Vision-Based Autonomous Hovering for the Biomimetic Underwater Robot—RobCutt-II

Rui Wang, Shuo Wang, Yu Wang, Mingxue Cai, and Min Tan

***Abstract*—This paper aims to investigate the method of autonomous hovering control for the biomimetic underwater robot named RobCutt-II. First, a modular configuration for mimicry of undulatory propulsion is presented, followed by a coordinated control of dual fins for enhancing the maneuverability and stability of the RobCutt-II. Then the vision-based hovering problem is formulated. A dual-loop hovering control framework integrating an image processing module, a tracking differentiator, a hovering controller, a velocity controller, and a fuzzy-based control allocation module is proposed to govern the locomotion of the RobCutt-II. Finally, simulations and experimental results demonstrate the effectiveness of the proposed mechanical design and control methods.**

***Index Terms*—Autonomous hovering, biomimetic underwater robot, dual fins, undulatory propulsion, visual servoing.**

I. Introduction

IN RECENT years, underwater robots are playing an increasingly important role in ocean exploration [1], [2]. Autonomous hovering in aquatic environments is a critical capability during the deployment of underwater robots in numerous practical applications [3]. Being a free-floating device, underwater robots cannot hover naturally in water. Specifically, an underwater robot must deal with unknown currents and forces, which could cause its position and attitude to change undesirably. Hovering mode has been reported to take 35% of the total work done in a typical subsea task. As a result, autonomous hovering plays a vital role in the development of underwater robots.

For more than ten years, the hovering control or station keeping control for autonomous underwater vehicles (AUVs) or surface ships has received considerable attention from worldwide researchers and various control strategies have been proposed [4]–[7]. As for the hovering control of AUVs, Zhao *et al.* presented some theory and experimental work of the adaptive plus disturbance observer controller for position and attitude control of an AUV ODIN III [8]. Stone Aerospace developed a four degrees of freedom (DOF) hovering AUV named ENDURANCE with six thrusters, whose controller block uses proportional-integral-derivative (PID) loops to hold the desired position and heading during station keeping [9]. Santhakumar *et al.* proposed a control scheme for station keeping of a flat-fish type AUV with the addition of dedicated thrusters to reduce power consumption [10]. Jin *et al.* designed a selective switching controller to solve the hovering control problem of an underwater robotic platform with four tilting thrusters [11].

Dynamic positioning (DP) systems maintain ships in fixed position or predetermined track exclusively by means of thrusters and main propellers aft of the ship [12]. Sorensen gave an overview of some of the major research advances in the DP controller design [13]. Katayama considered the nonlinear sampled-data stabilization of dynamically positioned ships [14]. Tannuri *et al.* presented a DP system composed of a feedforward compensator, a proportional-derivative feedback controller, and an extended Kalman filter. Station keeping experiments on a reduced model of a tanker with three thrusters were carried out to validate the control scheme [15].

Vision-based hovering, with the properties of low cost and high precision, attracts much attention, although it is susceptible to nonuniform illumination and low-quality imaging [16], [17]. Instead of maintaining an absolute position and attitude in underwater space, vision-based hovering generally stabilizes underwater robots relative to some visual landmarks, which is expected to be used in underwater object grasping, underwater photography, and other missions. Researchers have made some attempts to use vision sensors for hovering control [18]–[20]. Marks *et al.* presented some pioneer work in visual servoing control for hovering in 1994 [21]. Hereafter, Lots *et al.* introduced a two-dimensional (2-D) visual servoing technique for the station keeping of an unmanned underwater vehicle with respect to planar unmarked objects on the sea bed [22]. Negahdaripour *et al.* proposed an automatic optical hovering system for an underwater vehicle with three thruster. Specially, they focused on how to

Manuscript received April 20, 2018; revised August 3, 2018 and October 28, 2018; accepted December 2, 2018. Date of publication December 24, 2018; date of current version June 28, 2019. This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB1300103, and in part by the National Natural Science Foundation of China under Grant U1713222. *(Corresponding authors: Shuo Wang and Yu Wang.)*

R. Wang and Y. Wang are with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: rwang5212@ia.ac.cn; yu.wang@ia.ac.cn).

S. Wang is with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, with the University of Chinese Academy of Sciences, Beijing 100049, China, and also with the CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai 200031, China (e-mail: shuo.wang@ia.ac.cn).

M. Cai and M. Tan are with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: caimingxue2017@ia.ac.cn; min.tan@ia.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2018.2886755

estimate the vehicle's translational and yaw motion by exploiting the information in sea floor images [23]. Zwaan *et al.* developed a decoupled control system for automatic vision-based station keeping of an underwater vehicle with nonholonomic motion constraints, relative to some planar landmarks [24]. Gao *et al.* designed an adaptive neural network image based visual servo controller for dynamic hovering of an underwater vehicle [25]. Wang *et al.* presented a neural network based visual servo control to achieve hovering for a bio-inspired robot. The validity of the method was demonstrated by simulation [26].

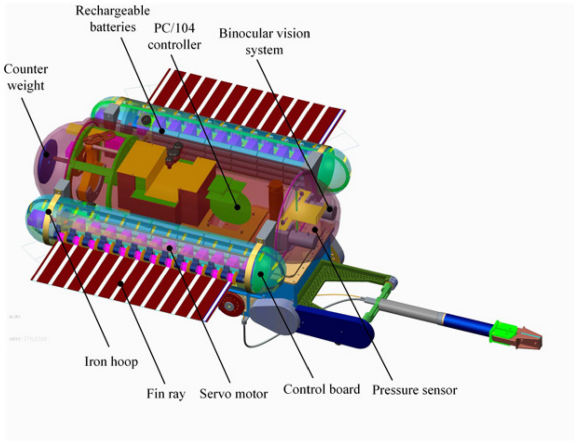
Most of the above literatures have achieved acceptable performance under certain conditions, but the controllers were generally designed for AUVs or surface ships using conventional screw propellers. When these underwater robots perform low speed motion, such as hovering, the screw propellers are in nonfull-rotation states, which may significantly reduce its efficiency and produce unpredictable fluid pulses. In addition, the screw propellers are easy to wrap around the water grass, causing power failure. In contrast, some fish, such as cuttlefish or stingrays, use their flexible undulatory fins for moving and hovering, exhibiting astounding swimming maneuverability and hovering stability. Taking inspiration from nature, mimicking the unique undulatory propulsion mode may help the underwater robot to perform stable and effective hovering in confined spaces with external disturbances.

Based on our previous work on undulatory propulsion and its control methods [28], [29], this paper focuses on the design and implementation of autonomous hovering control for a biomimetic underwater robot named RobCutt-II. The RobCutt-II adopts undulatory propulsion mode, obtaining good low-speed locomotion stability. In turn, the propulsion modes and control inputs of the RobCutt-II are quite different from that of AUVs and surface vessels, such that the model uncertainty and hydrodynamic complexity are more challenging and pose challenges to the design of the hovering control system.

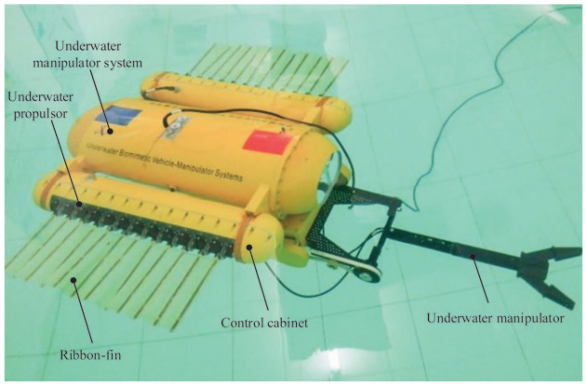
The main contributions of this paper are summarized as follows:

1) development of a vision-based hovering control framework integrating an image processing module, a tracking differentiator (TD), a hovering controller, a velocity controller, and a fuzzy-based control allocation module;
2) practical validation of the proposed control strategy on a biomimetic underwater robot prototype prototype.

