RoboTuna's body is represented by a planar spline curve and its lunate caudal tail by an oscillating foil. The spline curve starts from fish's center of inertia to the caudal joint, which is assumed to take the form of a traveling wave originally suggested by Lighthill.

$$y_{body}(x,t) = [(c_1 x + c_2 x^2)][\sin(kx + \omega t)] \qquad (1)$$

where y_{body} is transverse displacement of body, x is displacement along main axis, k is body wave number ($k = 2\pi / \lambda$), λ is body wave length, c_1 is linear wave amplitude envelope, c_2 is quadratic wave amplitude envelope, ω is body wave frequency ($\omega = 2\pi f = 2\pi / T$).

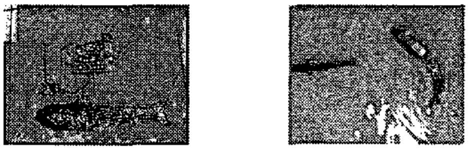


(a) Front view



(b) Fop view

Fig.1 Mechanical configuration of robot fish



(a) (b)

Fig.2 (a) Prototype robot fish and its remote controller
(b) Swimming robot fish vs. real fish (Carp)

Based on oscillation model of propulsive mechanism, a radio-controlled, 4-link and free-swimming biomimetic robot fish mimicking carangiform-like locomotion is designed. The mechanical configuration of a robot fish is shown in Fig.1 and the photo of robot fish in Fig.2. The robot fish primarily consists of five parts: control unit (microprocessor + peripherals), communication unit (wireless receiver), support (aluminum exoskeleton + head + forebody), actuation unit (DC servomotors) and accessories (battery, waterproofed skin, tail fin). For the robot fish, its speed is adjusted by modulating joint's oscillating frequency f, and its orientation is tuned by different joint's deflection $\{\phi_1, \phi_2, \phi_3, \phi_4\}$, where, ϕ_1, ϕ_2, ϕ_3 and ϕ_4 are joint angle for 4 links respectively. Hence a parameters set $\{\phi_1, \phi_2, \phi_3, \phi_4, f\}$ can be reduced to control fish's motion. Since the mechanical robot fish is composed of four links, all calculations and experiments in this paper are carried out on a four-link model.

III. SPEED CONTROL ALGORITHM

As robot fish works in water, the regulation of its body (center of mass) speed is realized by changing servomotors' oscillatory frequency. Just for this, there are some unfavorable factors against the robot fish's speed control. On one hand, interaction between robot fish and surrounding fluid medium will lead to resonance at the resonant frequency, accompanying with robot fish's rolling along the fish body's axis and yawing along axis perpendicular to water surface. On the other hand, for the sake of lacking necessary stopping mechanism, we can't stop robot fish at once due to the momentum conservation. Even if the speed of each joint drops to zero, fish body will still drift a short distance along the current direction. So how to find a trade-off between speed and energy is crucial for robot fish's steady motion.

Through a lot of experiments we find a steady, maximum speed V_f at which the rolling and swaying of fish body are minimum. Like elevator control, an acceptable trade-off between energy and stability can be realized by carefully manipulating the moving speed. We limit inertia forces by limiting the maximum acceleration to a value of A_m during robot fish swims from stationary state to maximum steady speed V_f; at the same time, robot fish starts decelerating when the distance between fish and goal equals L_{cs} (acceleration keeps $-A_m$ during deceleration), at last drifts to goal by means of inertial forces. By using speed distribution function shown in formula (2), a speed profile like "S" is produced. As shown in Fig.3, the motion process can be divided into four phases: acceleration phase, constant phase, deceleration phase and drift phase. Different speed strategies are taken at different phase so that the robot fish moves rapidly and steadily.

$$V(t) = \begin{cases} 0.5V_f(1-\cos(\pi t / T)) & (0 < t \le T) \\ V_f & (T < t \le T_d) \\ 0.5V_f(1-\cos(\pi(t - T_d - T)/T)) & (T_d < t \le T_d + T) \\ 0 & (t > T_d + T) \end{cases} \qquad (2)$$

where $T = (\pi / 2)(V_f / A_m)$, A_m can be determined experimentally in advance.

