2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Real-Time Estimation for the Swimming Direction of Robotic Fish Based on IMU Sensors*

Shikun Li[1], Yufan Zhai[1], Chen Wang[2] and Guangming Xie[1,3]

Abstract— **An increasing number of underwater robots inspired by Carangidae are developed, which is characterized by high efficiency and flexibility. However, estimating the swimming direction of these robotic fish is challenging due to the constant swinging of the head during movement, which complicates precise control. In this study, we installed two low-cost inertial measurement unit (IMU) sensors separately on the head and tail parts of a double-joint robotic fish and presented a method for accurately and timely estimating the swimming direction. Firstly, we effectively compensated for the yaw angle drift of the IMU sensors through a fused Kalman Filter. Furthermore, we propose the Anti-Shake Estimation (ASE) algorithm to calculate the real-time swimming direction using filtered yaw angles at a high updating rate of 100Hz. Finally, we applied the method to swimming direction feedback control for evaluation and comparison. The results show that our ASE method performs better than other existing methods in straight-line swimming experiments. The experiment of S-curve swimming also demonstrates the effectiveness of our method in complex missions.**

I. Introduction

Due to the complexity of the underwater environment, the exploitation of marine resources is more difficult than that on land. The development of underwater robots, particularly bio-inspired robotic fish, has provided new methods for ocean exploration. Fish have lived underwater for extended periods and have adapted remarkably well to this environment. They can swim rapidly and flexibly [1], which aids them in evading predators [2] and navigating through complex underwater terrain [3]. Inspired by the excellent performance of real fish, various types of robotic fish have been developed for diverse underwater missions and applications, including the collection of water information [4], object detection [5] and swarm exploration [6].

The excellent performance of robotic fish underwater relies on precise motion control. However, the flow field in underwater environments is often complex, which adds uncertainty to the movement of robotic fish. Moreover, the fish bends its body to provide forward thrust [7], implying that the heading direction is not exactly the swimming direction. When real fish swim, the center of mass oscillates axially and laterally [8] and the head oscillates around the center of mass during each swinging period [9]. It is reasonable to assume that robotic fish exhibit a similar difference between swimming and heading directions as observed in real fish. In addition, the robotic fish have inferior propulsive performance and efficiency compared to the real fish, which leads to more intense head shaking and increases the difficulty of real-time estimation of swimming direction [10]. Thus, to control the robotic fish underwater more accurately and stably, the precise estimation of swimming direction in real-time needs to be solved first.

Several sensors have been employed to calculate the swimming direction of robotic fish, such as IMU sensor [11], artificial lateral line system [12], [13], visual camera [14] or combinations of different kinds of sensors [15], [16]. However, these sensors face challenges in directly measuring real-time swimming direction due to the inherent swinging motion of the fish. Additionally, despite efforts in material and design optimization [17], it remains challenging to completely eliminate head shaking. The six-axis IMU is an electronic device that combines accelerometer and gyroscope to directly measure angular information in three-dimensional space. However, in existing methods, based on the yaw angle measured by IMU, the swimming direction of robotic fish is usually estimated with delay or low updating rates because of the head shaking. For example, some methods report that the swimming direction is estimated by calculating the average yaw angle of the past periods (MLP method), which causes the estimation delay [11], [18]. Besides, the instantaneous yaw angle at peak angular velocities (PAV method) is regarded as the actual swimming direction but is updated at a low frequency (once every half period) [19], [20]. Furthermore, certain techniques have been developed to reduce the head shaking of robotic fish and improve swimming stability [20]. Nevertheless, in several frameworks of trajectory control for robotic fish [21], [22], the swimming direction estimated through the above methods is frequently utilized. Consequently, errors in swimming direction estimation can cause inaccurate results of control and failure of navigation tasks.

In this paper, in order to improve the effectiveness of control strategies, we designed a new kind of double-joint robotic fish equipped with two IMU sensors—one positioned at the head and the other at the tail. We established a fused Kalman Filter to reduce the drift of IMU sensors in long-term measurement. Furthermore, we proposed the ASE algorithm

*This work was supported in part by the National Natural Science Foundation of China under Grants U22A2062, U23B2037, and 12272008. *(Corresponding author: Guangming Xie.)*

[1]Shikun Li, Yufan Zhai and Guangming Xie are with the State Key Laboratory for Turbulence and Complex Systems, Intelligent Biomimetic Design Lab, College of Engineering, Peking University, Beijing, 100871, China. lishikun@stu.pku.edu.cn; zhaiyufan@pku.edu.cn; xiegming@pku.edu.cn

[2]Chen Wang is with the National Engineering Research Center for Software Engineering, Peking University, Beijing 100871, China. wangchen@pku.edu.cn

[3]Guangming Xie is with the Institute of Ocean Research, Peking University, Beijing, 100871, China.

to calculate the real-time swimming direction using filtered yaw angles without delay at a high updating rate of 100Hz. The contributions of this paper are summarized as follows:

- Two simple and low-cost IMU sensors are installed on the double-joint robotic fish. By adopting a fused Kalman filter, the long-term yaw angle drifts of IMU sensors are reduced regardless of the external environment interference.
- A real-time estimation method is put forward to calculate the swimming direction of robotic fish based on the filtered yaw angles, despite that the robotic fish's head is continuously shaking. Our real-time estimation method can improve the control effect of robotic fish.