In particular, we consider the autonomous hovering based on visual input. The hovering task is defined locally in the neighborhood of some visually observable underwater targets and consists in stabilizing the underwater robot relative to the target so as to reject external disturbances. Note that our system uses binocular vision to perform visual hovering, which helps position the underwater robot relative to more underwater targets, instead of only planar landmarks. In the end, we investigate the performance of the proposed control method through simulations and experiments with a RobCutt-II prototype. According to our best knowledge, the closed-loop control for autonomous hovering of an underwater vehicle-manipulator system propelled by undulatory fins is realized for the first time.



(a)



(b)

Fig. 1. System design of the RobCutt-II. (a) Three-dimensional (3-D) model. (b) Prototype.

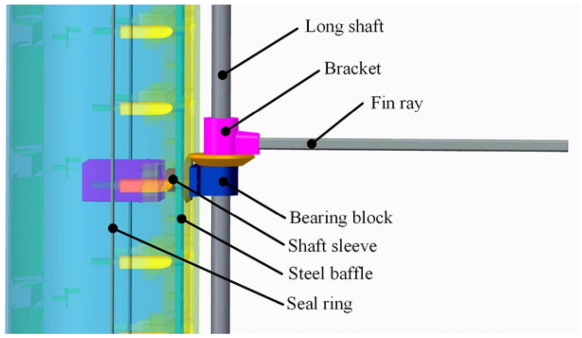
In the remainder of this paper, Section II presents the system overview of the biomimetic underwater robot named RobCutt-II. Section III details the proposed vision-based hovering control framework. Simulations and experimental results are further provided in Section IV. Finally, Section V concludes this paper with an outline of future work.

II. System Overview of the RobCutt-II

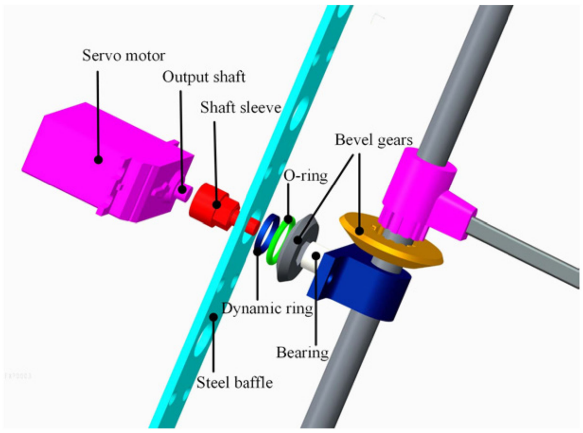
A. Mechanism of the RobCutt-II

Fig. 1 shows the system design of the bio-inspired underwater robot named RobCutt-II (ROBotic CUTTlefish). Specifically, the mechanical design of the RobCutt-II is based on modular concepts. The RobCutt-II can be divided into three modules, i.e., an underwater manipulator system and two underwater propulsors with fin propulsion. The undulatory propulsion mode may help the RobCutt-II to obtain good low-speed locomotion stability when performing underwater survey and intervention tasks. The detailed description about the mechanism of the underwater manipulator system can be found in [28].

On the other hand, two modular underwater propulsors are mounted on both sides of the underwater manipulator system symmetrically by iron hoops. It should be mentioned that



(a)



(b)

Fig. 2. Dynamic seal mechanism. (a) Mechanism of each fin ray. (b) Exploded view of the dynamic seal.

underwater propulsors can be conveniently installed and easily replaced, which facilitates maintenance and future development of the biomimetic underwater robot. In particular, the underwater propulsor consists of a control cabinet and a ribbon fin. Some electronic components, including a control board, batteries, and servo motors, are installed inside the control cabinet. Moreover, the ribbon fin consists of 12 fin rays, which are connected to one another by a flexible membrane made of thin rubber isometrically. Fig. 2 shows the dynamic seal mechanism of the underwater propulsor. As shown in Fig. 2(a), the oscillating motion of each fin ray is independently driven by the rotation of its corresponding servo motor through a gear transmission system. The gear transmission system is composed of two bevel gears that are coupled orthogonally. In order to increase the propulsive force of undulatory fins, the transmission ratio between two gears is set to 1.5:1. The exploded view of the dynamic seal is shown in Fig. 2(b). As can be observed, the output shaft of the servo motor connects to the shaft sleeve, which extends to the outside of the control cabinet through the shaft hole. A Glyd ring is attached to the shaft sleeve and a steel baffle is fixed to lock the Glyd ring. Note that the Glyd ring is a typical dynamic seal product, consisting of a dynamic ring made of polytetrafluoroethylene for high wear resistance and a rubber O-ring for elastic compensation. After a host of waterproofing tests, underwater propulsors can work normally in a pool with a depth of 3 m.

TABLE I
TECHNICAL SPECIFICATION OF THE ROBCUTT-II PROTOTYPE

Parameters	Value
Total size ($L \times W \times H$)	Approx. 1230 mm × 970 mm × 390 mm
Total mass	Approx. 52 kg
Fin size ($L \times W \times H$)	Approx. 460 mm × 165 mm × 0.82 mm
Number of fin rays	12 fin rays on each propulsor
Space of fin rays	43 mm
Onboard sensors	Pressure sensor, binocular vision system
Drive mode	DC motors, digital servo motors
Power supply	22.2 V and 7.4 V rechargeable Li-polymer batteries
Operation time	Approx. 3 h

The basic feature parameters of the RobCutt-II prototype are listed in Table I.

B. Coordinated Control of Dual Fins

Based on empirical observations of cuttlefish locomotion, we approximate the ribbon-fin motion as sinusoidal wave traveling from one end of the fin to the other. Meanwhile, in order to avoid the motion in sway and roll, and maintain dynamic stability of the RobCutt-II, we set the amplitudes of bilateral traveling waves equal to each other and so do the deflection angles and the wave lengths. Thus, the kinematic model of dual fins can be defined as follows:

$$\begin{cases} \theta_L(i,t) = A_L \cos[2\pi F_L t - (i-1)\phi_L] + \theta_{BL} \\ \theta_R(j,t) = A_R \cos[2\pi F_R t - (j-1)\phi_R] + \theta_{BR} \\ i,j = 1,2,\cdots n \end{cases} \tag{1}$$

$$\text{s.t.} \begin{cases} A_L = A_R = A \\ \phi_L = \phi_R = \phi \\ \theta_{BL} = \theta_{BR} = \theta_B \end{cases} \tag{2}$$

where θ_L, θ_R are the angular deflection of traveling waves on left fin and right fin, respectively, i, j represent the ith and jth fin ray, t is time, n is the number of fin rays, F, A, ϕ, θ_B are the frequency, amplitude of waves, the phase difference between two adjacent fin rays, and the deflection angle of the ribbon fin, respectively. According to the above model, the locomotion of the RobCutt-II can be determined by F_L, F_R, A, ϕ, θ_B, so we choose these five parameters as the control parameters of the RobCutt-II. Moreover, considering the elastic constraints of undulatory fins and the speed limits of servo motors, we limit the range of control parameters to $\text{CP} = \{(F_L, F_R, A, \phi, \theta_B) \in \mathbb{R}^5 : |F_L| \le 2\,\text{Hz}, F_R \le 2\,\text{Hz}, 10^\circ \le A \le 40^\circ, 0^\circ \le \phi \le 50^\circ, |\theta_B| \le 50^\circ\}$ to reduce the probability of failure problems of fins. Note that the sign of fin frequency represents the direction of the traveling waves.

