University of Essex

Research Repository

A novel bionic mantis shrimp robot for tracking underwater moving objects

Accepted for publication in Ships and Offshore Structures.

Research Repository link: https://repository.essex.ac.uk/38467/

Please note:

Changes made as a result of publishing processes such as copy-editing, formatting and page numbers may not be reflected in this version. For the definitive version of this publication, please refer to the published source. You are advised to consult the publisher's version if you wish to cite this paper.

www.essex.ac.uk

University of Essex

A novel bionic mantis shrimp robot for tracking underwater moving objects

Gang Chen[a,b], Yidong Xu[a], Jiajun Tu[c], Xin Yang[a], Huosheng Hu[b], Fei Dong[d], Jingjing Zhang[e] and Jianwei Shi[e]

[a]School of Mechanical Engineering, Zhejiang Sci-Tech University, Hangzhou, People's Republic of China; [b]School of Computer Science and Electronic Engineering, University of Essex, Colchester, UK; [c]Faculty of Intelligent Manufacturing, Jiaxing Vocational & Technical College, Jiaxing, People's Republic of China; [d]Zhejiang Chengshi Robot Co., Ltd., Jiaxing, People's Republic of China; [e]Zhejiang CathayBot Technology Co., Ltd., Jinhua, People's Republic of China

ABSTRACT

Inspired by the agility of mantis shrimp, this paper presents the development of a novel bionic mantis shrimp robot with multiple pleopods coupled motion for underwater target recognition and tracking. Firstly, the novel bionic robot is constructed based on the inspiration from biological mantis shrimp. Secondly, a Kalman filter-based algorithm, namely MobileNet-YOLO (KMY), is created to collect datasets and train neural networks. Thirdly, a moving target following system is developed for the bionic mantis shrimp robot, in which a dual PID controller is used to track underwater moving targets. The real robot testing results show that the proposed control system can enable our bionic robot to follow a specific moving target in a narrow pool (2m × 1m × 1m), and the minimum turning radius can be up to 0.55m when the angle between the robot's initial motion and the motion of target is 90°.

KEYWORDS
Underwater target sensing; underwater target following control; bionic mantis shrimp robot; bionic robots; underwater robot

1. Introduction

The development of underwater target recognition and following technology has become increasingly important with the growth of human demand for deep-sea resources and the exploration of the marine environment (Kiuru et al. 2014; Teng and Zhao 2020). This technology can help people better understand marine ecosystems, conduct marine scientific research, and play an important role in underwater rescue, seabed resource exploration and biological sampling, etc. (Teng and Zhao 2020; Chen et al. 2023a, 2023b). The current research on underwater target recognition and following faces many challenges.

First, the underwater environment is complex and changeable and contains various water currents, suspended objects and other disturbing factors that affect the robot's sensing ability and motion control (Patrón and Petillot 2008; Jung et al. 2018). Secondly, the diverse characteristics of underwater targets require robots to have strong target recognition capabilities (Azimi-Sadjadi et al. 2000). As marine organisms come in a wide variety of species and forms, robots need to be able to accurately identify different species of marine organisms and differentiate their behaviours and characteristics (Mackenzie 2002; Porte et al. 2006; Williams and Fakiris 2014). Thirdly, the appearance of underwater targets may be affected by water currents, light, and other factors, further increasing the difficulty of identification (Weitbrecht et al. 2002; Jian et al. 2021). Fourthly, the robot needs to have high response velocity and accurate motion control capabilities to follow the fast motion of underwater targets (Chen et al. 2023c). Moreover, the robot also needs to consider water resistance and tides so that a stable following effect can be achieved (Yuh and West 2001; Zhao and Yuh 2005; Chen et al. 2022).

To overcome these challenges, researchers have used a variety of methods and techniques (Chen et al. 2024a). Shi et al. (2022) used a binocular vision system to enable real-time underwater localisation and calculation of 3D coordinates for target tracking. Wei et al. (2018) proposed a dynamic target-tracking method for autonomous underwater vehicles based on the model predictive control (MPC) algorithm. Real-time identification and localisation of dynamic targets were achieved by processing the sonar images of underwater vehicles (Wei et al. 2018). Sattar and Dudek (2006) investigated the application of visual target tracking in the autonomous turning of an underwater robot, demonstrating the effect of suspended particles on tracking. Prasad et al. (2015) proposed a visual servo system for a spherical underwater robot capable of tracking and following a target underwater.

However, as the underwater targets have diverse features, the stable and fast underwater target recognition and following remain a great challenge (Wang et al. 2019; Honghui et al. 2022; Yuan et al. 2022). Mantis shrimps in nature can swim quickly through the water and achieve precision strikes on their prey, providing inspiration for our research (Steinhardt et al. 2021; Patek et al. 2004; Chen et al. 2023b). The contributions of this paper can be summarised below.

- Based on the motion mechanism of biological mantis shrimp, we designed a bionic mantis shrimp robot to adapt to the detection of underwater narrow environments. The body of the robot is composed of five pleopod bases and a flexible spine, which can realise flexible underwater motion.
- Based on underwater vision and embedded hardware, we have developed a robot vision detection and tracking system that can be used to track underwater moving objects, enabling autonomous underwater target recognition and tracking, effectively expanding the environmental perception capabilities of the bionic mantis shrimp robot.

- A double PID algorithm was proposed for underwater motion and target following, which was verified by a simulation model and a real experiment environment.

The structure of this paper is organised as follows. Section 2 describes the design of the bionic mantis shrimp robot. Section 3 proposes the underwater image target recognition and tracking algorithm. Section 4 establishes the underwater target following the simulation system. Section 5 is simulation and experimental verification. Finally, a brief discussion and summary is given in Section 6.

2. Design of bionic mantis shrimp robot

The mantis shrimp is a carnivorous marine crustacean characterised by powerful forelimbs that can deliver devastating blows to prey targets (Chen et al. 2024b). In addition, the mantis shrimp is also one of the fastest swimmers in the water. The flexible and soft swimming pleopods are the main source of power. The movement of pleopods and body bending can be adjusted to achieve rapid swimming and dextrous turning with excellent motion coordination (Burrows and Hoyle 1972).

As shown in Figure 1(a), the mantis shrimp mainly consists of four parts such as head, body, pleopods and telson. Here, the structure and motion of biological mantis shrimp are used as the basis for the design of a bionic mantis shrimp robot. The structure of the robot is shown in Figure 1(b), which is mainly divided into three parts: head, body, and telson. The head of the robot consists of a waterproof box, in which the control hardware system is installed. Figure 1(c) shows the bionic pleopod and flexible spine of the bionic mantis shrimp robot. The first joint of the bionic pleopod is an active joint, and the second and third joints are passive joints, which interact with the water flow to realise unfolding and folding. The bionic pleopod adopts a modular design, with simple structure, efficient movement, and easy maintenance. In order to realise the flexible motion of the mantis shrimp robot, the flexible spine is designed, and the bending angle of the body is controlled by rope driving. The flexible body adopts rigid-flexible coupling design to reduce the impact of water flow, and the movement is more stable, flexible and the turning response is faster.

Figure 2 shows the motion control hardware and vision hardware in the waterproof box, both of which communicate through the serial port. Our bionic robot deploys a camera to obtain real-time environmental information and adjust the robot motion. The camera model is a monocular camera OV2640 with a viewing angle of 120°. The visual development board is MaixBit, and the core processor is k210. Its neural processing image acceleration chip KPU (Kendryte Process Unit) can realise the hardware acceleration of four basic operations: convolution, batch normalisation, activation and pooling. Motion control board for ESP32 can achieve dual thread work. Table 1 shows the core parameters of the servo motor used by the robot. When the three joints of the pleopod are unfolding, the torque required for the maximum load work of a single pleopod is calculated to be 0.15 kg cm, which is much less than the torque of the selected servo 1.9 kg cm. When the body bending Angle is 35°, the maximum torque required for turning is calculated to be 6.2 kg cm, which is less than 11.3 kg cm of the servo. The motion of the servo motor is controlled by the PWM signal issued by the servo control board PCA9680. The motion control of the robot is all computationally controlled by embedded hardware.

3. Robotic target sensing algorithm for underwater moving objects

Robot target sensing algorithm for underwater moving objects includes robot target recognition and tracking. We propose a Kalman filter-based MobileNet-YOLO (KMY) target recognition and tracking algorithm, which is mainly divided into two parts: target recognition and target tracking. The target recognition algorithm is based on MobileNet-YOLO, which can recognise the desired underwater target and output its position in the image and the

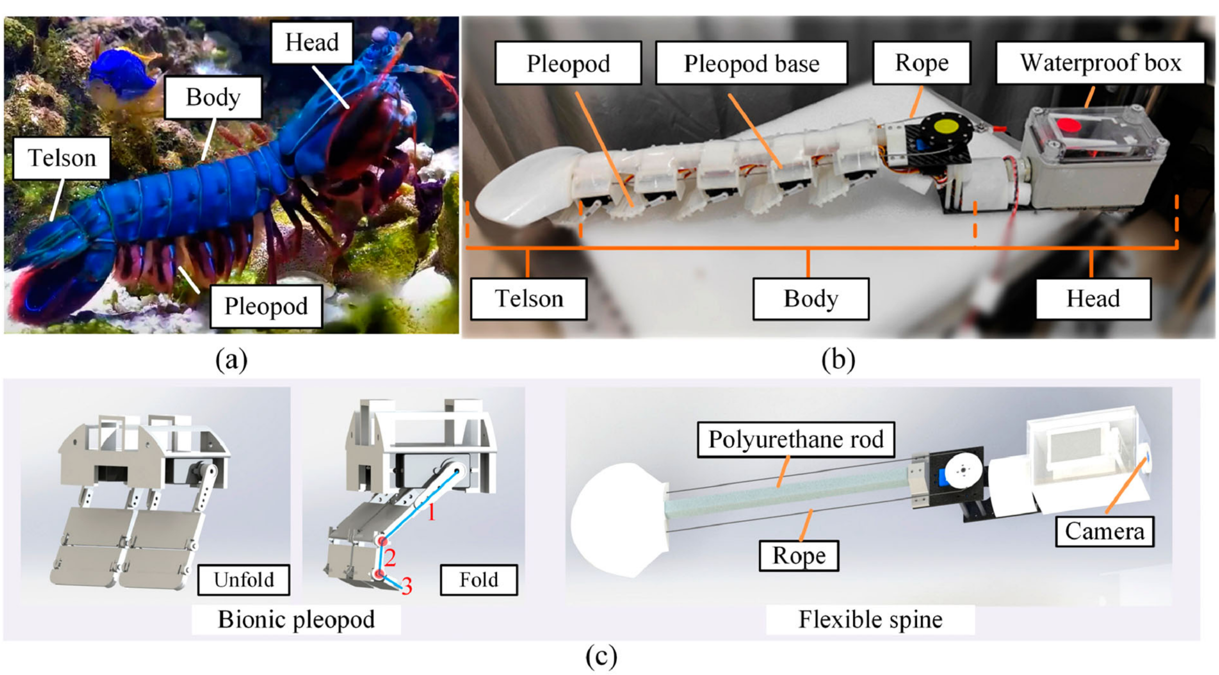


Figure 1. Bionic mantis shrimp robot system. (a) Biological mantis shrimp, (b) bionic mantis shrimp robot and (c) bionic pleopod and flexible spine.