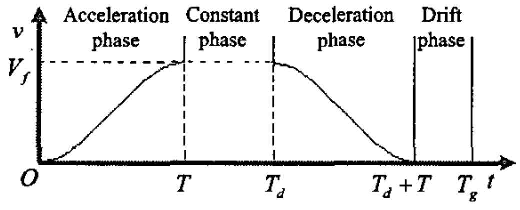


Fig.3 Speed profile of robot fish

For a given speed, a PID controller is used for a desired speed. The control structure with a PID controller is shown in Fig.4, where V_{set} is the desired speed, f_{set} is the expected oscillatory frequency derived from speed-oscillatory frequency function $f_1(v)$, fe denotes the error of oscillatory frequency, V is the body speed of the swimming fish, which is measured by a hanged visual subsystem, f is feedback oscillatory frequency derived from the function $f_2(v)$. Here, the speed-oscillatory frequency functions $f_1(v)$ and $f_2(v)$ take the same form. For simplicity, they can be fitted by a linear speed-oscillatory frequency function using the experimental data. In particular, when the error e nears to zero, PID doesn't work. The digitized PID control algorithm is referred to Tao et al [9].

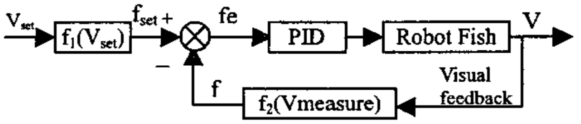


Fig.4 The structure of PID controller for a desired speed

IV. ORIENTATION CONTROL USING FUZZY LOGIC

As discussed in section II, the robot fish can be navigated to the desired position with a certain speed by choosing four proper joints angles $\{\phi_1, \phi_2, \phi_3, \phi_4\}$. As discussed in Hirata et al [10], there are three turning modes for the fish propelled only with oscillating tail fin. Combining with experiment on the robot fish, we redefine these three modes during turning: (A) turning during advancing, (B) acute turning, and (C) turning from rest. Combined with these turning modes, perhaps, path planning can be implemented if having a high quality control system for motion of servomotors. Since the Fuzzy Logic Algorithm introduced by Zadeh is effective for some control of highly nonlinear problem, here another application of fuzzy logic algorithm will be used for robot fish's orientation control. The objective is to make a fuzzy logic controller that generates deflections of the first two joint angles when robot fish moves from any initial position to its destination position. In the method, the deflections of the joint angle are added to the first two joint angles $\{\phi_1, \phi_2\}$ so that the fish can turn with different turning radius.

A fuzzy controller works in a similar way to a conventional system: it has an input value, performs some calculations, and generates an output value. This process is called the Fuzzy Inference Process and works in three steps: (a) fuzzification, where a crisp input is translated into a fuzzy value, (b) rule evaluation, where the fuzzy output truth values are computed, and (c) defuzzification, where the fuzzy output is translated to a crisp value [11].

The fuzzy orientation function control inputs are shown in Fig.5. The three state variables Fx, Fy and θ_f determine the current fish position. θ_f specifies the angle of the fish with respect to the horizontal. Theses variables are defined by the vision subsystem. The coordinate pair (Px,Py) specifies the position of the destination of the fish. The structure of fuzzy controller for orientation control is given in Fig.6. This fuzzy controller takes two inputs and produces two outputs. These inputs are θ_e and θ_{ec}. As shown in Fig.6, θ_e is the difference angle between desired angle θ_d and the fish's current angle θ_f, θ_{ec} denotes ratio of angular error. The outputs of fuzzy control are deflections of joint angle for the first two joints: u_1 and u_2, which will be used for various orientation adjusting. For simplicity, the same membership function and fuzzy rules is applied to u_1 and u_2, so during fuzzification and rule evaluation they can be viewed as one output variable u ($u=[u_1\ u_2]$), but in process of defuzzification they are multiplied by different scales: k_3 and k_4, respectively. The ranges of input and output variable values are: $-20° \le \theta_e \le 20°$, $-40° \le \theta_{ec} \le 40°$, $-240 \le u_1 \le 240$, $-200 \le u_2 \le 200$, where, the value of θ_e can be positive or negative, and a positive value signifies that the fish turns right otherwise fish turns left. The variables: θ_f, θ_{ec} and u are all graded to 13 levels from –6 to 6 (i.e.{-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6}).