In the view of practical implementation, a PC/104 module installed in the underwater manipulator system is employed as the core controller, which connects to the control board of bilateral underwater propulsors through serial ports. After receiving a control command from the PC/104 module, the control board checks the command, parses out waveform

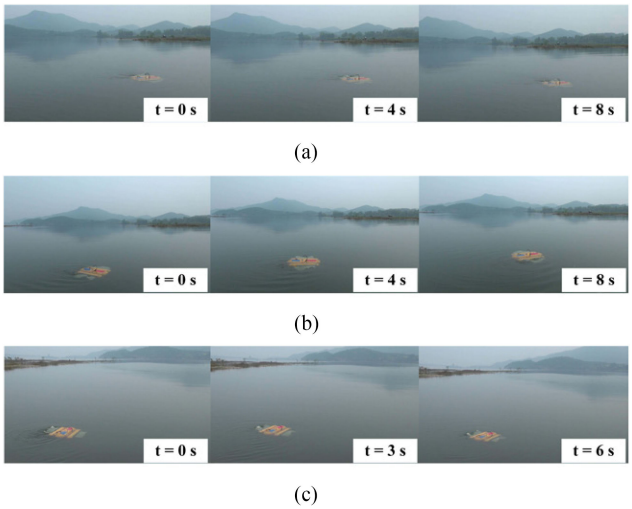


Fig. 3. Image sequences of multiple motion patterns in field experiments. (a) Forward marching. (b) Left turning. (c) Right turning.

parameters, calculates the angular deflection of each fin ray, and finally sends calculated angle values to corresponding servo motors under the specific communication protocol. With the coordinated control of dual fins, the RobCutt-II can accomplish multimodal motion, including forward/backward swimming, diving/surfacing motion, and turning maneuver with high maneuverability and stability.

Field experiments were performed at the Beijing Ming Tombs Reservoir. Fig. 3 shows the image sequences of multiple motion patterns in field experiments, where Fig. 3(a) shows the forward marching of the RobCutt-II, left turning maneuver is shown in Fig. 3(b), and Fig. 3(c) shows the results of right turning.

III. Vision-Based Hovering Control Design

A. Problem Statement

As shown in Fig. 4, consider a scenario in which the RobCutt-II needs to hover in front of an underwater target for observation or intervention task. The goal of hovering control is to drive the RobCutt-II to keep the desired view in the presence of external disturbances. Note that the RobCutt-II is an underactuated system and cannot directly move laterally, it cannot autonomously maintain in a fixed position in space. So our control objective is to keep the underwater target in the desired position on the image plane, while maintaining a fixed distance between the RobCutt-II and the underwater target.

B. Control System Framework

In order to solve the above problem, a dual-loop hovering control framework is proposed as shown in Fig. 5. The integrated control design mainly consists of an image processing module, a TD, a hovering controller, a velocity controller, and a fuzzy-based control allocation module.

More precisely, the control system inputs are the reference visual feature and distance. A binocular vision system is applied to provide live 320 × 280 pixels' images of the underwater target as feedback inputs. The target position and actual visual

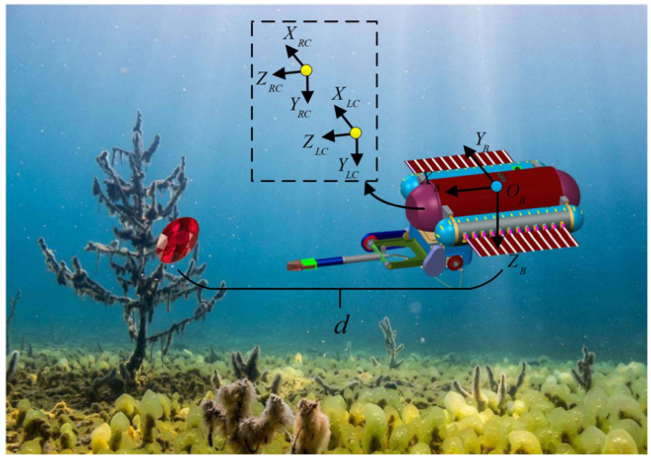


Fig. 4. Diagram of autonomous hovering of the RobCutt-II. The underwater target (here is a ruby) is located near the left of the center. $O_B X_B Y_B Z_B$ is the body-fixed frame, $O_{LC} X_{LC} Y_{LC} Z_{LC}$ and $O_{RC} X_{RC} Y_{RC} Z_{RC}$ are, respectively, the left camera frame and the right camera frame with the origins at the focal points (yellow points), d is the distance from the barycenter of the RobCutt-II (blue point) to the underwater target.

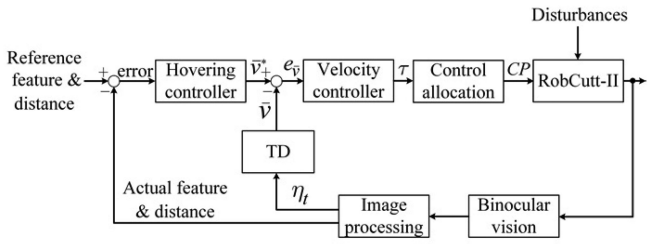


Fig. 5. Structure of the dual-loop hovering control framework of the RobCutt-II. $\bar{v}^* \in \mathbb{R}^3$ is the control output of hovering controller, $\bar{v} \in \mathbb{R}^3$ denotes the estimation of the controllable velocity vector, $e_{\bar{v}} = \bar{v}^* - \bar{v} \in \mathbb{R}^3$ represents the velocity error, $\tau = [\tau_u\, \tau_w\, \tau_r]^T \in \mathbb{R}^3$ is body-fixed propulsive force and moment, where τ_u, τ_w, τ_r describe the propulsive force or moment acting on surge, heave and yaw, respectively, CP denotes the control parameters of the RobCutt-II, $\eta_t = (x_t, y_t, z_t) \in \mathbb{R}^3$ is the calculated target position in the body-fixed frame.

feature are obtained by image processing, which includes image segmentation, feature extraction, image filtering, 3-D reconstruction, etc. For simplicity, the object with specific color feature is selected as the underwater target in this paper. We remark that other features could be used according to actual conditions of applications. The threshold segmentation is applied to obtain the target area, because some pixels with similar color feature in the image may mistakenly belong to the target, all the contours in the image should first be extracted in the binary image, and the contour with the largest area is chosen as the target. The center of all contour pixels is served as the position of the target center in the image. Then the target center in the images of both cameras can be calculated. Without loss of generality, we choose the target center of the right camera as visual feature. Furthermore, the position η_t of the target center in the body-fixed frame $O_B X_B Y_B Z_B$ and the distance d can be obtained by 3-D reconstruction [27]. Readers may refer to [28] for detailed description of the image processing.

Furthermore, a TD is used to output the estimations of the controllable velocity components of the RobCutt-II. A hovering

controller is proposed to establish the link between the error space and the velocity space. Then, a PID-based velocity controller outputs propulsive force and moment according to the velocity error, which can mathematically be described as

$$\tau = K_{p\bar{v}} e_{\bar{v}} + K_{i\bar{v}} \int e_{\bar{v}} + K_{d\bar{v}} \dot{e}_{\bar{v}} \tag{3}$$

where $K_{p\bar{v}}$, $K_{i\bar{v}}$, $K_{d\bar{v}}$ are 3×3 adjustable diagonal matrices.

As mentioned before, the control parameters of the RobCutt-II are the traveling wave parameters on bilateral fins. The relationship between the propulsive force/moment and the control parameters of the RobCutt-II is a 3-D to 5-D mapping. To our knowledge, there is currently no general expression for the quantitative analysis of the undulatory fin propulsion that owns unique hydrodynamic characteristics. Consequently, we still use the fuzzy logic model proposed in [29] to build this nonlinear relationship. Specifically, the fuzzy method consists of fuzzification, fuzzy rule base, fuzzy inference, and defuzzification. The universe of discourse of each variable is determined according to the evaluation of previous motion control experiments. The standard triangular membership functions are utilized in the fuzzification. Then, we built the fuzzy rule base of the propulsive force/torque and the actuation parameters according to the evaluation of the experimental results of basic motion control. We remark that the fuzzy rules are generated on the basis of following four key principles.

1) The priority control sequence is heading control > depth control > surge control.
2) Heave force is closely related to the deflection angle θ_B of the long fin. When the heave force is larger, the deflection angle needs to be set larger.
3) When required surge force and yaw torque are both small, the phase difference between two adjacent fin rays becomes small, while the frequency of bilateral fins, the amplitudes of waves and the deflection angles are determined by the heave force.
4) When required surge force is small while yaw torque is large, the directions of traveling waves are opposite.
5) When required surge force is large while yaw torque is small, the directions of traveling waves are consistent, and the required force and torque are output by adjusting the undulatory frequency.

Mamdani's Max–Min method is used for fuzzy inference and the weighted average method is applied for defuzzification. Thus this fuzzy model could be used for control allocation.

Next, more details on the TD and the hovering controller will be described.