The remainder of this paper is organized as follows. The robotic fish prototype is briefly introduced in Section II. The methods and algorithms for real-time estimation of swimming direction are presented in Section III. The experimental results are described in Section IV. Finally, the conclusion and future work are discussed in section V.

II. Robot System

The design of the robot is inspired by yellowfin tuna. As shown in Fig. 1, it comprises a streamlined head part, a tail part, a peduncle keel and a caudal fin. The size (L×W×H) of the robot is about 330 × 95 × 112 mm. Different from previous tuna-inspired robots [10], [23]–[25], a spring traverses the entire body of fish, creating a compliant structure. A waterproof servo is employed to drive the tail. Two six-axis IMU sensors (MPU6050) are horizontally placed on the head and tail parts. The data of sensors and robot state can be transmitted through a Nordic Semiconductor Radio Frequency (NRF) module. The main controller is a microcontroller unit (STM32F407), which is used for locomotion control, sensor data acquisition and data processing. The technical specifications of the robot are listed in Table I.

To mimic the excellent locomotion capability of real fish, we implement a Central Pattern Generator (CPG) network for the control of the robotic fish, which can adjust the amplitude, frequency and offset of the tail flexibly.

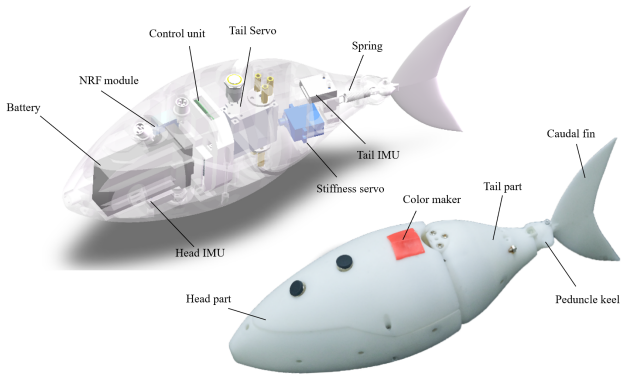

Fig. 1. The mechanical and electrical structures of robotic fish.

TABLE I
Technical Specifications of the Robot Prototype

Items	Characteristics
Dimension(L×W×H)	∼ 330×95×112 mm
Total mass	∼ 0.8kg
Power supply	7.4V rechargeable Ni-MH batteries
Driving mode	DC servomotor (30 kg·cm)
Operation time	∼ 5h
Control mode	Autonomous / Wireless mode

The CPG controller used here includes a liner oscillator [26].

$$r(t) = R - Re^{-\alpha t} \tag{1}$$

$$x(t) = X - Xe^{-\beta t} \tag{2}$$

$$\theta(t) = x(t) + r(t)sin(2\pi ft) \tag{3}$$

where $r(t)$ and $x(t)$ are the amplitude and offset of the oscillator respectively, which converge to R and X exponentially. The parameters α and β determine the convergence speed. θ is the output servo angle of oscillator, and f is the swinging frequency of the robotic fish. Based on CPG and the liner oscillator, the robotic fish can change smoothly to another motion state, which enhances the stability of the movement.

III. Methods and Algorithms

In this section, we analyzed the geometric angle relationships of robotic fish movement, which are the theoretical foundation for calculating the swimming direction. Besides, based on the two six-axis IMU sensors on the robotic fish, we developed a fused Kalman filter to reduce the yaw angle drift of IMU sensors. Finally, utilizing the filtered IMU yaw angles, we proposed the ASE algorithm to estimate the swimming direction of the robotic fish accurately in real-time.