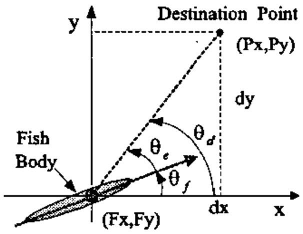


Fig.5 Control inputs

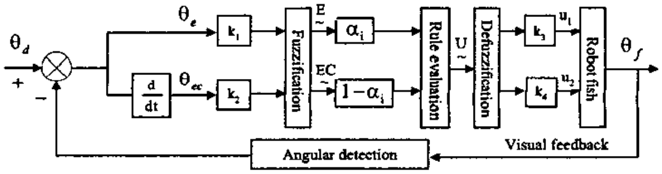


Fig.6 Structure of fuzzy controller for orientation control

The first step in developing a fuzzy logic system is to represent the fuzzy set variables into linguistic terms. The angular error θ_e, error ratio θ_{ec} and output variable u are graded into 7 levels represented as NB (negative big), NM (negative medium), NS (negative small), ZE (zero), PS (positive small), PM (positive medium), PB (positive big), and corresponding fuzzy variables denote as $\underset{\sim}{A}_i (i=1,2,\ldots,7)$, $\underset{\sim}{B}_j (j=1,2,\ldots,7)$ and $\underset{\sim}{C}_k (k=1,2,\ldots,7)$. The membership functions of θ_e and θ_{ec} are determined by experiment through actual turning test, and a triangular shaped membership function for U is defined.

At the rule evaluation step, a 2 dimensional (7×7) Fuzzy Associative Memory (FAM) matrix is formulated. The FAM matrices containing 49 rules can be interpreted as antecedent-consequent pairs or IF-THEN statements:

IF E is $\underset{\sim}{A}_i$ AND EC is $\underset{\sim}{B}_j$ THEN U is $\underset{\sim}{C}_{ij}$

Where $i=1,2,\ldots,7$, $j=1,2,\ldots,7$.

Here, a maximum- minimum inference method is used to evaluate the entire set of rules, and 31 rules are fired up.

At the defuzzification step, a calculation method called the Center of Area (COA) is used in order to produce the crisp output value of u for deflection of the joint angle. In reality, after the crisp value is multiplied by scalar factor (k_3, k_4) (k_3=35, k_4=40), the outputting angular variation of the first two servomotors is obtained.

V. PTP CONTROL ALGORITHM

In the previous section, the basic speed control and orientation control algorithms are discussed for the robot fish prototype. In this section, we explore the implication of these for steering the fish from an arbitrary initial position to destination point in 2-D Euclidean Space (*SE*(2)). Before demonstrating our method, we recall the results of Dubins for kinematic mobile robots [12]. It was shown that the optimal path for a car steering in *SE*(2) with limits on the turning radius is given by two circular arcs (one arc tangent to each of the initial and final headings) connected by a straight line segment. However, controlling the robot fish turning with precise radius is difficult at present. Here, we assume a possible path connecting initial point and destination point is just a straight line. This path is not necessary optimal but accessible for the fish.

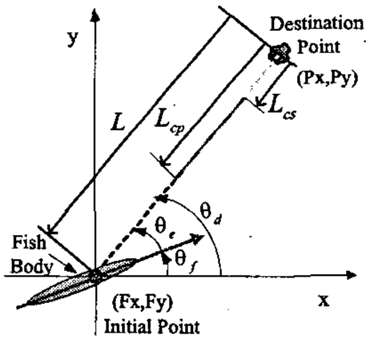


Fig.7 Decomposition of PTP control

In order to realize PTP (Point To Point) control of robot fish, the strategy we choose is to get of the error of the orientation between the fish body and the line from the initial point (Fx,Fy) to the destination point (Px,Py) while advancing along the specified line. The ideal PTP unit position vector (V_{PTP}) is given in formula (3):

$$V_{PTP} = \frac{1}{\sqrt{(\mathrm{Px}-\mathrm{Fx})^2+(\mathrm{Py}-\mathrm{Fy})^2}}\begin{bmatrix}\mathrm{Px}-\mathrm{Fx}\\ \mathrm{Py}-\mathrm{Fy}\end{bmatrix} \tag{3}$$

Where, as specified in Fig.5, the coordinate pair (Px,Py) specifies the position of the destination of the fish, the coordinate pair (Fx,Fy) is the current fish position. These positional variables and orientation are defined by the vision subsystem.