C. TD-Based Velocity Estimation

The velocity feedbacks are indispensable for closed-loop control, but difficult to measure practically. It is inevitable that the target position obtained by image processing is contaminated by noise. The velocity estimation implemented by direct derivative operations will amplify the noise. Fortunately, TD is effective in resolving noise amplification problems [30]. For a given setpoint v, a descrete-time TD that tracks the setpoint and gives the differential signal can be written as

$$\begin{cases} u = \mathrm{fhan}(v_1 - v, v_2, r_0, h_0) \\ v_1 = v_1 + h v_2 \\ v_2 = v_2 + h u \end{cases} \tag{4}$$

where h is the sampling period, r_0 is the speed factor related to the transient profile, h_0 denotes the filter factor related to the suppression of noise, $\mathrm{fhan}(\cdot)$ is the optimal control synthesis function that guarantees the fastest convergence from v_1 to v without any overshoot, v_2 is the derivative of v_1.

Algorithm 1: The Estimation of the Angular Velocity in Yaw.

```
Initialization
k = 1;
repeat
    Get the target position feedback data η_t;
    Calculate the yaw angle ψ_t(k) = atan2(y_t, x_t);
    u = fhan(ψ_1(k) − ψ_t(k), ψ_2(k), r_ψ, h_ψ0);
    ψ_1(k+1) = ψ_1(k) + h_ψ ψ_2(k);
    ψ_2(k+1) = ψ_2(k) + h_ψ u;
    Output the yaw rate estimation ψ̄_t(k) = −ψ_2(k+1);
until stop velocity estimation.
```

Take the yaw rate as an example, the algorithm for estimation of the angular velocity in yaw of the RobCutt-II is described in Algorithm 1.

In particular, k denotes the kth sample period, $\eta_t = (x_t, y_t, z_t)$ is the target position in the body-fixed frame obtained by image processing, $\mathrm{atan2}(Y, X)$ is the four quadrant arctangent of the elements of X and Y, y_t and x_t are the y-axis and x-axis coordinates of η_t, respectively, $r_{\psi 0}$ and $h_{\psi 0}$ are adjustable coefficients. We first calculate the yaw angle $\psi_t(k)$ of the target on the $O_B X_B Y_B$ plane. Second, a TD is employed to output ψ_1, ψ_2, which are the tracking signal of ψ_t and the differential signal of ψ_1, respectively. Hence, ψ_2 is exactly the yaw rate of the target relative to the RobCutt-II. Assume that the underwater target is stationary, so the angular velocity in yaw of the RobCutt-II is $\bar{\psi}_t(k) = -\psi_2(k+1)$. Analogously, another two noise-tolerant TDs are implemented for the estimations of the linear velocities in surge and heave.

D. Hovering Controller

Hovering controller design can be divided into two problems. One problem is the regulation of the centroid of the underwater target towards the reference visual feature, and the other one is to maintain the RobCutt-II at a fixed distance from the underwater target.

As for the first problem, the control objective is defined as the regulation to zero of a visual feature error function $e_s = s_n - s$, where $s = (x_g, y_g)^T$ is the centroid coordinate of the target in the image captured by the camera (the right camera of the binocular vision), s_n is the reference visual feature. The time derivative of s and the relative motion between the camera and

the target can be related by the image Jacobian [31]

$$\dot{s} = J_{\text{cam}} v_{\text{cam}} \tag{5}$$

with

$$J_{\text{cam}} = \begin{bmatrix} -1/Z_g & 0 & x_g/Z_g & x_g y_g & -(1+x_g^2) & y_g \\ 0 & -1/Z_g & y_g/Z_g & -(1+y_g^2) & -x_g y_g & -x_g \end{bmatrix} \tag{6}$$

where J_{cam} is the image Jacobian, Z_g is the depth between the camera and the target, and $v_{\text{cam}} = [v_x, v_y, v_z, w_x, w_y, w_z]^T$ is the camera velocity expressed in the camera frame.

The relationship between the vehicle velocity and the camera velocity can be given by the velocity Jacobian

$$v_{\text{cam}} = J_{\text{rob}} v_{\text{rob}} \tag{7}$$

where v_{rob} represents the 6×1 body-fixed velocity and J_{rob} is defined as the velocity Jacobian, which is a function of the camera position and attitude in the body-fixed frame.

For the RobCutt-II, the linear velocities in surge, heave and the angular velocity in yaw are controllable from bilateral undulatory fins. Therefore, the relationship between the controllable velocity and the camera velocity can be expressed as

$$v_{\text{cam}} = \bar{J}_{\text{rob}} \bar{v}_{\text{rob}} \tag{8}$$

where $\bar{J}_{\text{rob}}$ is the 6×3 matrix composed of 1st, 3rd, and 6th columns of J_{rob}, $\bar{v}_{\text{rob}} = [u, w, r]^T$ is the controllable velocity of the RobCutt-II. By substituting (8) into (5), the differential equations of visual feature can be rewritten as

$$\dot{s} = J_{\text{cam}} \bar{J}_{\text{rob}} \bar{v}_{\text{rob}}. \tag{9}$$

An exponential decrease of the error function is obtained by imposing $\dot{e}_s = -\lambda_s e_s$, with λ_s a positive constant. So we can solve for the control law of controllable velocity of the RobCutt-II necessary to guarantee the convergence of the image error function

$$\bar{v}_s^* = \lambda_s \left(J_{\text{cam}} \bar{J}_{\text{rob}}\right)^+ e_s \tag{10}$$

where $(\cdot)^+$ is the pseudo inverse of the matrix. Moreover, a PID control action on the visual feature for dynamic compensation is included

$$\bar{v}_s^* = \lambda_s \left(J_{\text{cam}} \bar{J}_{\text{rob}}\right)^+ \left(K_{ps} e_s + K_{is} \int e_s + K_{ds} \dot{e}_s\right) \tag{11}$$

where K_{ps}, K_{is}, K_{ds} are 2×2 adjustable parameters matrices.

When $|e_s|$ converges to a certain threshold, it may be considered that the Z_{RC} axis of the right camera is aimed at the underwater target. Then, the distance control is added. The reference distance is denoted by d_n. And the distance error is defined by $e_d = d_n - d$ with d the actual distance, which can be computed by 3-D reconstruction. When $e_d < 0$, the camera velocity in Z_{RC} axis v_z should be positive to increase e_d. While when $e_d > 0$, $v_z < 0$. We use another PID controller to control it

$$v_z = -\left(k_{pd} e_d + k_{id} \int e_d + k_{dd} \dot{e}_d\right) \tag{12}$$

Algorithm 2: Hovering Controller Algorithm.

```
Initialization
λ(0) = 0, k = 1;
repeat
    Compute visual error e_s(k) and distance error e_d(k);
    Compute image Jacobian J_cam;
    v̄*_s = λ_s (J_cam J̄_rob)^+ (K_ps e_s + K_is ∫ e_s + K_ds ė_s);
    v_z = −(k_pd e_d + k_id ∫ e_d + k_dd ė_d);
    v̄*_d = overline(J_rob^{-1}) v_cam;
    if (λ(k − 1) = 0) then
        if (e_s(k − 1) ≥ e_LT & e_s(k) ≤ e_LT) then
            λ(k) = 1;
        end if
    else if (λ(k − 1) = 1) then
        if (e_s(k − 1) ≤ e_UT & e_s(k) ≤ e_UT) then
            λ(k) = 0;
        end if
    end if
Compute control law v̄* = v̄*_s + λ(k) v̄*_d;
i++;
until stop hovering control.
```

where k_{pd}, k_{id}, and k_{dd} are proportional gain, integral gain, and derivative gain, respectively. Therefore, choosing $v_{\text{cam}} = [0, 0, v_z, 0, 0, 0]^T$, and the control law

$$\bar{v}_d^* = \overline{J_{\text{rob}}^{-1}} v_{\text{cam}} \tag{13}$$

makes the d converges to d_n. $\overline{J_{\text{rob}}^{-1}}$ is the 3×6 matrix composed of 1st, 3rd, and 6th rows of J_{rob}^{-1}. Finally, the overall hovering controller design is summarized in Algorithm 2.