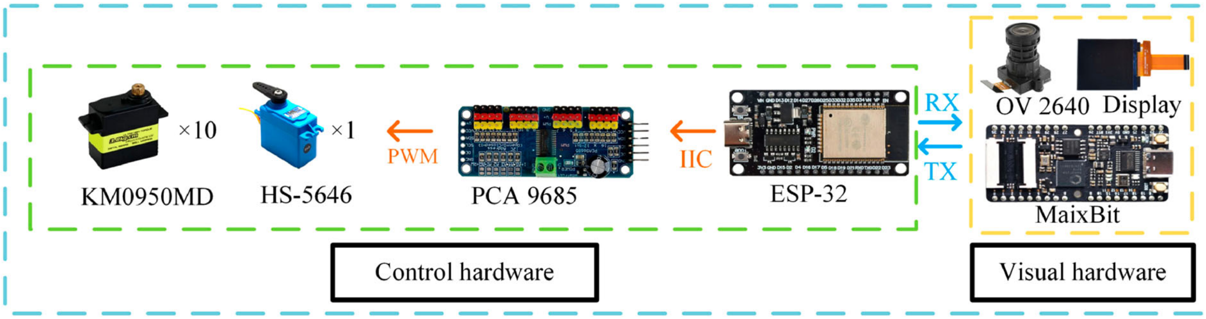


Figure 2. Robot control hardware.

Table 1. Servo motor parameters.

Servo motor type	KM0950MD	HS-5646
Weight	18 g	61 g
Dimensions	26 × 14 × 22.6 mm	41.8 × 21 × 40 mm
Dead band	2 μs default	2 μs default
Speed (6.0 V)	0.12 s/60°	0.2 s/60°
Rated torque	1.90 kg cm	11.30 kg cm
Angle of motion	0–90°	0–180°

size of the detection frame. The target tracking algorithm is based on the Kalman filter method, which establishes a stable tracking model for the desired tracking target.

Figure 3 shows the flowchart for a single algorithm work cycle of the underwater target recognition and tracking algorithm. After the algorithm successfully detects a desired tracking target, the result is output to the Kalman filter tracker as a new state observation. Then,

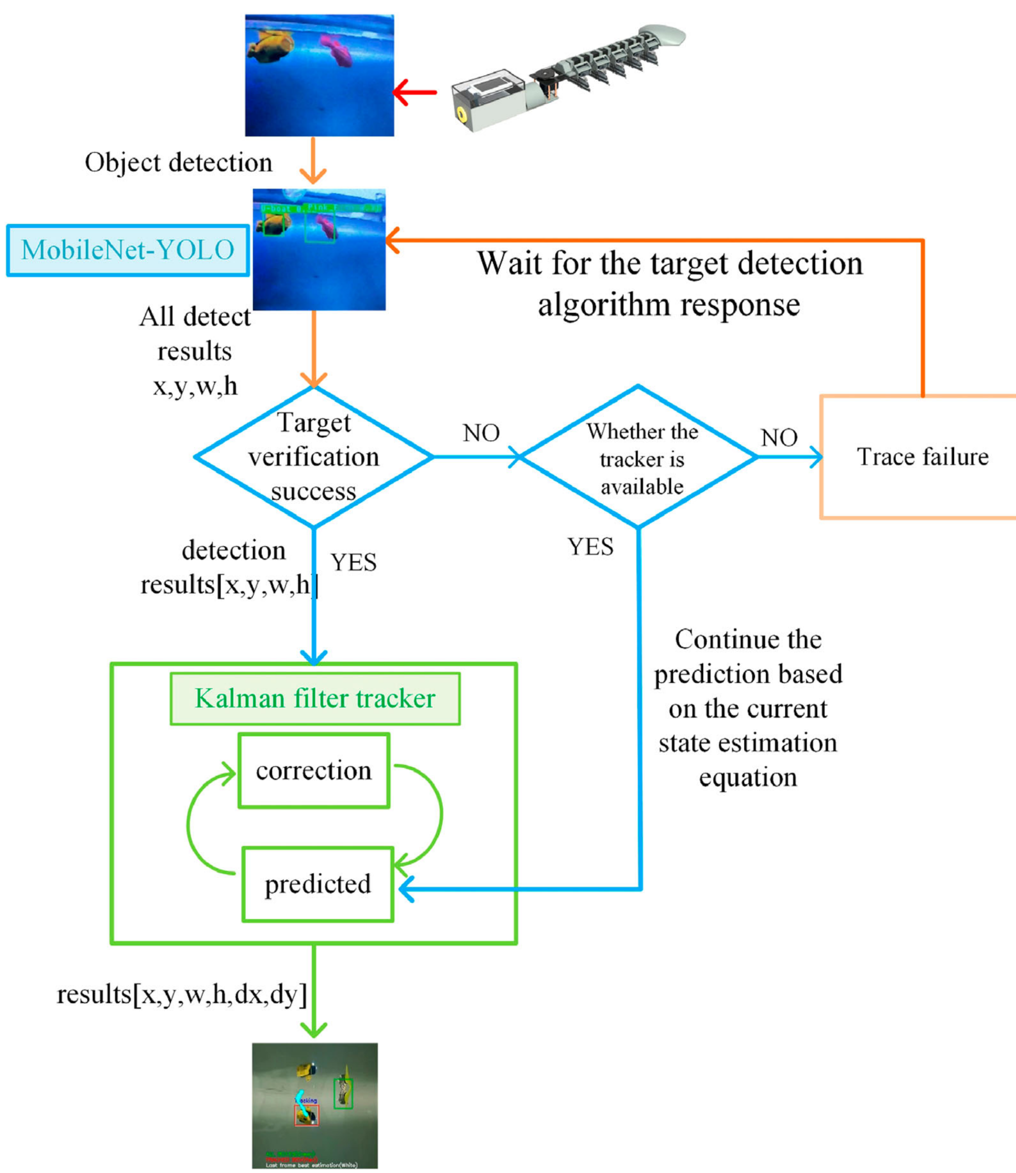


Figure 3. Flowchart of KMY algorithm in a single work cycle.

the state estimation covariance and Kalman gain matrix are modified in real time. The target state is corrected, and the optimal estimation result of the desired tracking target is obtained. If the target detection algorithm fails, different operations are performed according to whether the current tracker is available or not. When there is no available tracker, the program enters into a silent state waiting for the successful detection result of the next target detection algorithm. If there is an available tracker, the algorithm continues to estimate the target state at the next moment according to the state equation of the current tracker and outputs it as the tracking result.

3.1. MobileNet-YOLO-based underwater target recognition algorithm

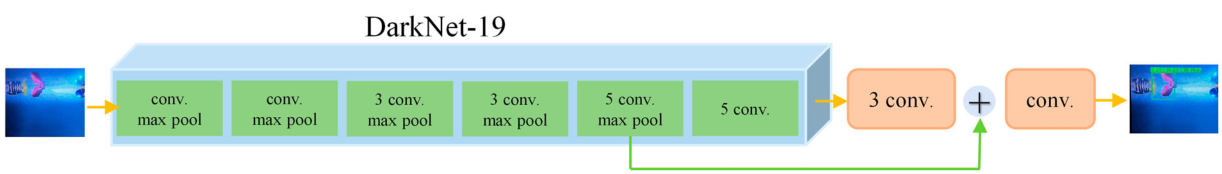
3.1.1. Structure of robotic underwater target recognition algorithm based on MobileNet-YOLO

In this study, MobileNet-YOLOv2 was chosen as a target detection method for underwater images. It was the feature extraction backbone network of YOLOv2 and replaced the original DarkNet-19 with MobileNet, thus forming a target detection algorithm with fewer parameters and faster speed (Redmon and Farhadi 2017; Li et al. 2020; Shafi et al. 2022). This algorithm can run in low-computing power, low-power embedded devices to show excellent detection performance. The network model of YOLOv2 (Figure 4(a)) was used, and the image targets were extracted by the feature extraction backbone network (Backbone) DarkNet-19 after the image input. Finally, the detection target classification and bounding box information were extracted by the convolutional layer. Figure 4(b) shows the network structure of the backbone network DarkNet-19, which includes 19 convolutional layers and 5 Maxpooling layers. The application of this backbone network can reduce the amount of computation and parameters in the process of feature extraction (Al-Haija et al. 2021).

As shown in Figure 5(a), the comparison between the standard convolutional operation (left) and the depth-separable convolutional operation (right) is conducted. The core innovation of MobileNet is that a depth-separable convolutional operation is proposed, which can reduce the amount of computation with the same number of weights and parameters, and significantly increase the speed of network operation (Li et al. 2018). The network structure of MobileNet is shown in Figure 5(b), which has a total of 28 layers. Its input is a $224 \times 224 \times 3$ image, and the output is the result of the feature classification.