For robot fish works in hydrodynamic environment (in the following experiment, the robot fish works in an environment with no moving fluid), on one hand, a mathematical model for decision-making is enormously tough to create; on the other hand, robot fish drifts with fluctuate of water surface due to remaining momentum. In addition, it is hard to track straight-line. So there are great difficulties in precise PTP control.

On the basis of step control, as schematic show (Fig.7), different strategies are chosen according to different distance (l) between the fish body (RobFish) and the destination point. Here, the measure that is taken is from crude to fine. If $l>L_{cp}$, RobFish speed up to approach destination; If $L_{cs}<l<L_{cp}$, accurate control is utilized, that is, RobFish slow down and approaches within defined orientation error; If $l<L_{cs}$, RobFish set straight (adjust four joints to zero-position which is in a line with the forebody), drifting onward with zero-joint-speed. The detailed algorithm is as follows:

PTP CONTROL ALGORITHM

Step0. Initialize environment and destination point (Px, Py) of RobFish, RobFish set straight.

Step1. Update information of environment and RobFish (including its speed and orientation) obtaining from vision subsystem, and calculate the distance difference l and the orientation difference θ_e.

If $l < LengthError$ AND $|\theta_e| < OrientationError$

(*LengthError* and *OrientationError* are the algorithm's terminating condition for the distance difference and orientation difference, respectively, which are identified by actual experiments). RobFish stops oscillating and sets straight, and the algorithm exits. Otherwise, go to Step2.

Step2. Call the speed control algorithm to plan RobFish's speed strategy according to the value of l.

I. Get RobFish's current l and v, if $l<L_{cs}$, go to III; otherwise, go to II.

II. 1) If $v<V_f$, speed up (increase the oscillatory frequency f), and go to Step3;

2) If $v=V_f$, keep uniform speed using PID controller (keep the oscillatory frequency f) and go to Step3.

III. 1) If v=0, check if RobFish is in start state:

(1) If it is true, let $v=V_f$, and go to Step3;

(2) If it is false, let RobFish set straight and drift to destination, and go to Step1.

2) If $v \neq 0$, slow down till it reaches zero-body-speed (decrease the oscillatory frequency f), and go to Step3.

Step3. Combining the fish's turning modes, use fuzzy controller for orientation control to plan RobFish's orientation strategy according to the value of θ_e.

I. Get RobFish's current θ_e, if $|\theta_e|$ >*AngleThreshod* (*AngleThreshod* is also identified by actual experiments), go to II; otherwise, go to III.

II. 1) If θ_e <-*AngleThreshod*, perform acute-turning-right in turning Mode B, and go to Step4;

2) If θ_e >*AngleThreshod*, perform acute-turning-left in turning Mode B, and go to Step4;

III. Calculate the error ratio θ_{ec}, according to values of θ_e and θ_{ec}, use fuzzy logic controller to determine the angular variation of the first two servomotors, and go to Step4;

Step4. Transfer results from Step2 and Step3 to fish's control parameters set $\{\phi_1, \phi_2, \phi_3, \phi_4, f\}$, then send them to RobFish by radio control module, and go to Step1.

Based on above algorithm, a steering function **MoveToGoal**(CPoint destpt, double dir) is designed and applied to the following experiments.

VI. EXPERIMENTAL SYSTEM AND RELATING RESULTS

To verify the feasibility and reliability of algorithms, an experimental robot fish system has designed and developed. The system is composed of four subsystems: robot fish subsystem, vision subsystem, decisions-making subsystem and communication subsystem. All aquatic experiments presented in this paper are carried out in a 2000mm × 1150mm pond with still water. The information of fishes and their surrounding captured by overhead CCD camera, is effectively processed and sent to decision-making module as input, then output of decision-making subsystem is transmitted to single robot fish through communication subsystem, thus robot fishes work effectively. In our vision subsystem, robot fishes, ball, and obstacles are equipped with specified colors. To locate robot fish and other objects quickly and accurately, a parallel visual tracking algorithm based on color information has developed, mainly by adaptive segmentation and a closure operation [13]. In particular, the visual tracking is performed in real-time, and provides a feedback signal to robot fish control.