A. Dynamic Analysis of the Robotic Fish

Consider a scenario shown in Fig. 2, where a robotic fish moves along a specified trajectory (the blue curve). The green dotted line represents the initial head direction and the yellow dotted line represents the current direction. Therefore, the swimming direction γ is defined as the angle of these two dotted lines. The amplified picture presents the current state and kinematic angles while the robotic fish is swimming. The solid fish and the black line represent the current position, which indicates the real-time posture of the robotic fish. The transparent fish and the azure line represent the reference state without swinging. In other words, if we decompose the motion of robotic fish into translation (movement where the head direction and the swimming direction are identical) and swinging, the azure transparent fish is the posture of pure translation. x is the given offset of the tail in CPG, which is considered as the position reference of tail swinging. α_1 and α_2 are the IMU yaw angles of the head and tail parts. β_1 and β_2 are angles relative to the reference position for the head and tail. θ is the servo angle. Counterclockwise is

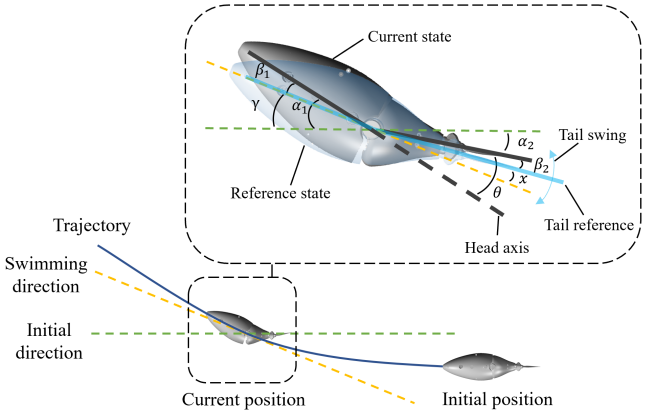

Fig. 2. Kinematic analysis of robotic fish movement from a top-down perspective.

defined as the positive direction when viewing the robotic fish from above. Thus,

$$\alpha_1 = \gamma + \beta_1 \tag{4a}$$

$$\alpha_2 = \gamma + \beta_2 + x \tag{4b}$$

$$\theta = \alpha_2 - \alpha_1 \tag{4c}$$

Since the overall angular momentum of the robotic fish remains conserved while it is swimming, a proportional relationship exists between the swinging angles of the head and tail parts, as expressed by the equation:

$$\beta_2 = K \times \beta_1 \tag{5}$$

where K is the proportional coefficient. Due to the nonlinear hydrodynamics property of robotic fish, K varies in different motion patterns. Since x and θ are the input and output of the CPG controller respectively, and α_1 and α_2 are the outputs of IMU sensors, the real-time swimming direction can be calculated by the following equation:

$$\gamma = \alpha_1 - \frac{\theta - x}{K - 1} \tag{6}$$

B. Two-IMU Fused Kalman Filter

Due to the inherent error of the gyroscope, the yaw angle measured by the six-axis IMU sensor has a drift as time goes on. Thus, we designed a two-IMU fused Kalman Filter to reduce the long-term drift and improve the accuracy of yaw angle measurement. As the accelerometer within the same IMU cannot effectively compensate for the gyroscope drift, we use the data from the other IMU and the servo to provide information about actual yaw angles. Specifically, the filter based on two six-axis IMU sensors can be written as

$$\tilde{\alpha_i}(k) = A\tilde{\alpha_i}(k-1) + Bu_i(k-1) + v_i(k-1), i = 1, 2 \tag{7}$$

$$\alpha_1(k) = H_1(\tilde{\alpha_2}(k) - \theta) + w_1(k) \tag{8a}$$

$$\alpha_2(k) = H_2(\tilde{\alpha_1}(k) + \theta) + w_2(k) \tag{8b}$$

where $\tilde{\alpha_i}(k)$ and $\alpha_i(k)$ are the estimated yaw angle and the measurement of yaw angle at time step k. u_i is the angular acceleration measured by the gyroscope. θ is the output angle of the servo. A, B, H_i are the identity matrices. $v_i(k)$ and $w_i(k)$ represent the process and measurement noise, which are independent white noise with normal distribution

$$v_i(k) \sim N(0, Q_i) \tag{9}$$

$$w_i(k) \sim N(0, R_i) \tag{10}$$

where Q_i and R_i are the process noise covariance and measurement noise covariance matrices respectively. Based on the mathematical models discussed above, the filtered yaw angles can be calculated using two classical Kalman filters in the subsequent steps. Further details will not be elaborated here.

Unlike the magnetometer which relies on Earth's magnetic field to provide absolute information, our method only uses the information from the robotic fish itself to reduce the yaw angle drift, which is not affected by external interference. In addition, our method does not need a calibration process like the magnetometer and is convenient to apply.