3.1.2. Algorithmic datasets

To test the system's tracking ability for dynamic targets in the subsequent study, three remotely controllable small underwater robots are selected as the preparatory tracking targets and the dataset is established. They were manually photographed and collected and converted to the required image size for YOLOv2, and the target information was labelled using Labelimg, and the software labelling interface is shown in Figure 6(a). The three small underwater robots are labelled as submarine, pink fish and striped fish (U-boat, Pink

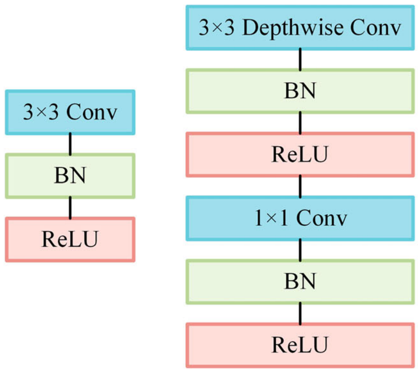


(a)

Type	Filters	Size/Stride	Output
Convolutional	32	3×3	224×224
Maxpool		$2 \times 2/2$	112×112
Convolutional	64	3×3	112×112
Maxpool		$2 \times 2/2$	56×56
Convolutional	128	3×3	56×56
Convolutional	64	1×1	56×56
Convolutional	128	3×3	56×56
Maxpool		$2 \times 2/2$	28×28
Convolutional	256	3×3	28×28
Convolutional	128	1×1	28×28
Convolutional	256	3×3	28×28
Maxpool		$2 \times 2/2$	14×14
Convolutional	512	3×3	14×14
Convolutional	256	1×1	14×14
Convolutional	512	3×3	14×14
Convolutional	256	1×1	14×14
Convolutional	512	3×3	14×14
Maxpool		$2 \times 2/2$	7×7
Convolutional	1024	3×3	7×7
Convolutional	512	1×1	7×7
Convolutional	1024	3×3	7×7
Convolutional	512	1×1	7×7
Convolutional	1024	3×3	7×7
Convolutional	1000	1×1	7×7
Avgpool		Global	1000
Softmax			

(b)

Figure 4. YOLOv2 algorithm framework. (a) YOLOv2 network model. (b) Darknet-19 feature extraction network.

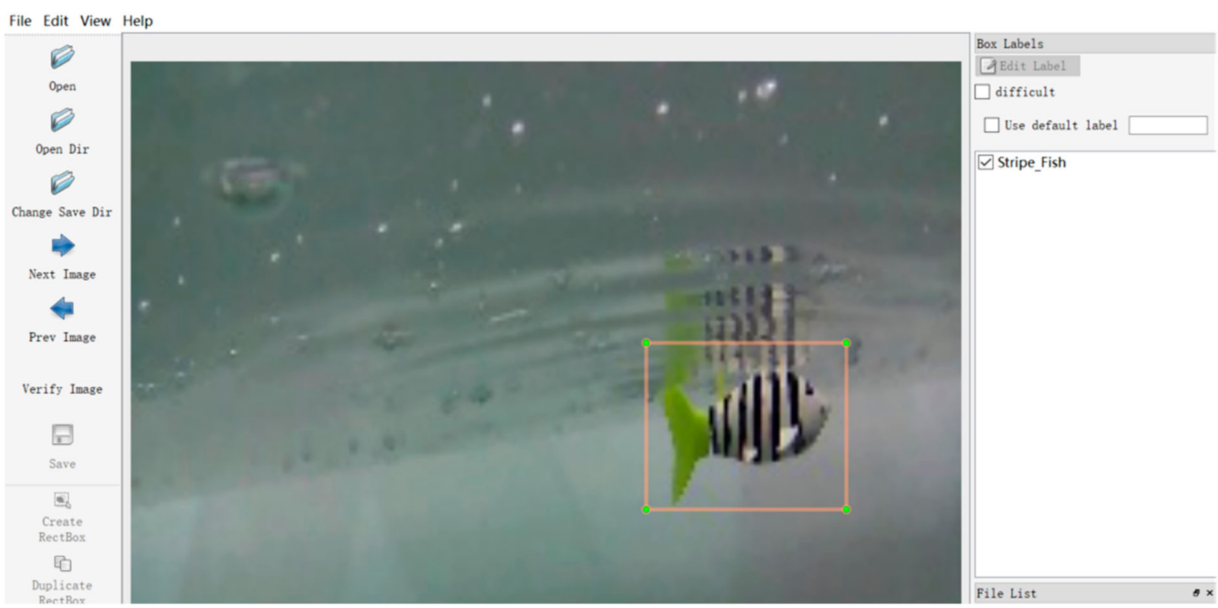


(a)

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5\times$ Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
$5\times$ Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

(b)

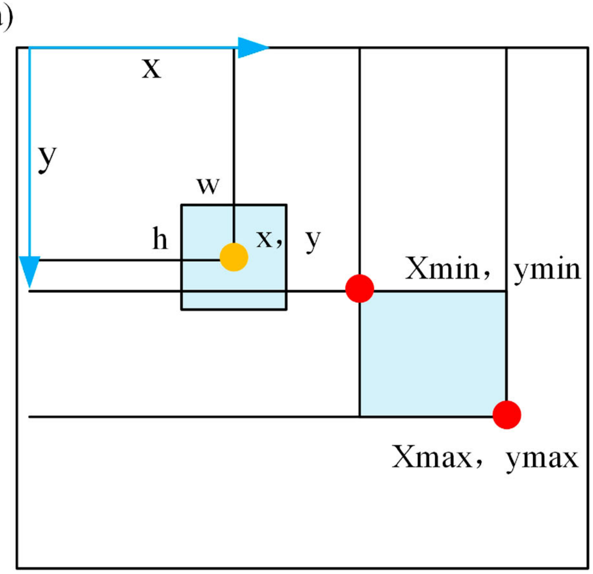
Figure 5. Depth-separable convolution and MobileNet network structure. (a) Depth-separable convolution operation. (b) MobileNet network structure.



(a)

```
<annotation>
        <folder>images</folder>
        <filename>957.jpg</filename>
        <size>
                <width>224</width>
                <height>224</height>
                <depth>3</depth>
        </size>
        <object>
                <name>Stripe_Fish</name>
                <pose>Unspecified</pose>
                <truncated>0</truncated>
                <difficult>0</difficult>
                <bndbox>
                        <xmin>67</xmin>
                        <ymin>127</ymin>
                        <xmax>158</xmax>
                        <ymax>190</ymax>
                </bndbox>
        </object>
</annotation>
```

(b)



(c)

Figure 6. Production process of the dataset.

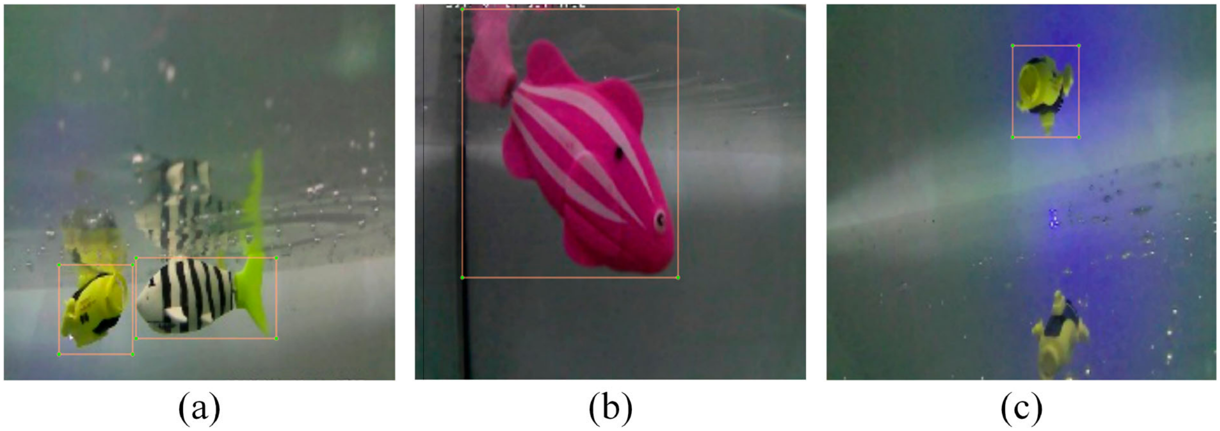


Figure 7. Target images with their labelling results. (a) Submarine and striped fish. (b) Pink fish. (c) Submarine.

Fish, and Striped Fish) according to their colours and features, respectively.

The file information generated by the labelling is shown in Figure 6(b), whose main information includes the image file name, the image size and the bounding box data of the labelled target. Figure 6(c) shows the definition of the image coordinate system and the parameters of the bounding box. As shown in the blue coordinate system in Figure 6(c), the image coordinate system takes the upper left corner of the image as the origin, to the right is the coordinate system x-axis, and down is the coordinate system y-axis.

The dataset has a total of 1964 images of the three tracking targets. All the images in the dataset are divided into a training set, a validation set, and a test set according to the ratio of 8:1:1. The labelling results of the three targets in the training set are shown in Figure 6. The labelled targets are submarine and striped fish in Figure 7(a), pink fish in Figure 7(b), and submarine in Figure 7(c).

3.1.3. Algorithm training and deployment

The MobileNet-YOLOv2 target detection algorithm was trained after the training dataset was produced. For the configuration of the training task, the number of scheduled training sessions was 8200. The loss function variation with mean average precision (mAP) for the training process is shown in Figure 8(a). After 4100 training iterations, the network model has basically converged, and the loss function drops to about 0.2 and then no longer decreases significantly, indicating that the network model has been trained at this time. mAP is an evaluation index used to reflect whether the detection results are correct or not, and its data is basically stable at more than 90%, indicating that at this time, the network model shows excellent detection performance in the test set.

To enable the bionic mantis shrimp robot to complete target detection without external device communication, the trained network model needs to go through format conversion and parameter model downloading before it can run successfully in the Maixbit development board. Figure 8(b) shows the flowchart of the

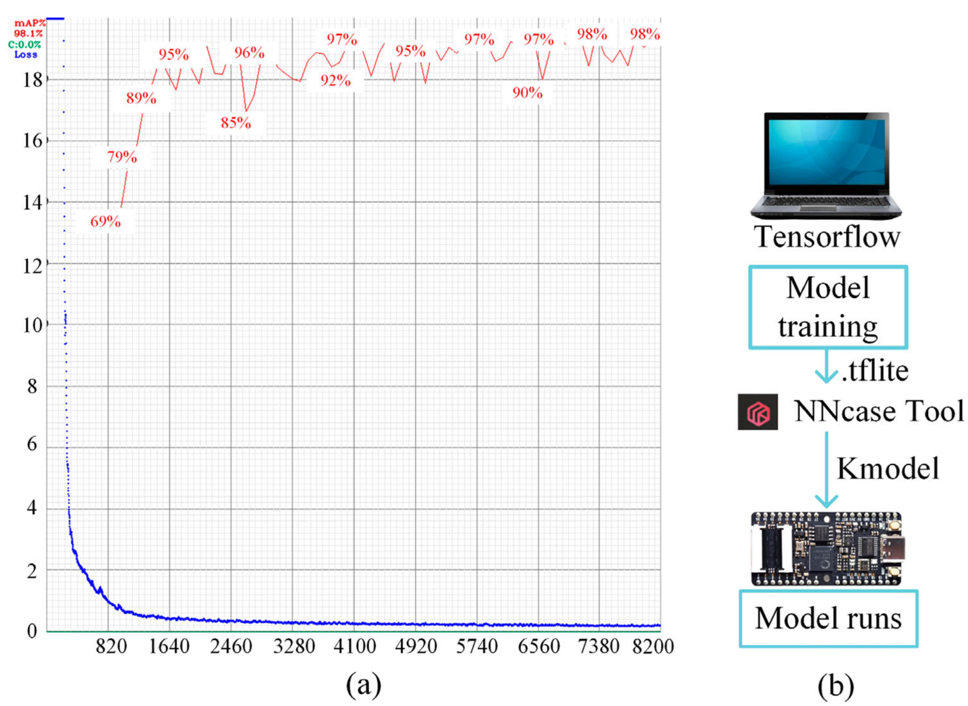


Figure 8. Training process and model deployment. (a) Data changes during training. (b) Edge deployment process for neural network algorithms.