Using the vision-based tracking system to provide real-time feedback, we perform two experiments with robot fish designed to evaluate our control strategies:

A. Experiment A: playing ball

In the pond with still water, a floating ball (Radius 45mm) is regarded as a target, and the robot fish swims to the ball from an arbitrary initial position and orientation. The fish (Fx,Fy, θ_f)and the ball (Px,Py) are located by vision subsystem. By calling the steering function MoveToGoal(CPoint destpt, double dir) (where, destpt=CPoint(Px,Py), dir=0) continuously, the fish swims toward the ball, and sometime push it. Because ball is too light to remain still, the fish lost it and pushed it again just like playing a game. An experimental playing-ball video is available at the website (http://compsys.ia.ac.cn/robotfish/singlefishen.html). Fig.8 (a) shows a photo of experimental scenario during playing ball, Fig.8 (b) shows a moving trajectory of the fish swimming towards ball, where, the positions of fish and ball are denoted in image plane coordinates in which the whole view field is a plane with 320 pixel × 240 pixel, Fig.8 (c) shows corresponding orientation difference θ_e, where the characteristic points are sampled at a interval of 0.3 seconds.

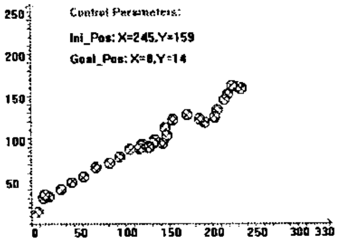


Fig.8 (a) Scenario of playing ball Fig.8 (b) Moving trajectory

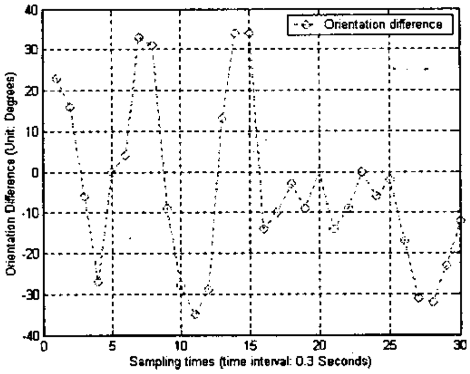


Fig.8 (c) Orientation difference θ_e

B. Experiment B: passing a hole

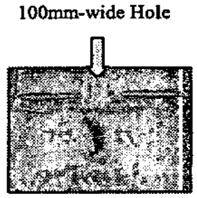


Fig.9 Scenario of passing hole

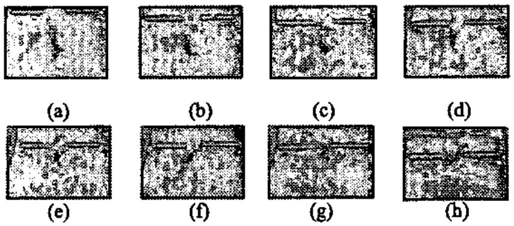


Fig.10 Image sequence of passing hole (from (a) to (h))

To test controllability of robot fish in a narrow space, a clearance of 100mm between two bars aligned in a line was considered as "HOLE". For the robot fish, its task was to pass through the specified hole from an arbitrary initial position and orientation. The fish (Fx,Fy, θ_f) and the Hole(Hx,Hy) (marked with predefined color) are located by vision subsystem. When the fish was far from the hole (i.e. *l>FISHLEGTH*), the steering function MoveToGoal(CPoint destpt, double dir) was called continuously, where destpt=CPoint(Hx,Hy) and dir=0, so that the fish swam toward the hole. When the fish was near the hole (*FISHFOREBODY<l<FISHLEGTH*), the orientation

difference θ_e (degrees) was checked to see whether it lies between –20 and 20. If yes, let the fish set straight and move forwards; otherwise, call the steering function to adjust its orientation till it meets the requirement. When the fish was in the hole (*–FISHFOREBODY<l<FISHFOREBODY*), let the fish swim with full-speed V_f. After the fish passed through the hole (*l<–FISHFOREBODY*), the above-mentioned algorithm was repeated to let fish pass again from the contrary direction. Here, the constants (*FISHLEGTH, FISHFOREBODY, FISHFOREBODY*) were specified by experiment. Fig.9 shows the scenario of passing hole, and Fig.10 demonstrates an image sequence of passing the hole. The experimental video is also available at the website (http://compsys.ia.ac.cn/robotfish/singlefishen.html).