Specifically, $\lambda \in \{0, 1\}$ represents the startup factor of distance control, when $\lambda = 1$, the distance control is added, otherwise when $\lambda = 0$, we turn OFF the distance control. k denotes the kth sample period. We first calculate current visual error $e_s(k)$ and distance error $e_d(k)$ based on reference inputs and feedbacks. Then, the target position in the $O_{RC} X_{RC} Y_{RC} Z_{RC}$ frame is updated by 3-D reconstruction. Thus, the image Jacobian J_{cam} can be calculated. Furthermore, the control actions $\bar{v}_s^*$ for the convergence of e_s and $\bar{v}_d^*$ for the convergence of e_d can be derived. The controller chooses whether to enable the distance control according to the value of e_s. Notice that in order to avoid oscillation of e_s near the threshold, two thresholds are employed to achieve switching control. Specially, e_{UT}, e_{LT} are, respectively, the upper threshold and the lower threshold. At last, the control output of the hovering controller is given by

$$\bar{v}^* = \bar{v}_s^* + \lambda(k) \bar{v}_d^*. \tag{14}$$

IV. Simulations and Experimental Results

A. Simulations

Simulations were implemented to validate the performance of the proposed hovering control method. We used the 4 DOF mathematical model given in [29] to represent the RobCutt-II's dynamics. Note that the control inputs of the RobCutt-II in the

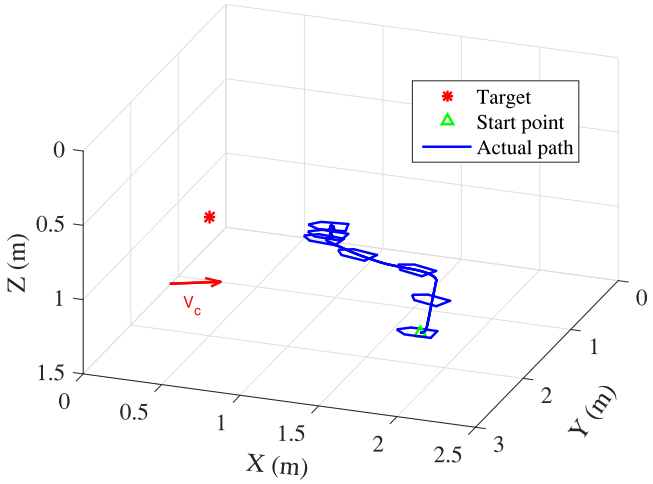


Fig. 6. 3-D trajectory of the RobCutt-II in simulation.

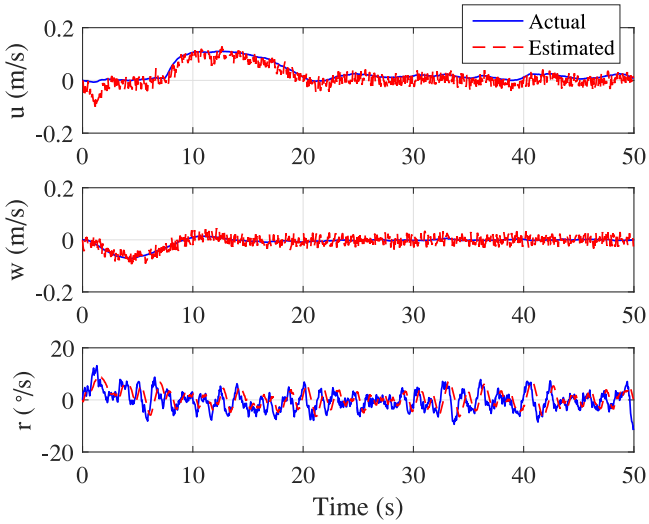


Fig. 7. Time evolution of the actual body-fixed velocities and the estimated ones based on TD in simulation.

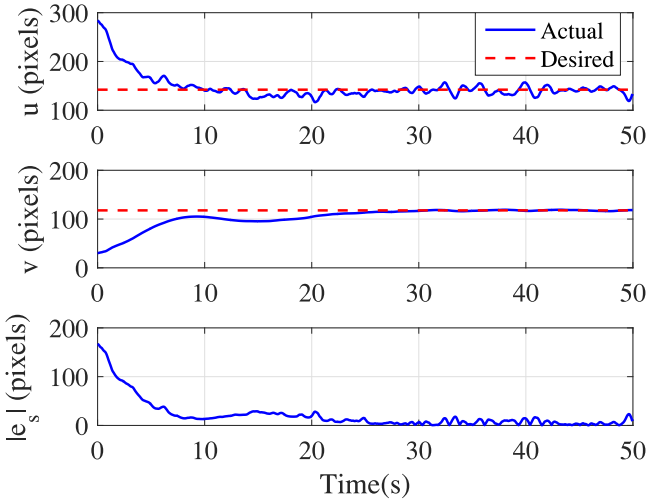


Fig. 8. Time evolution of the pixel coordinates in image plane and the pixel error in simulation.

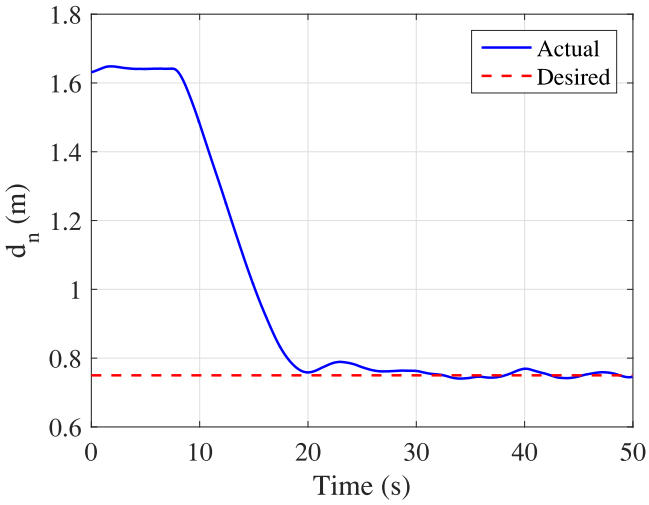


Fig. 9. Time evolution of the distance from the barycenter of the RobCutt-II to the underwater target in simulation.

simulation were propulsive force in surge and heave and moment in yaw. The calibration of the parameters of the binocular vision system and the eye to hand calibration had been prematurely carried out in water. In the simulation, the relationships between various frames were set to be consistent with the real situation.

For the convenience of description, we set a world frame $O_W X_W Y_W Z_W$ with the origin O_W at the upper left corner of the pool. The water surface was served as the plane formed by X_E axis and Y_E axis of the. Z_E axis was perpendicular to water surface. The target position was $(0.5\,, 2\,, 0.7$ m) and the RobCutt-II started with initial position $(2\,, 2.5, 1.1$ m) in $O_W X_W Y_W Z_W$ frame. The initial heading of the RobCutt-II was parallel to the X_W axis, while the initial velocity was zero. The reference visual feature was $(0, 0)$, which corresponds to the pixel coordinates as $(142.15, 117.79)$. The reference distance was preset as 0.75 m.

To validate the disturbance rejection performance of the control method, we introduced two external disturbances to the dynamic model of the RobCutt-II. For one thing, zero mean uniform random noises with amplitudes 1 N or 1 N·m were incorporated into the surge, sway, heave, and yaw dynamics of the RobCutt-II. In addition, a constant flow turbulence, which is unknown from the point of view of the controller, with intensity $V_c = 0.015$ m/s was also added to the vehicle's dynamic model. Meanwhile, considering that the position measurement errors of the binocular vision system are at the millimeter level, we added uniformly distributed random measurement errors in the range of $[-0.01\,\text{m}, 0.01\,\text{m}]$ to the actual 3-D coordinates of the underwater target.