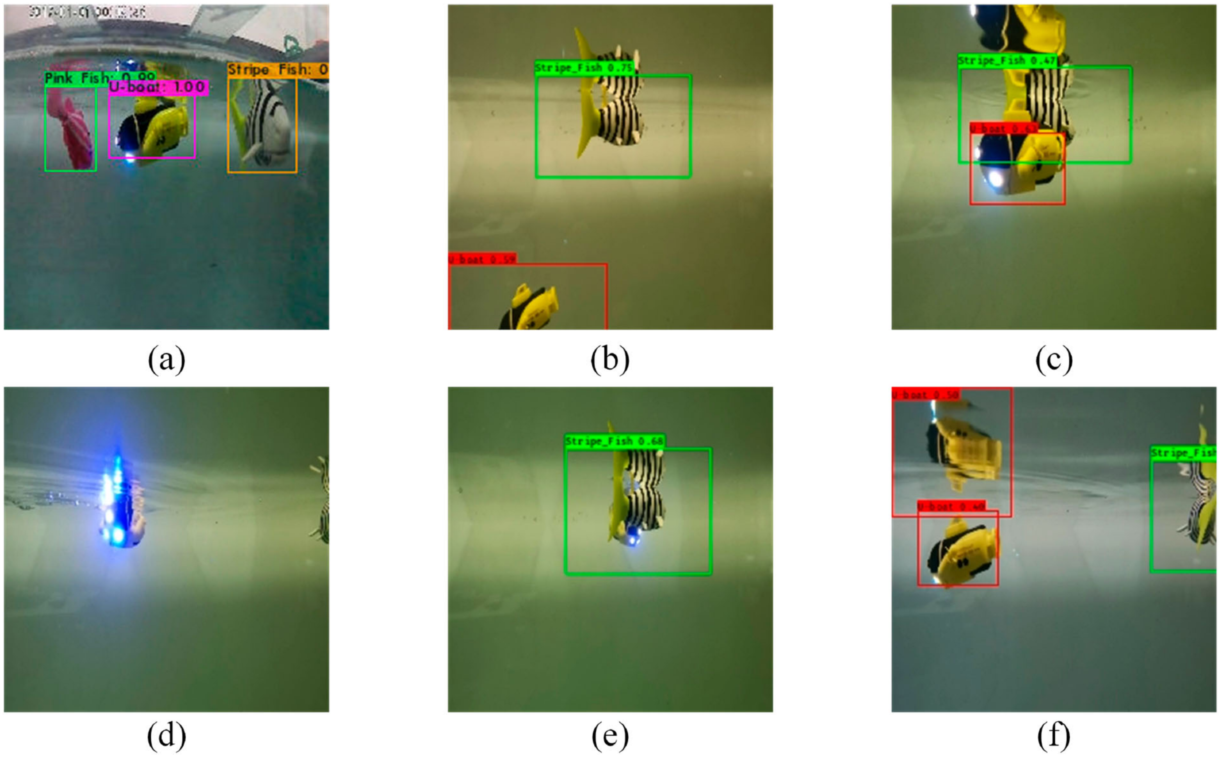


Figure 9. Results of model inspection.

algorithm deployment. After completing the model training under the Tensorflow framework on the PC, the model is verified, compiled, and transformed using the NNcase open-source tool to obtain the Kmodel format for the Maixbit development board. Figure 9 shows the actual detection results after the model training is completed. More specifically, Figure 9(a)–(c) shows the successful detection, and Figure 9(d)–(f) shows the missed detection, failed detections due to occlusion, and the water surface reflection, respectively. Because the camera used is a monocular camera, the target is at the junction of the water surface, so there will be the situation of the target and its reflection, resulting in the misjudgment of the recognition algorithm.

3.2. Kalman filter-based underwater target tracking algorithm

The MobileNet-YOLO-based target detection algorithm can obtain the detection coordinates and target size of the tracked target in the current image. However, its detection results will have numerical jumps and loss as the detection results do not have the correlation between the previous and subsequent frames and the problem of leakage detection will inevitably occur, which will lead to tracking failure. Kalman filtering can provide an effective recursive solution based on the least squares method and solve the general problem of controlled state change of a system described by linear difference equations (Chui and Chen 2017; Grewal et al. 2020). It obtains the model of the state change of a signal in the presence of a noisy signal to realise the estimation of the past and future values of the target signal (Grewal and Andrews 2014).

For a discrete linear dynamic system, it can be modeled as follows.

$$x_k = A \cdot x_{k-1} + B \cdot u_k + w_{k-1} \tag{1}$$

$$z_k = H \cdot x_k + v_k \tag{2}$$

In Equation (1), x_k is the state matrix of the system at the moment k, A is the state transfer matrix, B is the control input matrix, x_{k-1} is the state matrix of the system at the moment $k-1$, and w_{k-1} is the process noise of the system at the moment $k-1$.

In Equation (2), z_k is the state observation equation of the system at the time k, H is the state observation matrix, and v_k is the measurement noise of the system at the time k.

To estimate the state of the system, three state evaluation parameters are defined: the state truth value x_k, the state prediction value $\tilde{x}_k^-$ (a prior state estimate), and the state optimal estimate $\tilde{x}_k$ (a posteriori state estimate).

Based on Equations (1) and (2), the state prediction equation is:

$$\tilde{x}_k^- = A \cdot \tilde{x}_{k-1} + B \cdot u_k \tag{3}$$

The state update equation is:

$$\tilde{x}_k = \tilde{x}_{k-1} + K \cdot (z_k - H \cdot \tilde{x}_k^-) \tag{4}$$

In the formula, $K = P_k^- H^T (H P_k^- H^T + R)^{-1}$ represents the Kalman gain in the range of $K = P_k^- H^T (H P_k^- H^T + R)^{-1}$. The value of Kalman gain represents the weight of the prediction error and measurement error of the system model in the process of optimal state estimation.

Under the optimal estimation condition, the Kalman gain matrix $K = P_k^- H^T (H P_k^- H^T + R)^{-1}$ is:

$$K = P_k^- H^T (H P_k^- H^T + R)^{-1} \tag{5}$$

where P_k^- represents the covariance between the true value of the state and the predicted value of the state, and P_k represents the covariance between the true value and the optimal estimate,

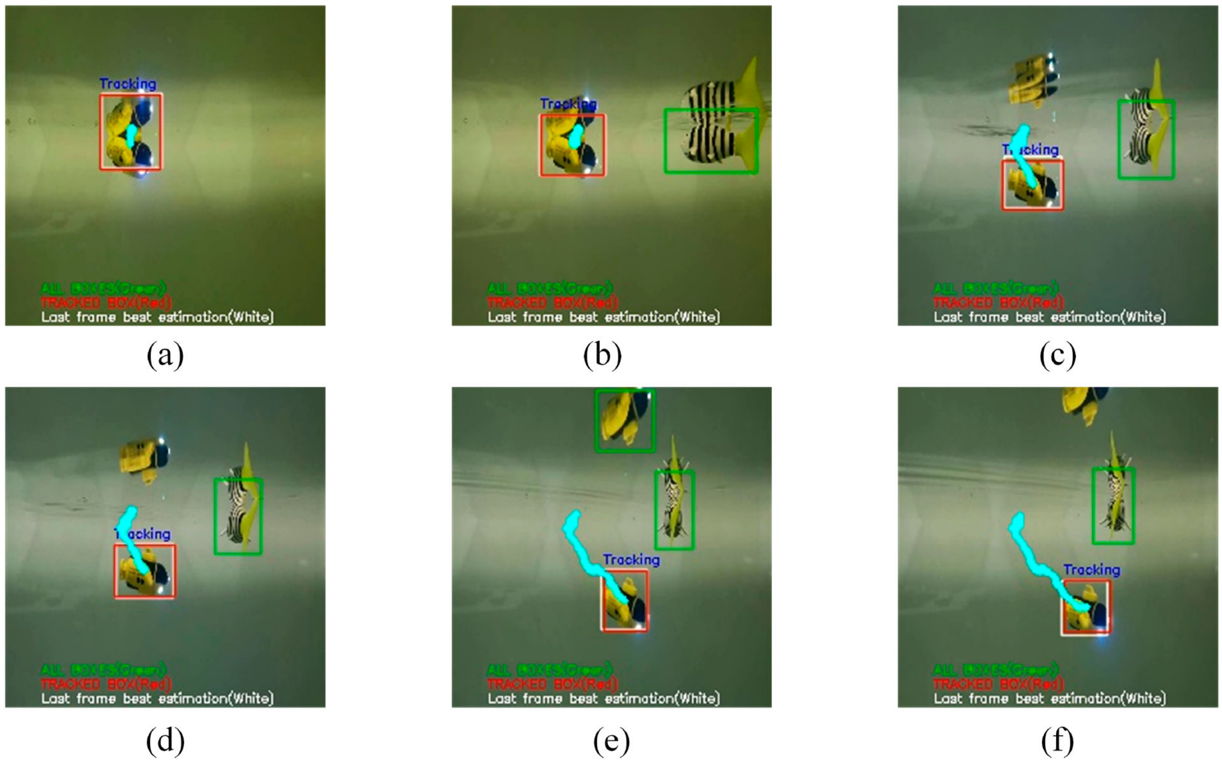


Figure 10. Validation results of the KMY algorithm.

which is expressed as follows.

$$P_k = (I - KH) \cdot P_k^- \tag{6}$$

Combining Equations (5) and (6) yields the prediction covariance matrix $P_{k+1}^- = AP_kA^T + Q$ as:

$$P_{k+1}^- = AP_kA^T + Q \tag{7}$$

Based on the above formula, the Kalman filter will complete the state prediction and state update of the model in real time and optimise the value of the Kalman gain in a rolling manner to correct the state prediction value and reduce its deviation from the true value of the state. The observation matrix of the Kalman filter is: $H = [xyhw]^T$. The result of the target detection algorithm can get the centre coordinates of the tracking target: x, y, and the width h and height w of the detection frame. For realising the dynamic tracking of the target and predicting the target's motion, the target's speed dx and dy in the image coordinate system should be considered. The following state equation can be established for the tracking target:

$$\text{state} = \begin{bmatrix} x & y & h & w & dx & dy \end{bmatrix}^T \tag{8}$$

where the computational expressions for dx and dy are:

$$\begin{cases} dy = \dfrac{(y_t - y_{t-1})}{\triangle t} \\ dx = \dfrac{(x_t - x_{t-1})}{\triangle t} \end{cases} \tag{9}$$

where $\triangle t$ is the work cycle interval of the algorithm.