C. ASE-Algorithm

Although the yaw angle drift of IMU can be reduced through the above method, the measured yaw angle is not exactly the swimming direction. To calculate the swimming direction of robotic fish, we designed the ASE algorithm which leverages the filtered data from IMU sensors. The main purpose of the ASE algorithm is to predict the proportional coefficient K, which is the only unknown parameter in Equation 6, and then calculate the real swimming direction γ at each time step. K is estimated by comparing the swinging ranges of the head and tail parts. However, when calculating the swimming direction, the output value of the servo is not exactly the angle θ of two adjacent joints due to the mechanical oscillation and water resistance. Consequently, there is a phase difference $\Delta\phi$ between the yaw angles from IMU sensors. For a more accurate estimation of the swimming direction, we calculate the phase difference $\Delta\phi$ by aligning the curves of yaw angles and then obtain the actual angle θ precisely.

Considering that the robotic fish can change the motion parameters, K and $\Delta\phi$ are updated at every time step. Unlike existing methods that rely on historical IMU data to estimate swimming direction, our algorithm only uses the historical data to estimate K and $\Delta\phi$, which characterizes the swinging behavior of robotic fish and remains relatively stable. Other parameters in Equation 6 are all real-time values, which ensure the real-time property of the algorithm. Besides, the swimming direction can be estimated at a high frequency of 100 Hz, which is the same as the sampling frequency of IMU sensors. Furthermore, our method considers the uniqueness of maximum and minimum at each time step, in case of the local extreme values due to the mechanical oscillation or sensor noise. In summary, our ASE algorithm can estimate the swimming direction precisely and has great robustness.

Algorithm 1 Anti-Shake Estimation (ASE)

Initialize α_i, $\alpha_i^{\max}$, $\alpha_i^{\min}$, $\Delta\phi$, γ, Δx, K, n_2
for n=1:$+\infty$ **do**
  $\alpha_i^{\max} \leftarrow$ the maximum value of filtered α_i in the past period
  $\alpha_i^{\min} \leftarrow$ the minimum value of filtered α_i in the past period
  Find the phase difference $\Delta\phi$ between α_i
  Calculate the offset change Δx and update the amplitude proportion coefficient K:

$$K(n) \leftarrow \frac{\Delta x + \alpha_2^{\max} - \alpha_2^{\min}}{\alpha_1^{\max} - \alpha_1^{\min}}$$

  Align the curve of α_2 with α_1 according to $\Delta\phi$ and find the index n_2 of α_2 which corresponds to $\alpha_1(n)$
  Calculate the swimming direction:

$$\gamma(n) \leftarrow \alpha_1(n) - \frac{\alpha_1(n) - \alpha_2(n_2) - x(n)}{K(n) - 1}$$

end for

IV. Results

In this section, we first evaluated the effectiveness of the Two-IMU fused filter and the ASE algorithm using a set of test data. Furthermore, we conducted two experiments within a circulating flow tank, including straight-line swimming and S-curve swimming to compare the performance of our ASE method and other existing methods.

A. Test Experiments

To evaluate the performance of the two-IMU fused filter, We established a fused Kalman filter and simulated the yaw angle curves of IMU sensors in Matlab. The Gaussian noise with a variance of 0.001 is added to the simulated curves as both process noise and measurement noise, to generate the drift and test the accuracy and stability of the fused filter over a long period. As shown in Fig. 3, the drift of head IMU reaches -3.58° in the 50s and the filter reduces it to only -0.54°. Besides, the drift of tail IMU is 2.07° in 50s and is effectively reduced to -0.48°.

Moreover, we carried out a swinging test without water. The robotic fish was put on a bracket in the air and controlled to swing randomly. we recorded the measured yaw angles of two IMU sensors and further calculated the estimated swimming direction based on the ASE algorithm, which is shown in Fig. 4. The results demonstrate that the ASE algorithm can estimate the real swimming direction even when the swimming direction changes rapidly.

B. Swimming Experiments

a) Experiment Platform: As shown in Fig. 5, we conducted the experiments in a flow tank, where the robotic fish can swim freely. The size of the test section is 87.5×25×25 cm (length×width×height). The tank can generate a uniform flow whose velocity can be adjusted by changing the motor rotational speed:

$$U_{\text{tank}} = \frac{4}{15}U - 4 \tag{11}$$

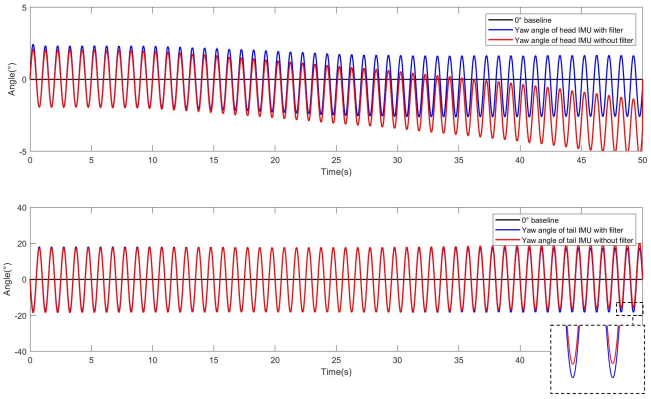

Fig. 3. The results of Two-IMU fused filter in Matlab. Here, we show the yaw drifts of two IMU sensors with (blue curve) and without (red curve) Two-IMU fused filter.