The simulation results based on these conditions are shown in Figs. 6–9. Fig. 6 shows the actual trajectory of the RobCutt-II. It is observed that the proposed hovering control was able to force the RobCutt-II to hover in front of the target. Because of the effect of the constant flow turbulence, there was a certain amount of position drift when hovering, but the RobCutt-II always headed toward the target. Fig. 7 shows the time evolution of the actual body-fixed velocity and the estimated one output by the TD. Although estimated velocity

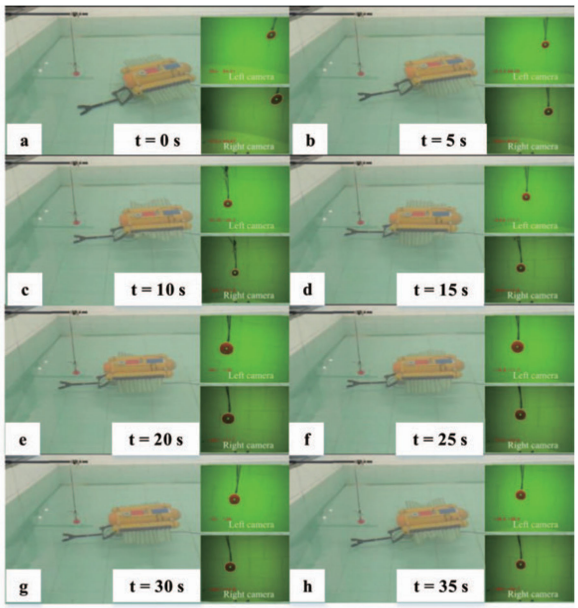


Fig. 10. Image sequences of the RobCutt-II in hovering experiment.

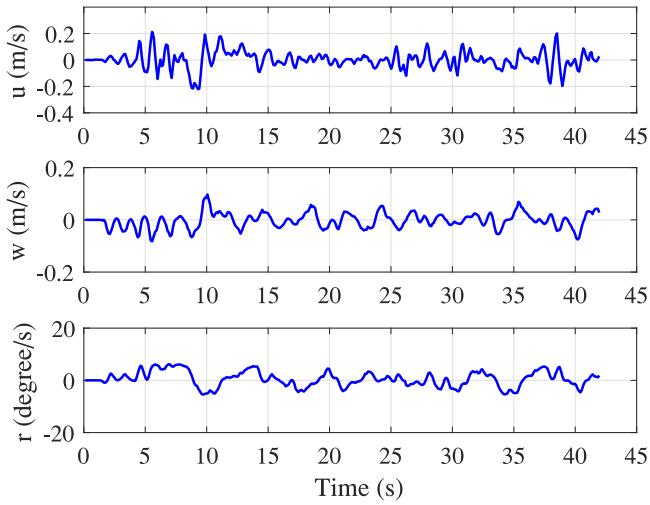


Fig. 11. Time evolution of the estimated body-fixed velocities in hovering experiment.

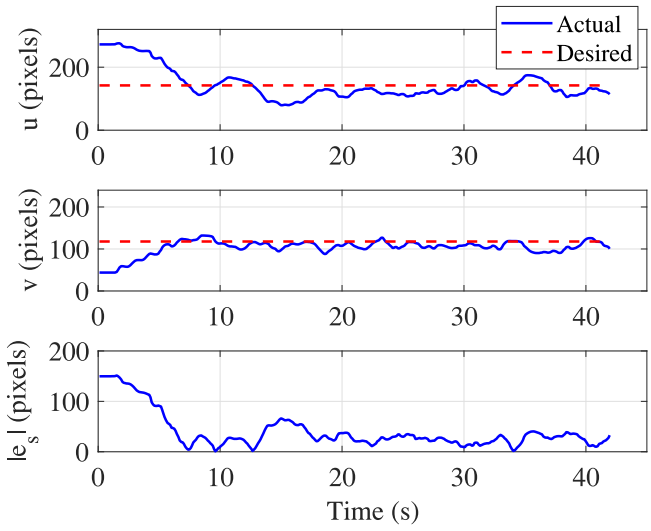


Fig. 12. Time evolution of the pixel coordinates in image plane and the pixel error in hovering experiment.

has some jitter due to large random measurement errors, it well follows the actual velocity, which verifies the validity of TD in velocity estimation. Fig. 8 shows the time evolution of the visual feature. Intuitively, we present pixel coordinates instead of visual feature. Fig. 9 shows the time evolution of the distance between the RobCutt-II and the underwater target. In Figs. 8 and 9, the convergence of the pixel error and the distance error states that autonomous hovering was achieved.

B. Hovering Experiment

In order to further evaluate the performance of the proposed control system of the RobCutt-II, hovering experiment on the RobCutt-II prototype was performed in an indoor pool with dimensions of $5 \times 4 \times 1.1$ m (length $\times$ width $\times$ depth).

We hanged a red ball with a bamboo bar and immersed it in water to simulate the underwater target. The reference visual feature and distance were the same as those of the simulation. A pressure sensor is used to get real-time depth of the RobCutt-II. Initially, the RobCutt-II was at a certain distance in front of the target, when the pixel coordinate of the target in the image captured by the right camera was $(272.36, 44.03)$ and the initial distance was 0.97 m. Note that binocular vision positioning was far less accurate when the RobCutt-II was at a longer distance from the underwater target, so the initial distance was closer than that of the simulation. The improvement in underwater visual environment perception is a subject for future work. The initial swimming velocity was zero.

The image sequences captured by an external camera and the binocular vision system during hovering control experiment are shown in Fig. 10, where Fig. 10(a)–(c) shows the process of the adjustment of the heading and depth, Fig. 10(c)–(f) shows the process of approaching the target, and Fig. 10(f)–(h) shows the process of stable hovering in front of the target. Fig. 11 shows the time evolution of the estimated body-fixed velocities based on TD. Fig. 12 shows the time evolution of the pixel coordinates and the pixel error. Fig. 13 shows the time evolution of the distance from the barycenter of the RobCutt-II to the underwater target. At 0–10 s, the RobCutt-II had not started the distance control, when it responded sensitively to reduce the visual feature error by adjusting the heading and the depth. After the visual feature error was less than the threshold, the distance control was started and the distance between the RobCutt-II and the target gradually decreased. Since there was small swing of the attitude during the process of approaching the target, the visual feature error increased to some extent. When the RobCutt-II reached the desired distance, the visual feature gradually converged to the reference value. The final steady-state pixel error remained within 40 pixels, and the distance error is within 0.08 m. Fig. 14 shows the time evolution of the control parameters of the RobCutt-II. Under the hovering control, the

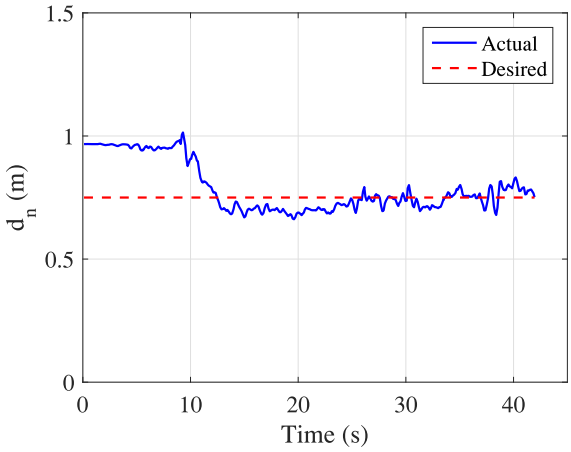


Fig. 13. Time evolution of the distance from the barycenter of the RobCutt-II to the underwater target in hovering experiment.

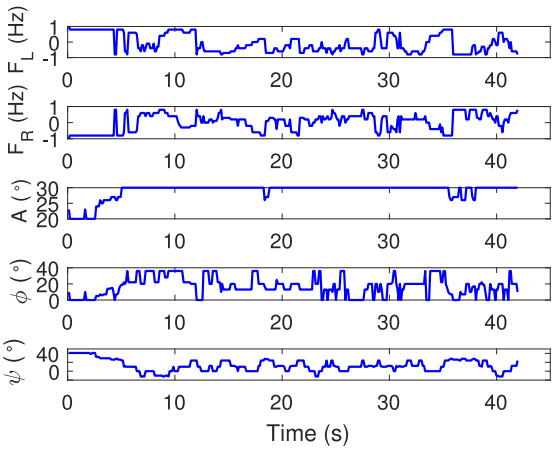


Fig. 14. Time evolution of the control parameters of traveling waves on bilateral fins in hovering experiment.

control parameters of traveling waves on bilateral fins changed in real time, and the autonomous hovering of the RobCutt-II was achieved.