Figure 10 shows the results of the algorithm's tracking validation for targets in water. The validation process was carried out in a pool, where a total of two detection targets appeared, with the yellow submarine as the desired tracking target for this validation. The algorithm is run on the bionic mantis shrimp robot and the camera returns images with pixel size of 224×224. Three colour bounding box markers exist in the figure, where the green bounding box represents the target detection result, the white bounding box represents the optimal estimation bounding box of the previous moment, and the red bounding box represents the latest tracking result of the desired tracking target. The blue line represents the centre motion trajectory of the tracking result.

More specifically, Figure 10(a) shows that the algorithm correctly selects the desired tracking target based on the results of the target detection algorithm. Figure 10(b) shows that the tracker continues to track the desired tracking target without misdetection when the new target appears. Figure 10(c,d) shows the tracker continues to track it and records the tracking target's motion tracking data when the desired tracking target is in

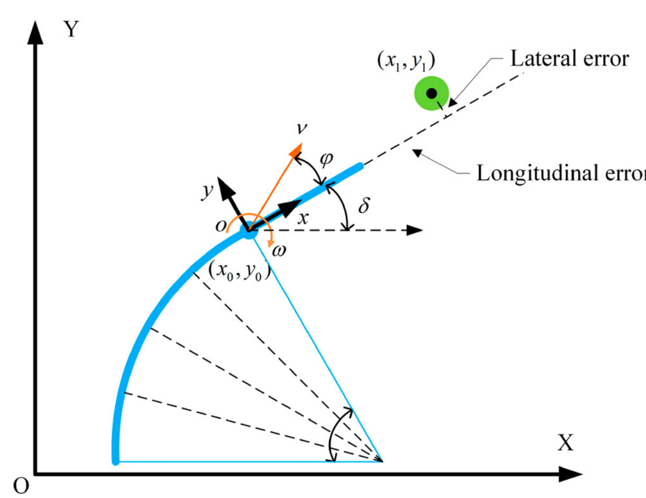


Figure 11. Coordinate system and error definition.

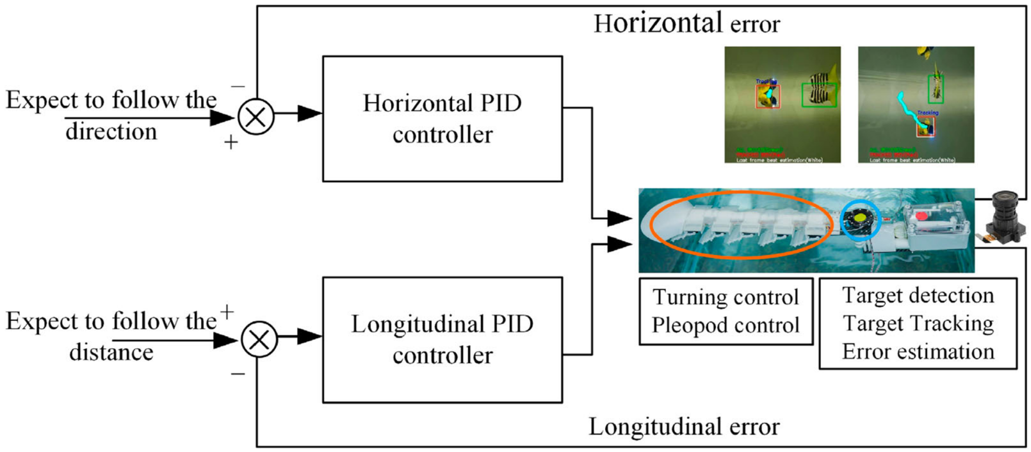


Figure 12. Block diagram of an underwater target following system.

motion. In Figure 10(e), the reflection of the submarine appears due to the occurrence of water surface reflection and the detection algorithm results in misdetection. But the tracker continues its operation based on the detection results of the previous moment to maintain the tracking state of the target without influences by the misdetection results. Figure 10(f) shows that the target detection results are restored to be correct, and the tracking algorithm continues to output the tracking results of the desired tracking target.

4. Design of a PID-based underwater target following system

When the bionic mantis shrimp robot moves in the water, it is only capable of accomplishing forward acceleration, deceleration, and turning motions. Depending on the motion state of the desired following target, the distance errors in the horizontal and longitudinal direction will exist. Figure 11 shows the definition of the body motion state, the desired tracking target motion state, and the horizontal and longitudinal error.

Here, we propose a dual PID underwater target following control system for bionic mantis shrimp robot based on the principle of PID controller implementation. The block diagram of the system is shown in Figure 12. The mathematical expression of the PID controller is as follows:

$$u(t) = K_p \cdot e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \tag{10}$$

where K_p, K_i and K_d represent the proportional, differential, and integral gains of the PID, respectively.

Its working principle is that: the robot detects the desired tracking target and then tracks it. According to the real-time tracking

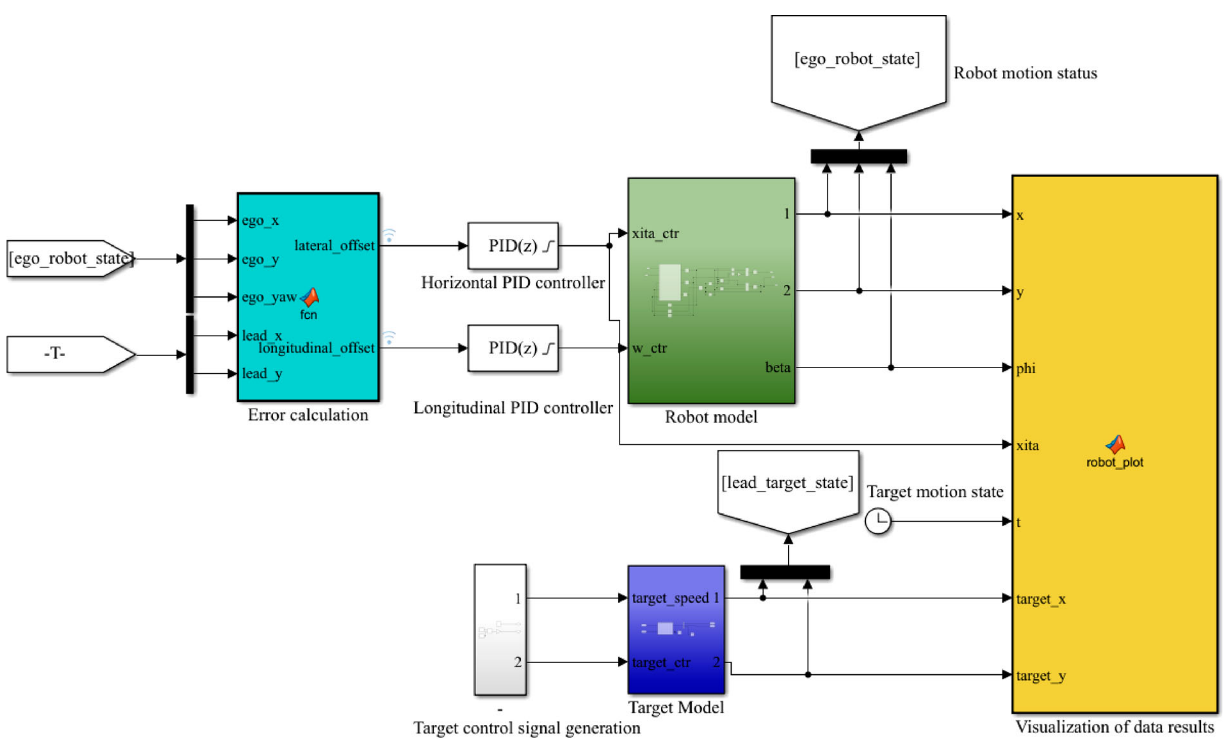


Figure 13. Simulation model of underwater target following system.