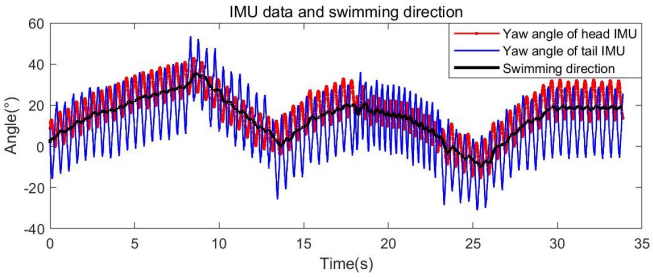

Fig. 4. Swimming direction estimation of the robotic fish based on the ASE algorithm. Here, the swimming direction is calculated from the yaw values of the head IMU and the tail IMU.

where U_{tank} (cm · s^{-1}) is the flow velocity, and U (rpm) is the rotational speed of motor.

A red marker is placed on the top of the robotic fish. With an overhead camera, the position of the robotic fish can be precisely recorded. Since the movement of robotic fish is limited in the narrow flow tank and the position is relatively stable, we cannot capture the absolute trajectory and the real swimming direction. Therefore, the result of our estimation method cannot be directly compared to the real swimming direction. It's worth noting that the direction of the marker on the fish's head is the yaw angle measured by IMU instead of the real swimming direction.

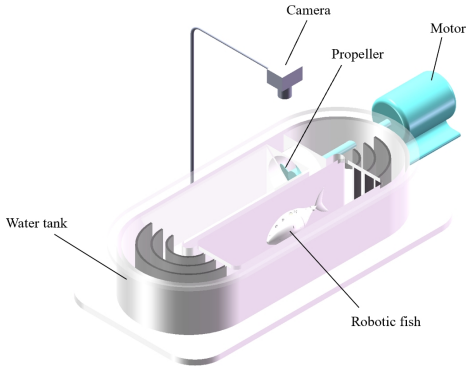

Fig. 5. Experiment platform includes the robotic fish and a flow tank.

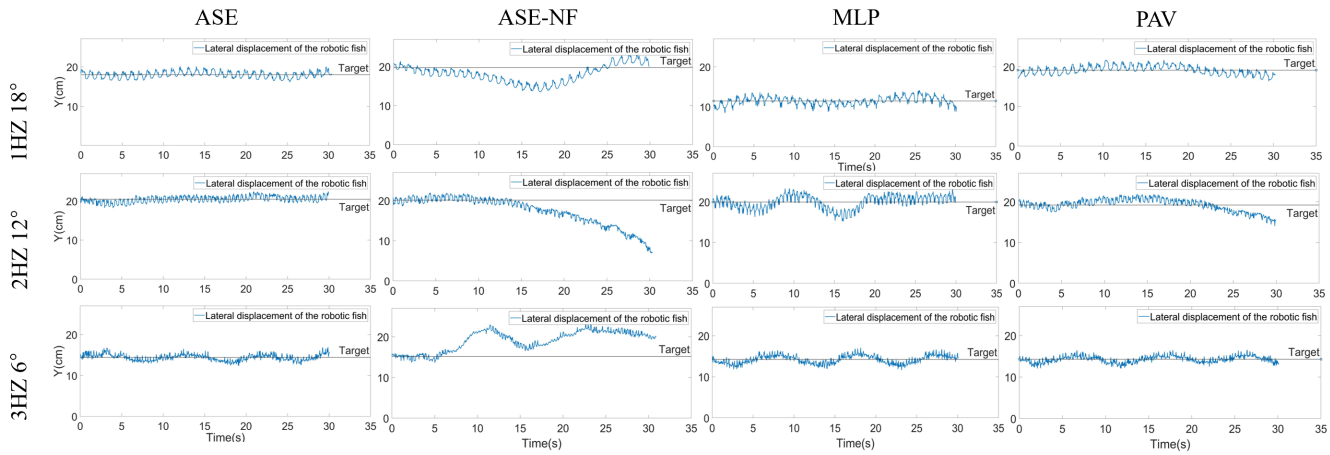

Fig. 6. Lateral displacement error of the robotic fish in straight-line swimming experiment based on different methods.