C. Comparative Study

Under the same conditions, we conduct some comparative experiments including the velocity direction transform matrix (VDTM) method in [26] and the direct derivative velocity estimation (DDVS) method. In particular, the VDTM method calculates the desired controllable velocity using a PID controller and the VDTM. Note that because the neural network requires a large amount of input and output data to train the network, we replaced the neural network-based control allocation module in [26] with the control allocation module based on fuzzy logic model. Meanwhile, the velocity feedbacks were still obtained by the TD-based velocity estimation instead of an inertial navigation system. Meanwhile, compared with the method proposed in this paper, the DDVS method calculates the controllable velocity vector by direct derivative operations.

The results are shown in Figs. 15 and 16. Fig. 15 describes the time evolution of the pixel coordinates and the pixel errors

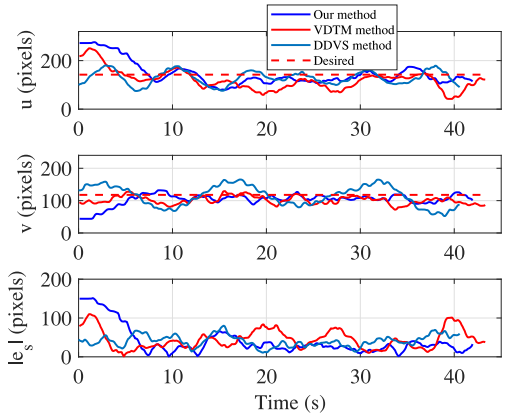


Fig. 15. Comparative experimental results of the pixel coordinates in image plane and the pixel errors.

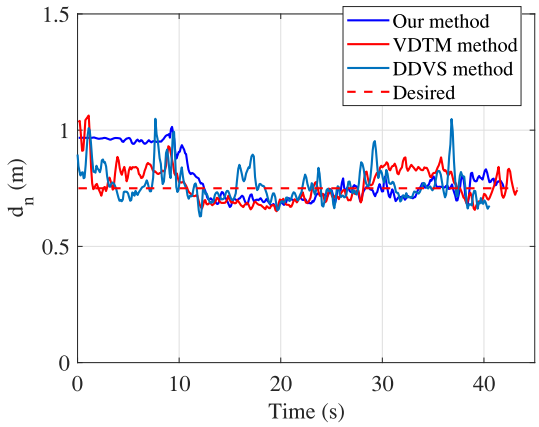


Fig. 16. Comparative experimental results of the distance from the barycenter of the RobCutt-II to the underwater target.

of these three approaches. The time evolution of the distance from the barycenter of the RobCutt-II to the underwater target is shown in Fig. 16. It is observed from the figures that the pixel error and distance error of the proposed method are better than those of the VDTM and DDVS method. The VDTM method does not include switching control, which means that the RobCutt-II always needs to adjust its heading, depth, and distance to the target at the same time. Specifically, the pixel errors of the VDTM method and the DDVS method after the RobCutt-II tends to stabilize are about 100 pixels and about 70 pixels, and the maximum distance errors are 0.133 and 0.298 m, respectively. As mentioned above, the steady-state pixel error of the proposed method is within 40 pixels, while the distance error is within 0.08 m. In addition, the DDVS method has bigger jitter than the TD-based method. It can be concluded that the proposed hovering control method is more effective and accurate.

D. Antidisturbance Experiment

To ensure a stable hovering performance, an antidisturbance experiment was chosen as the test condition. In the experiment, the RobCutt-II was first required to hover in front of the target. Then some disturbances were deliberately added to the RobCutt-II. Specifically, at 10 s, we made some water

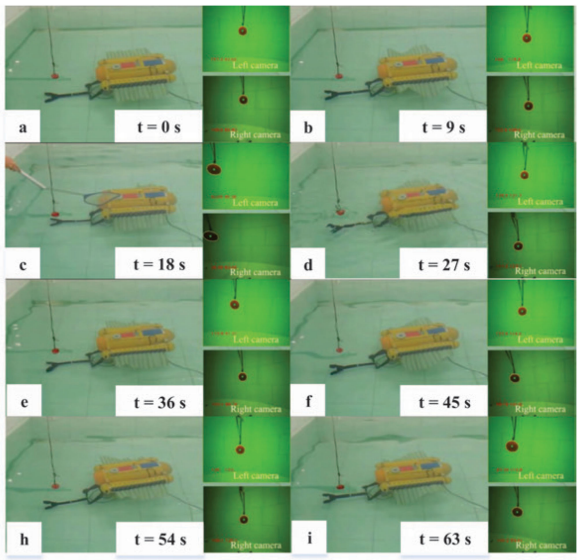


Fig. 17. Image sequences of the RobCutt-II in antidisturbance experiment.

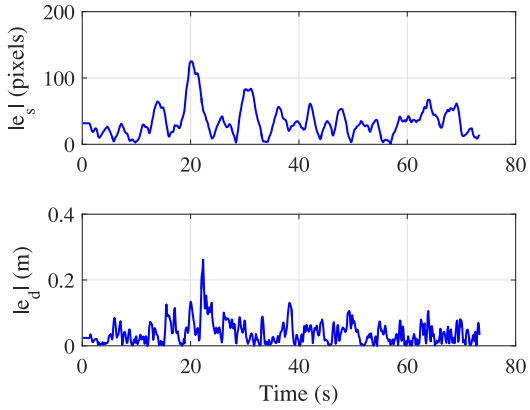


Fig. 18. Time evolution of the pixel error and the distance error in antidisturbance experiment.

turbulence and pushed the head of the RobCutt-II by a pool leaf skimmer with telescopic pole.

Fig. 17 gives the image sequences of the RobCutt-II during anti-interference experiment. Fig. 18 describes the time evolution of the pixel error and the distance error. It can be observed from these figures that the pixel error increased to 125 pixels and the distance error increased to about 0.3 m due to external disturbances. Thanks to the proposed hovering control method, the RobCutt-II resumed hovering in front of the target and the underwater target was always in the cameras' view. Because there were still some disturbances of water turbulence, the pixel error was larger than that of the previous hovering experiment, but it was still within 70 pixels. While the maximum distance error is only 0.11 m. The experimental results verify the antidisturbance performance of the proposed hovering control method.

At this point, the vision-based hovering control for the RobCutt-II is validated. To the best of our knowledge, the closed-loop control for autonomous hovering of a biomimetic underwater robot propelled by undulatory fins is realized for the first time.

V. Conclusion

The effectiveness of a vision-based hovering control was experimentally investigated on a biomimetic underwater robot called RobCutt-II. A biomimetic underwater propulsor with an undulatory fin was developed for mimicry of cuttlefish-like propulsion. Furthermore, with the coordinated motion control of dual undulatory fins, the RobCutt-II can accomplish multimodal motion with high maneuverability and stability. The dual-loop hovering control framework consisted of an image processing module, a TD, a hovering controller, a velocity controller, and a fuzzy-based control allocation module. The experimental results showed that the RobCutt-II achieved autonomous hovering under external disturbances.

Despite the successful implementation of autonomous hovering of the RobCutt-II, a more robust control is needed for practical applications. In our future work, closed-loop control of some moveable sliders will be applied to the underwater robot to stabilize its attitude during motion of dual fins. Some robust control methods, such as H-infinity loop-shaping, sliding mode control, will be used to get better control performance. The improvement in environment sensing based on binocular camera in outdoor environments is another ongoing effort.

References

[1] B. Sun, D. Zhu, and S. Yang, "A bioinspired filtered backstepping tracking control of 7000-m manned submarine vehicle," *IEEE Trans. Ind. Electron.*, vol. 61, no. 7, pp. 3682–3693, Jul. 2014.

[2] H. Li, P. Xie, and W. Yan, "Receding horizon formation tracking control of constrained underactuated autonomous underwater vehicles," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5004–5013, Jun. 2017.

[3] M. Maciver, E. Fontaine, and J. Burdick, "Designing future underwater vehicles: Principles and mechanisms of the weakly electric fish," *IEEE J. Ocean. Eng.*, vol. 29, no. 3, pp. 651–659, Jul. 2004.