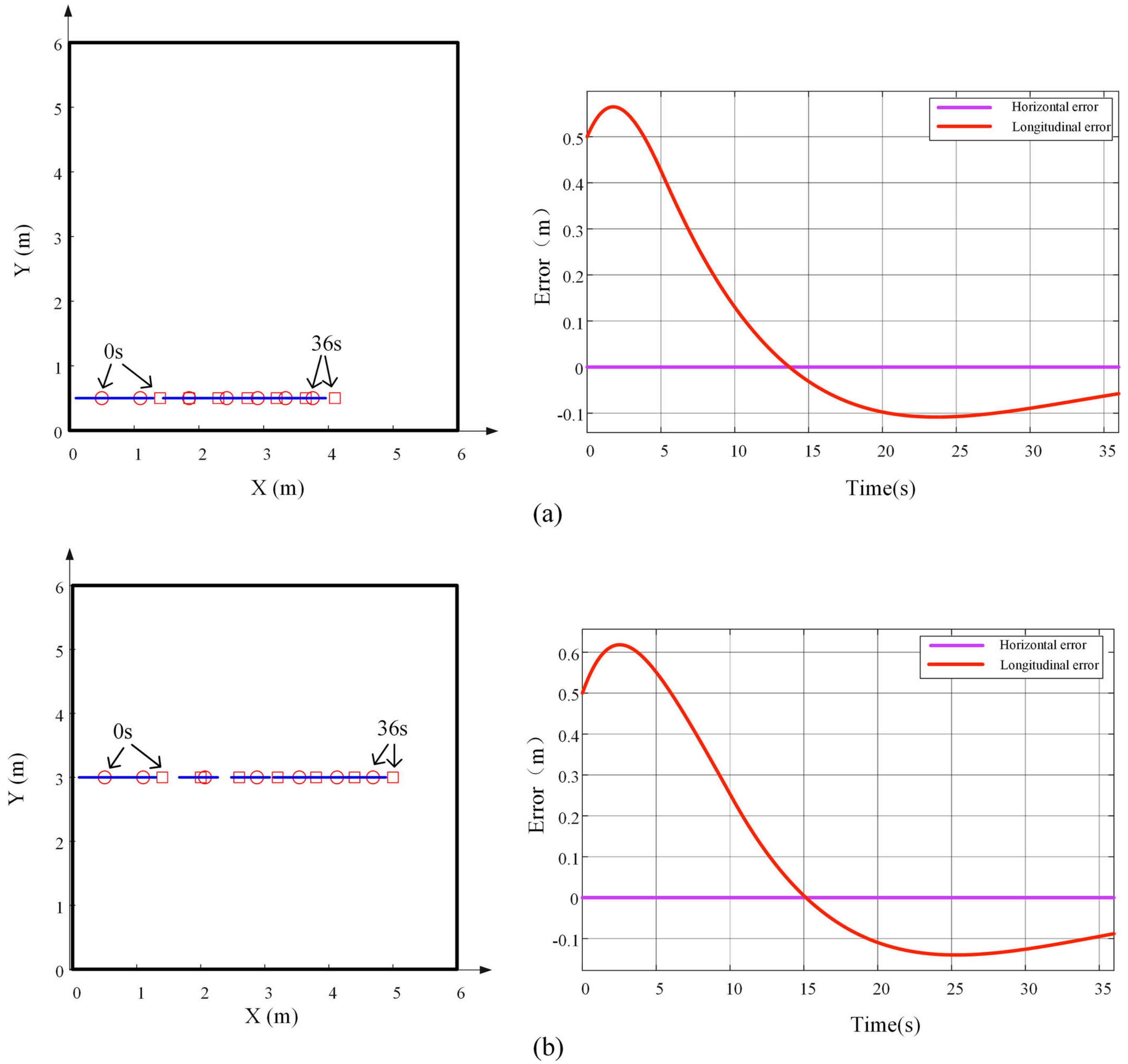


Figure 14. Longitudinal following simulation results.

results and a prior information, the horizontal and longitudinal error information between the desired tracking target and the body is estimated and outputted to the corresponding PID controller. Then, the PID controller calculates according to the configured PID gain coefficients and the error information and outputs control signals for the bionic mantis shrimp robot. Thus, it realises the robot's turning and acceleration/deceleration accurate motions in the water.

5. Simulation and experiments

5.1. Simulation of underwater target following system

Figure 13 shows the simulation model of the underwater target following control system applicable to the bionic mantis shrimp robot built in Matlab/Simulink. The control inputs of the motion model of the bionic mantis shrimp robot are the bending angle of the body and the oscillation frequency control signals of the pleopod and the outputs are the position of the robot in the world coordinate system as well as the heading angle.

The simulation tests are mainly divided into two parts. The first one is the longitudinal following simulation with an initial horizontal error of 0, and the second one is the horizontal and longitudinal direction coupled following simulation. In the longitudinal following simulation, the desired following error in the longitudinal direction is 0.2 m. The longitudinal coordinates of the model and the following target are the same and the following target is moving at a given constant velocity. Its turning control signal was set to 0 rad and the initial motion velocity, the initial body bending angle, and the initial heading angle of the model were all set to 0. The simulation time was set to 36s and the interval in the 2D following view was 6s. Figure 13 shows the simulation results.

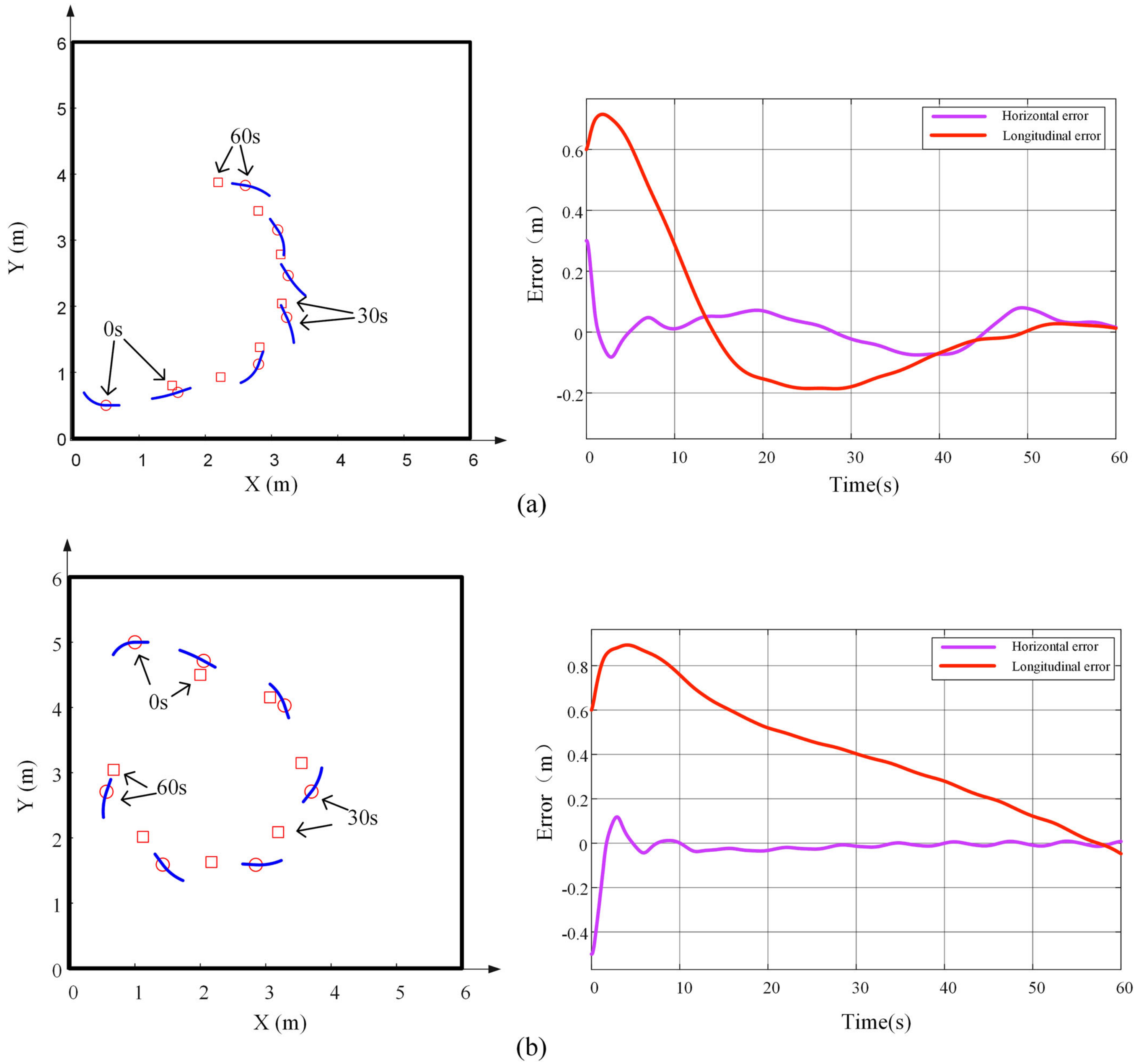


Figure 15. Horizontal and longitudinal direction coupled following simulation.

Figure 14(a) corresponds to the first set of simulations, where the initial coordinates of the robot body were (0.5, 0.5), the initial coordinates of the desired tracking target were (1.4, 0.5), and the motion velocity was 0.075 m/s. The left side of Figure 14(a) shows the two-dimensional following view, and the right side shows the horizontal and longitudinal error data. The horizontal error was always 0 and the longitudinal error decreased rapidly after the start of the motion. The maximum overshot error was about −0.12 m and then slowly tended to 0 m. The given desired following error was 0.2 m and the minimum distance between the robot and the target was about 0.08 m. Thus, no collision occurred, and the following task was successfully realised.

Figure 14(b) corresponds to the second set of simulations, where the initial coordinates of the robot body were (0.5, 3), the initial coordinates of the desired following target were (1.4, 3), and the motion velocity was 0.1 m/s. The left side of Figure 14(b) shows the two-dimensional following view, and the right side of Figure 14(b) shows the horizontal and longitudinal error data. The horizontal error was always 0 and the longitudinal error decreased rapidly after the start of the motion. The maximum overshoot error occurred at about −0.15 m and then slowly converged to 0 m. The given desired following error was 0.2 m and the minimum distance between the robot and the target was about 0.05 m. Thus, no collision occurred, and the following task was successfully realised.