To solve this problem, we integrated the estimation method into the feedback control of swimming direction and indirectly compared the performance of different methods by analyzing the control results. In the straight-line swimming experiment, where the target swimming direction remains constant, the desired trajectory is a straight line parallel to the tank wall. We evaluated different methods by comparing their lateral displacement errors and evaluated the effectiveness of our ASE method. For the S-curve swimming experiment, the control target is the swimming direction so we do not know the reference trajectory of the robotic fish. Hence, our evaluation is based on observing the shapes of S-curve trajectories rather than comparing specific values of lateral displacement errors.

b) Straight-Line Swimming Experiment: In the straight-line swimming experiment, firstly, the tail of the robotic fish was automatically initialized to the middle. Then, we put the robotic fish into the water with the head direction parallel to the flow and balanced the buoyancy and gravity to reach a stable state before it began swimming. When the robotic fish was static, two IMU sensors were initialized and the yaw angles were set to 0°. When the robotic fish started to swim forward, we adjusted the flow velocity according to the swimming speed so that the absolute position of the robotic fish was stable in the flow tank. With the PID controller, the robotic fish can adjust the offset of the tail to change the swimming direction and swim along the straight line continuously in case the robotic fish is interfered by the lateral force generated by the wall reflection. We compared the lateral error of the position based on the MLP method, PAV method, our proposed ASE method and ASE method without fused filter. In the MLP method, the swimming direction was calculated from the average value of IMU yaw angle data in the last period [11]. In the PAV method, the instantaneous yaw angle when the angular velocity reaches peaks was regarded as the real swimming direction [19]. The same PID controller was applied to control the robotic fish in all methods. We carried out three experiments with different swimming patterns. The frequency is 1 Hz, 2 Hz and 3 Hz, while the swinging amplitude is 18°, 12° and 6° respectively.

To improve the data quality, we removed the acceleration stage of the robotic fish and selected the data after the robotic fish reached a stable state and then compared the results of different methods. The stable-state error (Ess) is defined as the average lateral displacement error between the actual trajectory and straight line in the 30s. To ensure robustness, we conducted the same experiment three times and calculated the mean of the errors to reduce the inherent randomness. Therefore, Ess is defined as follows:

$$Ess = \frac{1}{3}\left(\sum_{i=1}^{3}\frac{1}{n}\sum_{j=1}^{n}(y_j^i - y^*)^2\right) \tag{12}$$

where j represents the time step and n is the total number of time steps over 30s. y_j^i is the actual lateral displacement and y^* is represents the target straight line. Another index to compare the performance of the methods is defined as the range of the lateral displacement in the y-axis during the stable state, which is expressed as:

$$R = \frac{1}{3}\sum_{i=1}^{3}(y_{max} - y_{min}) \tag{13}$$

where y_{max} and y_{min} are the maximum and minimum lateral displacement of the actual trajectory, respectively.

TABLE II
RESULTS ANALYSIS OF THE METHODS

Index	f=1Hz, A=18°		f=2Hz, A=12°		f=3Hz, A=6°	
(cm)	Ess	R	Ess	R	Ess	R
ASE	0.69	3.94	0.70	4.14	0.64	4.51
ASE-NF	5.16	9.78	12.81	15.01	6.15	9.54
MLP	1.01	5.63	2.52	7.20	1.03	5.23
PAV	1.22	5.25	2.37	7.71	0.67	4.41

As Table II shows, the Ess and R based on the ASE without filter (ASE-NF) is much larger than that of the ASE method. The robotic fish deviates from the straight line after a long time due to the IMU drift in Fig. 6. The importance of Two-IMU fused filter in long-term control is demonstrated. In MLP, PAV and our ASE method, the performances of these estimation methods are reflected by the control results. Compared to the MLP and PAV method, our ASE method reduces Ess by 47.54% and 25.38%, and reduces R by 39.73% and 23.03%, respectively. The swimming direction is estimated at each time step but relies on historical data in the past period in the MLP method, which has a delay in estimation and lacks real-time property in the feedback control framework. In the PAV method, although the instantaneous yaw angle is regarded as the swimming direction to solve the problem of time delay, the estimated swimming direction is updated only once every half period. Both time delay and low updating frequency will lead to inaccurate estimation of the swimming direction, and then cause the control error. In our ASE method, both of the problems have been considered. The robotic fish can estimate the swimming direction and change the tail offset to adjust its movement in real-time at a frequency of 100Hz or higher. Therefore, the robotic fish performs better in the straight-line swimming control experiment based on our ASE method.