C. Discussions

Seen from Fig.8(c), the orientation difference is fluctuant between –40 degrees to 40 degrees, this implies that the fish body is oscillating during moving. The body's oscillations are related to body's advance speed to some extent, which can be reduced partly by careful speed control. In essence, it is considered that the balance of gravitation and buoyancy at the flapping tail effects propulsive performance and steadiness. An addition pectoral mechanism maybe lend itself to fish body's steadiness. Lacking backward action, if the robot fish is given a target behind its initial position, it has to be turn around. Hence, the combination of various turning modes must be discussed. The fuzzy logic is effective in a range of orientation difference, but the membership functions are very hard to determine.

VII. CONCLUSIONS AND FUTURE WORK

We have developed an experimental system for closed-loop control of 4-link and free-swimming biomimetic robot fish. The speed of the fish is adjusted by modulating the joint's oscillating frequency, and its orientation is tuned by different joint's deflection. Therefore, the fish's motion control task, based on control performance of the fish, is decomposed into on-line speed control and orientation control. We propose an algorithm for speed control, and design a fuzzy logic controller for orientation control. On the basis of speed and orientation control, a point-to-point control algorithm is realized. Experiments with this system have demonstrated the good performance of the robot fish using vision-based position sensing feedback.

Future work with the developed robot fish system will involves expanded closed-loop control, performing a full planar motion planning algorithm for complex and cluttered environments with obstacles based on visual feedback. At the same time, simple sensors like ultrasonic and infrared detectors are planned to add to the fish body so that the robot fish owns local autonomy. We also plan to add new degrees of freedom (Up/Down) to the robot fish that will enable it to navigate in a 3-D workspace. Eventually, based on former sensor and control technology, an autonomous robot fish that can swimming skillfully (high efficiency) and intellectually (autonomous obstacle-avoidance with its own sensors) will be developed.

VIII. ACKNOWLEDGMENTS

This work is funded by research grants from the robotics subject of 863 Program (Grant No. 2001AA422370).

IX. REFERENCES

[1] M. Sfakiotakis, D.M. Lane, and J.B.C. Davies, "Review of fish swimming modes for aquatic locomotion," *IEEE Journal of Oceanic Engineering*, 1999, vol. 24, pp. 237-252.

[2] B.G. Tong, "Propulsive mechanism of fish's undulatory motion," *Mechanics In Engineering*, 2000, vol. 22, pp. 69-74 (in Chinese).

[3] M.S. Triantafyllou, and G.S. Triantafyllou, "An efficient swimming machine," *Scientific American*, 1995, pp. 64-70.

[4] J.M. Anderson, M.S. Triantafyllou, P.A. Kerrebrock, "Concept design of a flexible-hull unmanned undersea vehicle," *Proceedings of the International Offshore and Polar Engineering Conference*, 1997, pp. 82-88.

[5] M. Mojarrad, "AUV biomimetic propulsion," *Oceans Conference Record (IEEE)*, 2000, pp. 2141-2146.

[6] M.S. Triantafyllou, G. S. Triantafyllou, and D.K.P. Yue, "Hydrodynamics of fishlike swimming," *Annu. Rev. Fluid Mech.*, 2000, vol. 32, pp. 33-53.

[7] F. Masahiro, "AIBO: toward the era of digital creatures," *IJRR*, 2001, vol. 20, pp. 781-794.

[8] D. Barrett, M. Grosenbaugh, M. Triantafyllou, "The optimal control of a flexible hull robotic undersea vehicle propelled by an oscillating foil," *Proceedings of the 1996 IEEE Symposium on Autonomous Underwater Vehicle Technology*, 1996, pp. 1-9.

[9] Y. Tao, Y. Yin, L. Ge, *New-style PID control and its application*, Beijing: Mechanical publishing house, 1998 (in Chinese).

[10] K. Hirata, T. Takimoto, and K. Tamura, "Study on turning performance of a fish robot," *First International Symposium on Aqua Bio-Mechanisms*, 2000, pp. 287-292.

[11] J. Zhu, *Mechanism and application of Fuzzy Control*, Beijing: Mechanical publishing house, 1995 (in Chinese).

[12] L.E. Dubins, "On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, 1957, vol. 79, pp. 497-516.

[13] Junzhi Yu, Shuo wang, Min Tan. A parallel algorithm for visual tracking of multiple free-swimming robot fishes based on color information, in *IEEE International Conference on Robotics, Intelligent Systems and Signal Processing 2003*, October 2003, Changsha, China.