In the horizontal and longitudinal direction coupled following simulation, the desired following error in the longitudinal direction was 0.2 m and the horizontal and vertical coordinates of the robot body and the following target were different. The following target moved at a given constant velocity and the turning control signal was set to be a sinusoidal signal with an amplitude of $\pi/6$, a frequency of 1 rad/s, and a centre of oscillation value of $\pi/7$. The initial velocity, the initial body bending angle, and the initial heading angle of the body model were all set to 0. The simulation

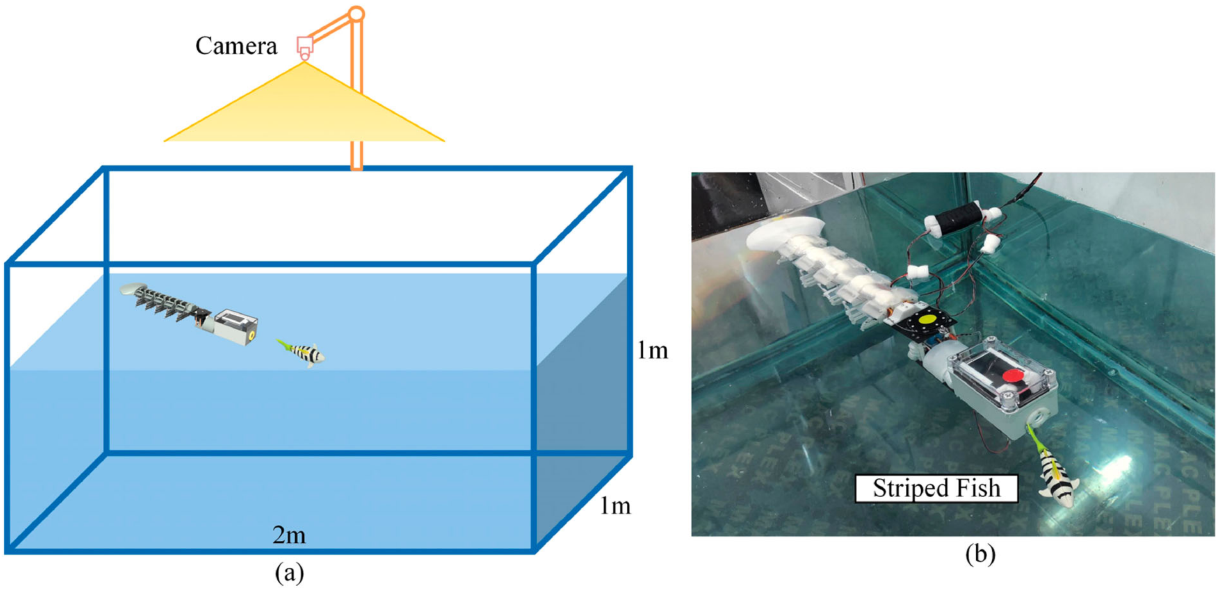


Figure 16. Experimental platform.

time was set to 60s and the plotting interval of the two-dimensional following view was 10s. The simulation results are shown in Figure 15.

The first set of simulations is shown in Figure 15(a), where the initial coordinates of the robot body are (0.5, 0.5), the initial coordinates of the desired following target are (1.5, 0.8), and the motion

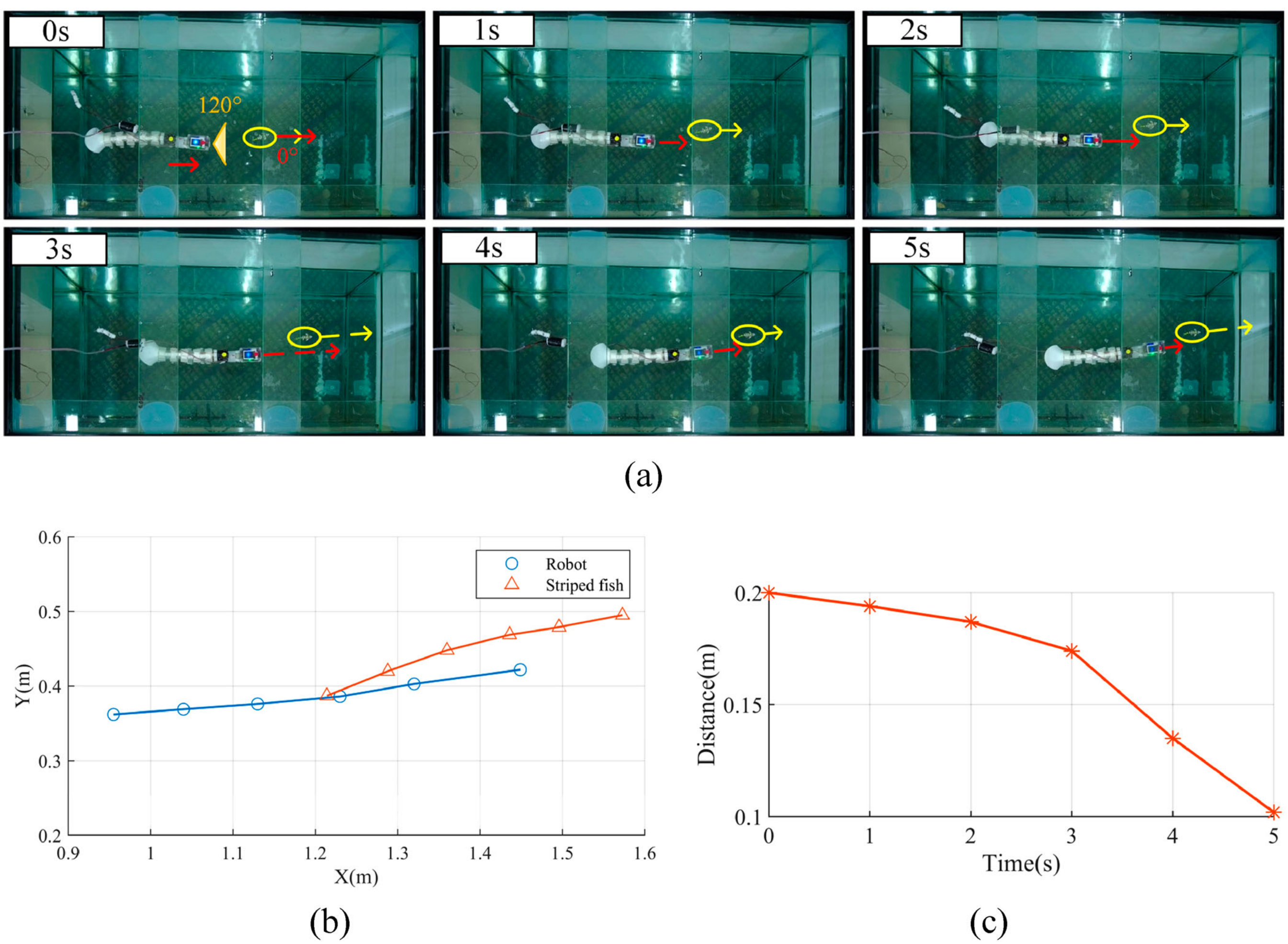


Figure 17. Robot straight following experiment: (a) Time series graph of robot motion, (b) motion trajectory of robot and tracking target and (c) distance of robot and tracking target.

velocity is 0.075 m/s. The left side of Figure 15(a) shows the 2D following view and the right side of Figure 15(a) shows the horizontal and longitudinal error data. The horizontal error was continuously adjusted from the initial value of 0.3 m to 0 m. The maximum overshot error was about −0.18 m and then towards 0 m. The given desired following error was 0.2 m and the minimum distance between the robot and the target was about 0.01 m. The following process can be successfully realised.

The second set of simulations is shown in Figure 15(b). The initial coordinates of the robot body were (1, 5), the initial coordinates of the desired following target were (2, 4.5), and the motion velocity was 0.115 m/s. The left side of Figure 15(b) shows the 2D following view and the right side of Figure 15(b) shows the data of the horizontal and longitudinal errors. The horizontal error quickly converged to 0 from the initial value of −0.5 m and only minor oscillation occurred. The longitudinal error started from 0.6 m and the maximum error was about 0.9 m and then slowly converged to 0 m. No collision occurred between the robot and the target during the process. The following process can be successfully realised.

The simulation results show that the robot does not collide with the target being followed, the following error is within an acceptable range, and the following task is successfully achieved. It verifies that the underwater target following system based on dual PID can output corresponding body control signals according to the lateral and longitudinal errors between the body and the desired target being followed, controlling the mantis shrimp robot to achieve underwater target following control.

5.2. Bionic mantis shrimp target following experiment

Figure 16(a) shows the robot experiment platform, and the pool size is 2 m × 1 m × 1 m. A camera is installed at the top of the pool to record the robot motion. The distance between the robot and the

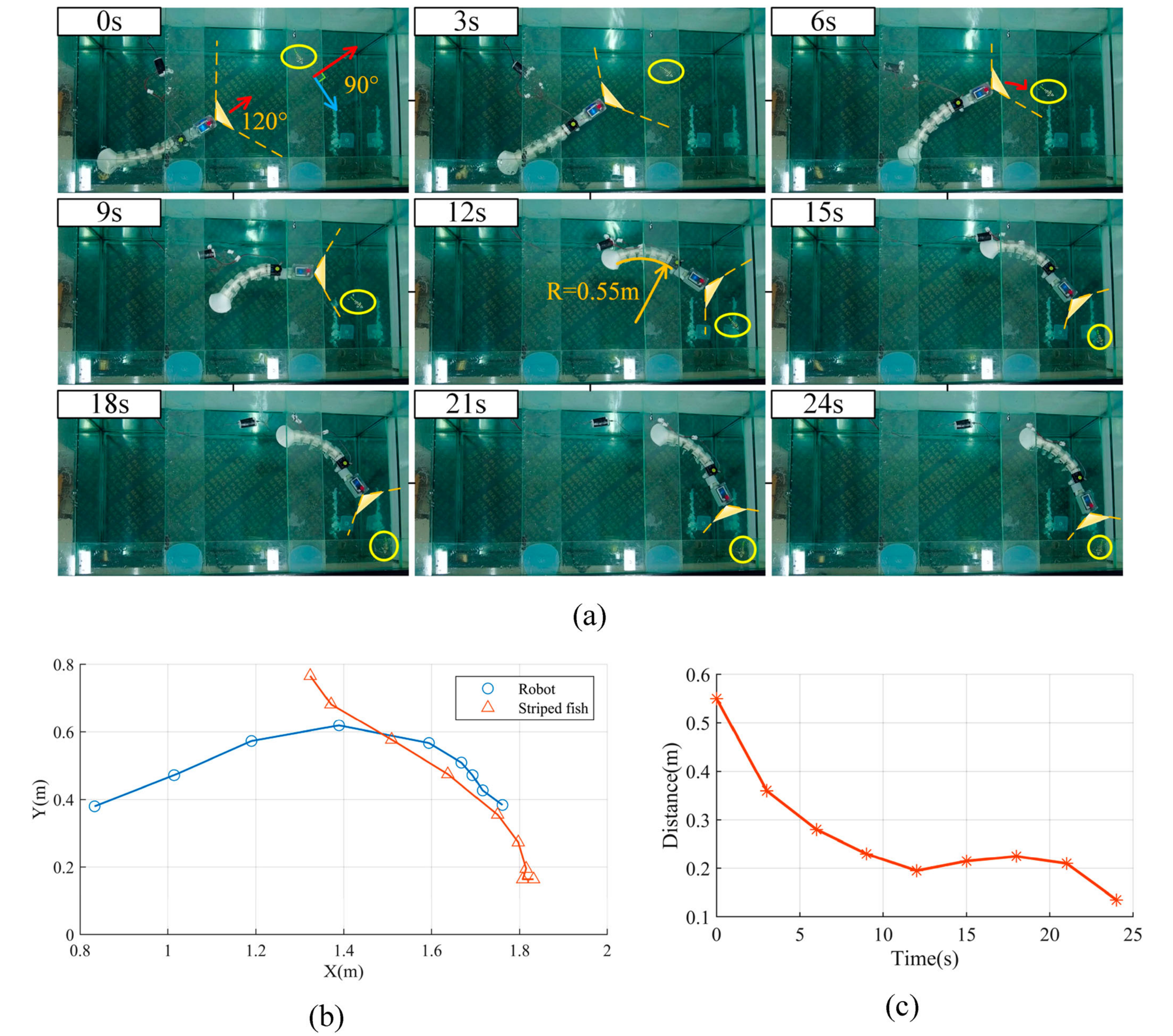


Figure 18. Turning following experiment: (a) time series graph of robot motion, (b) motion trajectory of robot and tracking target and (c) distance of robot and tracking target.