c) S-Curve Swimming Experiment: To verify our method in complex swimming patterns, the S-curve swimming experiment is conducted based on the different methods above. The target swimming direction α^* is given as:

$$\alpha^* = \sin(\frac{1}{10}\pi t) \tag{14}$$

so that the trajectory of the robotic fish looks like a S-curve. The parameters of PID feedback control are adjusted appropriately so that the robotic fish will not collide with the tank walls. We recorded a 2-minute video of the robotic fish swimming. During our experiments, we exposed the robotic fish to two types of interference. The first is the randomness of flow velocity, which is an artificial interference to simulate the acceleration and deceleration process of the robotic fish. The second is the vortex street generated by the swinging tail and the wall reflection effect, which is the natural interference while the robotic fish is swimming. Affected by the lateral hydrodynamic force generated by the vortex street, the movement of robotic fish is not stable, especially when it deviates from the center of the flow tank, which is a challenge for control. As shown in Fig. 7, our ASE method consistently produced a well-fitted S-curve trajectory over 2 minutes, even in the presence of interference. However, when we employed the ASE method without filter or the MLP method, the S-curve trajectory showed instability. In the PAV method, there is also some deviation in the last 30 seconds of the experiment. These results highlight the robustness and effectiveness of our ASE method in complex swimming environments, demonstrating its capability in control tasks.

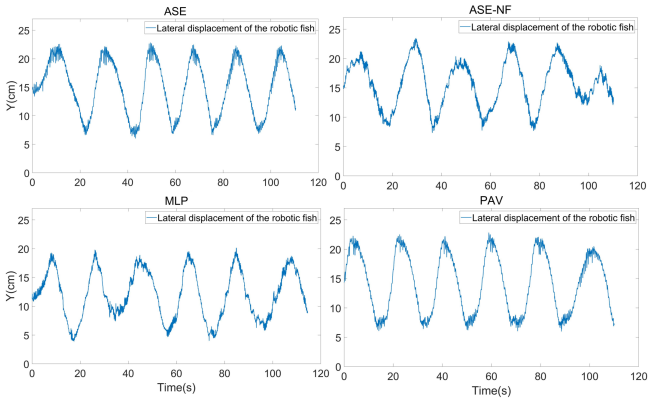

Fig. 7. Lateral displacement error of S-curve swimming based on different methods

C. Discussion

In straight-line and S-curve swimming experiments, if our ASE method was added to the feedback control framework, the robotic fish would be more stable than other methods because our method considered the real-time property and the high updating frequency simultaneously. However, the effectiveness of the ASE method is reflected through feedback control results, which is not intuitive enough. If we could conduct free swimming experiments in a large water tank and record the real trajectory of robotic fish, the performance of our methods could be evaluated more objectively by directly comparing the estimated and real swimming direction. For more universal applications, the fusion of IMU sensors could improve the accuracy of real-time estimation algorithm, regardless of the structures, driving modes and motion patterns of robotic fish. The movement of various robotic fish follows the rule that overall angular momentum is conserved and thus our ASE algorithm works.

V. Conclusion and Future Work

In this paper, we have developed a real-time and high-updating-rate method to enhance the estimation accuracy of swimming direction for robotic fish by integrating two IMU sensors which are respectively installed on the head and tail part. For reliable estimation in the long term, a fused filter was adopted to reduce the yaw angle drift of IMU sensors. With the ASE method, the robotic fish can estimate the swimming direction precisely and respond fast under the PID controller. In straight-line and S-curve swimming experiments, the difference between ASE and ASE without the filter demonstrated the importance of the fused filter to reduce the drift. Besides, the performance of our ASE method was better than other existing methods, which proved the advantages and generality of our method. However, it is unverified whether our ASE method can be applied to more complex flow environments and robot fish trajectories.

Our future work will concentrate on estimation of more gestures of the robotic fish with further fusion of multiple sensors. Moreover, we will conduct different trajectory control tasks in bigger water region and complex flow environments to validate the universality of our method.

REFERENCES

[1] P. W. Webb, "Stability and maneuverability," in *Fish Biomechanics*, ser. Fish Physiology. Academic Press, 2005, vol. 23, pp. 281–332.

[2] P. Domenici and M. E. Hale, "Escape responses of fish: a review of the diversity in motor control, kinematics and behaviour," *Journal of Experimental Biology*, vol. 222, 2019.

[3] J. C. Liao, "A review of fish swimming mechanics and behaviour in altered flows," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 362, pp. 1973 – 1993, 2007.

[4] Y. Ryuh, G.-H. Yang, J. Liu, and H. Hu, "A school of robotic fish for mariculture monitoring in the sea coast," *Journal of Bionic Engineering*, vol. 12, no. 1, pp. 37–46, 2015.

[5] S. Hu, A. Feng, J. Shi, J. Y. Li, F. Khan, H. Zhu, J. Chen, and G. Chen, "Underwater gas leak detection using an autonomous underwater vehicle (robotic fish)," *Process Safety and Environmental Protection*, vol. 167, pp. 89–96, 2022.

[6] R. K. Katzschmann, J. DelPreto, R. MacCurdy, and D. Rus, "Exploration of underwater life with an acoustically controlled soft robotic fish," *Science Robotics*, vol. 3, no. 16, p. eaar3449, 2018.