[4] T. Shimura, Y. Amitani, T. Sawa, and Y. Watanabe, "A basic research on the improvement of propulsion maneuvering system and the automatic motion control of SHINKAI6500," in *Proc. MTS/IEEE OCEANS*, Oct. 2002, pp. 1332–1338.

[5] T. Koh, M. Lau, G. Seet, and E. Low, "A control module scheme for an underactuated underwater robotic vehicle," *J. Intell. Robot. Syst.*, vol. 46, pp. 43–58, 2006.

[6] N. Leonard, "Control synthesis and adaptation for an underactuated autonomous underwater vehicle," *IEEE J. Ocean. Eng.*, vol. 20, no. 3, pp. 211–220, Jul. 1995.

[7] A. Aguiar and A. Pascoal, "Dynamic positioning and way-point tracking of underactuated AUVs in the presence of ocean currents," *Int. J. Control*, vol. 80, no. 7, pp. 1092–1108, 2007.

[8] S. Zhao and J. Yuh, "Experimental study on advanced underwater robot control," *IEEE Trans. Robot.*, vol. 21, no. 4, pp. 695–703, Aug. 2005.

[9] W. Stone *et al.*, "Design and deployment of a four-degrees-of-freedom hovering autonomous underwater vehicle for sub-ice exploration and mapping," *Proc. Inst. Mech. Engineers, Part M: J. Eng. for Maritime Environ.*, vol. 224, no. 4, pp. 341–361, 2010.

[10] M. Santhakumar and T. Asokan, "Power efficient dynamic station keeping control of a flat-fish type autonomous underwater vehicle through design modifications of thruster configuration," *Ocean Eng.*, vol. 58, pp. 11–21, 2013.

[11] S. Jin, J. Kim, J. Kim, and T. Seo, "Six-degree-of-freedom hovering control of an underwater robotic platform with four tilting thrusters via selective switching control," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 5, pp. 2370–2378, Oct. 2015.

[12] A. Loria, T. Fossen, and E. Panteley, "A separation principle for dynamic positioning of ships: Theoretical and experimental results," *IEEE Trans. Control Syst. Technol.*, vol. 8, no. 2, pp. 332–343, Mar. 2000.

[13] A. Sorensen, "A survey of dynamic positioning control systems," *Annu. Rev. Control*, vol. 35, no. 1, pp. 123–136, 2011.

[14] H. Katayama, "Nonlinear sampled-data stabilization of dynamically positioned ships," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 2, pp. 463–468, Mar. 2010.

[15] E. Tannuri and H. Morishita, "Experimental and numerical evaluation of a typical dynamic positioning system," *Appl. Ocean Res.*, vol. 28, pp. 133–146, 2006.
[16] J. Yu, F. Sun, D. Xu, and M. Tan, "Embedded vision-guided 3-D tracking control for robotic fish," *IEEE Trans. Ind. Electron.*, vol. 63, no. 1, pp. 355–363, Jan. 2016.
[17] J. Wang and H. Cho, "Micropeg and hole alignment using image moments based visual servoing method," *IEEE Trans. Ind. Electron.*, vol. 55, no. 3, pp. 1286–1294, Mar. 2008.
[18] L. Deng, F. Janabi-Sharifi, and W. Wilson, "Hybrid motion control and planning strategies for visual servoing," *IEEE Trans. Ind. Electron.*, vol. 52, no. 4, pp. 1024–1040, Aug. 2005.
[19] Z. Cao, X. Chen, Y. Yu, J. Yu, X. Liu, and M. Tan, "Image dynamics-based visual servoing for quadrotors tracking a target with a nonlinear trajectory observer," *IEEE Trans. Syst., Man, Cybern.: Syst.*, to be published, doi: 10.1109/TSMC.2017.2720173.
[20] F. Lin, X. Dong, B. Chen, K. Lum, and T. Lee, "A robust real-time embedded vision system on an unmanned rotorcraft for ground target following," *IEEE Trans. Ind. Electron.*, vol. 59, no. 2, pp. 1038–1049, Feb. 2012.
[21] R. Marks, H. Wang, M Lee, and S. Rock, "Automatic visual station keeping of an underwater robot," in *Proc. MTS/IEEE OCEANS*, Sep. 1994, pp. II/137– II/142.
[22] J. Lots, D. Lane, E. Trucco, and F. Chaumette, "A 2-D visual servoing for underwater vehicle station keeping," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2001, pp. 2767–2772.
[23] S. Negahdaripour, X. Xu, and L. Jin, "Direct estimation of motion from sea floor images for automatic station-keeping of submersible platforms," *IEEE J. Ocean. Eng.*, vol. 24, no. 3, pp. 370–382, Jul. 1999.
[24] S. Zwaan and J. Santos-Victor, "Real-time vision-based station keeping for underwater robots," in *Proc. MTS/IEEE OCEANS*, Nov. 2001, pp. 1058–1065.
[25] J. Gao, A. Proctor, and C. Bradley, "Adaptive neural network visual servo control for dynamic positioning of underwater vehicles," *Neurocomputing*, vol. 167, pp. 604–613, 2015.
[26] R. Wang, Y. Wang, S. Wang, C. Tang, and M. Tan, "Visual servo control for dynamic hovering of an underwater biomimetic vehicle-manipulator system by neural network," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, 2017, pp. 1168–1173.
[27] D. Xu, M. Tan, and Y. Li, *Visual Measurement and Control for Robots*. Beijing, China: National Defense Industry Press, 2016.
[28] Y. Wang, S. Wang, Q. Wei, M. Tan, C. Zhou, and J. Yu, "Development of an underwater manipulator and its free-floating autonomous operation," *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 2, pp. 815–824, Apr. 2016.
[29] S. Wang, Y. Wang, Q. Wei, M. Tan, and J. Yu, "A bio-inspired robot with undulatory fins and its control methods," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 1, pp. 206–216, Feb. 2017.
[30] J. Han, "From PID to active disturbance rejection control," *IEEE Trans. Ind Electron.*, vol. 56, no. 3, pp. 900–906, Mar. 2009.
[31] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. Robot. Autom.*, vol. 8, no. 3, pp. 313–326, Jun. 1992.

Rui Wang received the B.E. degree in automation from the Beijing Institute of Technology, Beijing, China, in 2013, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences (IACAS), Beijing, in 2018.

He is currently an Assistant Professor with the State Key Laboratory of Management and Control for Complex Systems, IACAS. His research interests include intelligent control, robotics, underwater robots, and biomimetic robots. He has authored over 20 refereed journal and conference papers, e.g., the IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS: SYSTEMS, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, IEEE JOURNAL OF OCEANIC ENGINEERING, the IEEE International Conference on Robotics and Automation (ICRA), the IEEE International Conference on Automation Science and Engineering (CASE), and the International Ocean and Polar Engineering Conference (ISOPE).

Shuo Wang received the B.E. degree in electrical engineering from Shenyang Architecture and Civil Engineering Institute, Shenyang, China, in 1995, the M.E. degree in industrial automation from Northeastern University, Shenyang, China, in 1998, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2001.

He is currently a Professor with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interests include biomimetic robot, underwater robot, and multirobot systems.

Yu Wang received the B.E. degree in automation from the Beijing Institute of Technology, Beijing, China, in 2011, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2016.

He is currently an Associate Professor with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interests include intelligent control, robotics, and biomimetic robots.

Mingxue Cai received the B.E. degree in mechanical and electronic engineering from the School of Mechanical Engineering, University of Jinan, China, in 2014, and the M.E. degree in mechanical and electronic engineering from the School of Mechanical Engineering and Automation, Beihang University, Beijing, China, in 2017. He is currently working toward the Ph.D. degree in control theory and control engineering at the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing.

He is with the University of Chinese Academy of Sciences, Beijing. His research interests include intelligent control, biomimetic robots, and autonomous underwater manipulation.

Min Tan received the B.E. degree in automation from Tsinghua University, Beijing, China, in 1986, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, in 1990.

He is currently a Professor with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interests include advanced robot control, biomimetic robot, and multirobot system.