[7] J. D. Altringham and D. J. Ellerby, "Fish swimming: patterns in muscle function," *Journal of Experimental Biology*, vol. 202, no. 23, pp. 3397–3403, 12 1999.

[8] E. D. Tytell, *Do trout swim better than eels? Challenges for estimating performance based on the wake of self-propelled bodies*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 63–74.

[9] W. Wang, G. Xie, and H. Shi, "Dynamic modeling of an ostraciiform robotic fish based on angle of attack theory," in *2014 International Joint Conference on Neural Networks (IJCNN)*, 2014, pp. 3944–3949.

[10] C. H. White, G. V. Lauder, and H. Bart-Smith, "Tunabot flex: a tuna-inspired robot with body flexibility improves high-performance swimming," *Bioinspiration & Biomimetics*, vol. 16, 2020.

[11] X. Zheng, W. Wang, M. Xiong, and G. Xie, "Online state estimation of a fin-actuated underwater robot using artificial lateral line system," *IEEE Transactions on Robotics*, vol. 36, no. 2, pp. 472–487, 2020.

[12] Y. Zhai, X. Zheng, and G. Xie, "Fish lateral line inspired flow sensors and flow-aided control: A review," *Journal of Bionic Engineering*, vol. 18, pp. 264 – 291, 2020.

[13] T. Salumäe and M. Kruusmaa, "Flow-relative control of an underwater robot," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 469, 2013.

[14] J. Yu, Z. Wu, X. Yang, Y. Yang, and P. Zhang, "Underwater target tracking control of an untethered robotic fish with a camera stabilizer," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 10, pp. 6523–6534, 2021.

[15] J. Zhang, W. Wang, G. Xie, and H. Shi, "Camera-imu-based underwater localization," in *Proceedings of the 33rd Chinese Control Conference*, 2014, pp. 8589–8594.

[16] J. Zheng, T. Zhang, C. Wang, M. Xiong, and G. Xie, "Learning for attitude holding of a robotic fish: An end-to-end approach with sim-to-real transfer," *IEEE Transactions on Robotics*, vol. 38, no. 2, pp. 1287–1303, 2022.

[17] B. Lou, Y. Ni, M. Mao, P. Wang, and Y. Cong, "Optimization of the kinematic model for biomimetic robotic fish with rigid headshaking mitigation," *Robotics*, vol. 6, p. 30, 2017.

[18] J. Chen, B. Yin, C. Wang, F. Xie, R. Du, and Y. Zhong, "Bioinspired closed-loop cpg-based control of a robot fish for obstacle avoidance and direction tracking," *Journal of Bionic Engineering*, vol. 18, pp. 171 – 183, 2021.

[19] Z. Wu, J. Yu, M. Tan, and J. Zhang, "Kinematic comparison of forward and backward swimming and maneuvering in a self-propelled subcarangiform robotic fish," *Journal of Bionic Engineering*, vol. 11, pp. 199–212, 2014.

[20] C. Qiu, Z. Wu, J. Wang, M. Tan, and J. Yu, "Multiagent-reinforcement-learning-based stable path tracking control for a bionic robotic fish with reaction wheel," *IEEE Transactions on Industrial Electronics*, vol. 70, no. 12, pp. 12 670–12 679, 2023.

[21] M. L. Castaño and X. Tan, "Model predictive control-based path-following for tail-actuated robotic fish," *Journal of Dynamic Systems, Measurement, and Control*, 2019.

[22] M. Wang, K. Wang, Q. Zhao, X. Zheng, H. Gao, and J. Yu, "Lqr control and optimization for trajectory tracking of biomimetic robotic fish based on unreal engine," *Biomimetics*, vol. 8, no. 2, 2023.

[23] S. Du, Z. Wu, J. Wang, S. Qi, and J. Yu, "Design and control of a two-motor-actuated tuna-inspired robot system," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 8, pp. 4670–4680, 2021.

[24] R. Tong, Z. Wu, D. Chen, J. Wang, S. Du, M. Tan, and J. Yu, "Design and optimization of an untethered high-performance robotic tuna," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 5, pp. 4132–4142, 2022.

[25] J. Zhu, C. White, D. K. Wainwright, V. D. Santo, G. V. Lauder, and H. Bart-Smith, "Tuna robotics: A high-frequency experimental platform exploring the performance space of swimming fishes," *Science Robotics*, vol. 4, no. 34, p. eaax4615, 2019.

[26] L. Li, C. Wang, and G. Xie, "A general cpg network and its implementation on the microcontroller," *Neurocomputing*, vol. 167, pp. 299–305, 2015.