Table 2. Experimental results.

Type of experiment	Maximum velocity of the target	The angle of motion with the target	Average following velocity	Average velocity of target motion	Minimum turning radius
Dynamic straight following	0.15 m/s	0°	0.19 m/s	0.15 m/s	/
Dynamic turn following	0.15 m/s	90°	0.1 m/s	0.09 m/s	0.55 m

target as well as the motion trend can be obtained by annotating the recorded motion sequence diagram. Figure 15(b) shows our robot and the target striped fish (11 cm × 6 cm × 5 cm), and the motion velocity is about 0.15 m/s.

The following experiments of the bionic mantis shrimp robot were categorised into straight following and turning following. The target object striped fish was moved by manual remote control. Figure 17 shows the process that the robot follows the striped fish in a straight line, starting from the stationary state. Figure 17(a) shows the time series graph of the robot following movement in a straight line in 5 s. Figure 17(b) shows the trajectories of the robot and the striped fish. The Y movement of the striped fish deviated 0.09 m and the forward distance along the X-axis was 0.36 m within 5 s. The Y deviation of the robot was 0.07 m and the forward distance along the X-axis was 0.49 m. Both basically keep a motion direction. Figure 17(c) shows the variation of the distance between the striped fish and the robot where the robot keeps approaching the striped fish from 0.2 to 0.13 m.

Figure 18 shows the bionic mantis shrimp robot turning to follow the striped fish. The striped fish remained in the camera's field of view throughout the following. The robot and the striped fish started from the stationary state with the initial motion direction 90°. Figure 18(a) shows the time series graph of the robot turning to follow the striped fish. The robot adjusted the bending angle of the body many times to adjust the motion direction and the smallest turning radius was 0.55 m. Figure 18(b) shows the motion trajectory of the robot and the striped fish. The forward distance of the striped fish along the X-axis in the turning following motion was 0.51 m, and the forward distance of the robot along the X-axis was 0.93 m. Figure 18(c) presents the distance change between the striped fish and the robot from 0.55 to 0.13 m. The distance changed 0.42 m and the following motion was realised. The farther away the distance was, the faster the robot's motion was and the closer it got, the slower its motion velocity was.

Table 2 summarises the key parameters of the experiment. The average velocity of the bionic mantis shrimp robot in dynamic linear following is 0.19 m/s, and the average moving velocity of the target is 0.15 m/s. In the dynamic turning following experiment, the average velocity of the robot is 0.1 m/s, the average moving velocity of the target is 0.09 m/s, and the minimum turning radius is 0.55 m. Combined with the motion time series diagram of the robot, it can be seen that the swimming velocity of the robot slows down as the distance from the target decreases, and the robot can adjust its own motion in real time according to the motion state of the target.

6. Conclusions and future work

In this study, a novel bionic mantis shrimp robot was designed and constructed based on the structure and movement mode of the biological mantis shrimp. For the task of underwater target following, the KMY target tracking algorithm was proposed and verified through experiments, which demonstrated good tracking performance and can continuously track the mis-detected target. In addition, a PID-based target following system was developed for the bionic mantis shrimp robot. The effectiveness of the longitudinal following strategy, as well as the horizontal and longitudinal direction coupled following strategy, were verified by simulations. Finally, the straight and turning following movement to the striped fish was completed through experiments. The experimental results show that with the target recognition and proposed following control system, the bionic mantis shrimp robot can quickly recognise and follow a specific underwater target.

In future research, we will upgrade hardware, including cameras, visual development boards to enhance the target recognition performance of robots in complex underwater environments such as low light and water flow disturbances. Additionally, we will optimise the robot's motion controller to improve its disturbance resistance and motion performance, providing assistance for marine scientific research, underwater exploration, and resource development.

Funding

This work was financially supported by National Natural Science Foundation of China [grant number 52275037], [grant number 51875528], and [grant number 41506116], Zhejiang Provincial Natural Science Foundation of China [grant number LR24E050002], the Key Research and Development Project of Zhejiang Province [grant number 2023C03015], the Key Research and Development Project of Ningxia Hui Autonomous Region [grant number 2023BDE03002].

Notes on contributors

Gang Chen received the Ph.D. degree in mechatronic engineering from the State Key Laboratory of Fluid Power and Mechatronic Systems, Zhejiang Univer-sity, Hangzhou, China, in 2014. He is currently a professor with the School of Mechatronic Engineering, Zhejiang Sci-Tech University, Hangzhou, China, specialising in intelligent control of bionic robots.

Yidong Xu received the bachelor's degree in engineering from Ningbo University of Technology, Ningbo, China, in 2021. He is currently a doctoral student at Zhejiang Sci-Tech University. His main research direction is under-water bionic robots.

Jiajun Tu received the master's degree in mechanical engineering from Zhejiang Sci-Tech University, Hangzhou, China, in 2021. He is currently a teacher in Intelligent Manufacturing College at Jiaxing Vocational and Technical College. His main research direction is bionic robotics.

Xin Yang is a master student at Zhejiang Sci-Tech University. His main research direction is underwater soft robots and underwater bionic robots. He received his bachelor's degree from China University of Mining and Technology in 2019.

Huosheng Hu received an M.Sc. degree in industrial automation from Central South University, Changsha, China, in 1982, and a Ph.D. degree in robotics from the University of Oxford, Oxford, U.K., in 1993. His current research interests include robotics, human-robot interaction, embedded systems, mechatronics, and pervasive computing.

Fei Dong received a master's degree in industrial design engineering from Donghua University, Shanghai, China, in 2015. She is currently a CTO of Zhejiang Chengshi Robot Co., Ltd. and has been engaged in special robot research for 8 years.

Jingjing Zhang is the VP of ZheJiang CathayBot Technology Co.,Ltd. He has been working on special robot research for over 13 years. He received his master's degree in Guangxi Minzu University.

Jianwei Shi is a RD engineer at Zhejiang Cathaybot Technology Co., Ltd., who is responsible for the hardware research and development of hydraulic robotic arm control system. He graduated from Southwest Jiaotong University with a major in electrical automation in 1998 and has been working on hardware and software for automatic microcontrollers for over 20 years.

References

Al-Haija QA, Smadi M, Al-Bataineh OM. 2021. Identifying phasic dopamine releases using DarkNet-19 convolutional neural network. 2021 IEEE International IoT, Electronics and Mechatronics Conference (IEMTRONICS). IEEE. p. 1–5.

Azimi-Sadjadi MR, Yao D, Huang Q, Dobeck Gerald J. 2000. Underwater target classification using wavelet packets and neural networks. IEEE Trans Neural Netw. 11:784–794. doi:10.1109/72.846748.

Burrows M, Hoyle G. 1972. Neuromuscular physiology of the strike mechanism of the mantis shrimp, *Hemisquilla*. J Exp Zool. 179:379–393. doi:10.1002/jez.1401790309.

Chen G, Han Y, Li Y, Shen J, Tu J, Yu Z, Zhang J, Cheng H, Zhu L, Dong F. 2024a. Autonomous gait switching method and experiments of a hexapod walking robot for Mars environment with multiple terrains. Intel Serv Robot. 1–21.

Chen G, Peng W, Wang Z, Tu J., Hu H, Wang D, Cheng H, Zhu L. 2023a. Modeling of swimming posture dynamics for a beaver-like robot. Ocean Eng. 279:114550. doi:10.1016/j.oceaneng.2023.114550.

Chen G, Xu Y, Yang C, Yang X, Hu H, Chai X, Wang D. 2023b. Design and control of a novel bionic mantis shrimp robot. IEEE/ASME Trans Mechatron. 28:3376–3385.

Chen G, Xu Y, Yang X, Hu H, Cheng H, Zhu L, Zhang J, Shi J, Chai X. 2024b. Target tracking control of a bionic mantis shrimp robot with closed-loop central pattern generators. Ocean Eng. 297:116963. doi:10.1016/j.oceaneng.2024.116963.

Chen G, Yang X, Xu Y, Lu Y, Hu H. 2022. Neural network-based motion modeling and control of water-actuated soft robotic fish. Smart Mater Struct. 32:15004. doi:10.1088/1361-665X/aca456.

Chen G, Zhao Z, Wang Z, Tu J, Hu H. 2023c. Swimming modeling and performance optimization of a fish-inspired underwater vehicle (FIUV). Ocean Eng. 271:113748. doi:10.1016/j.oceaneng.2023.113748.

Chui CK, Chen G. 2017. Kalman filtering. Springer.

Grewal MS, Andrews AP. 2014. Kalman filtering: theory and practice with MATLAB. California: John Wiley & Sons.

Grewal MS, Andrews AP, Bartone CG. 2020. Kalman filtering.

Honghui Y, Junhao L, Meiping S. 2022. Underwater acoustic target multi-attribute correlation perception method based on deep learning. Appl Acoust. 190:108644. doi:10.1016/j.apacoust.2022.108644.

Jian M, Liu X, Luo H, Lu X, Yu H, Dong J 2021. Underwater image processing and analysis: A review. Signal Process, Image Commun. 91:116088. doi:10.1016/j.image.2020.116088.

Jung MR, Horgen FD, Orski SV, et al. 2018. Validation of ATR FT-IR to identify polymers of plastic marine debris, including those ingested by marine organisms. Mar Pollut Bull. 127:704–716. doi:10.1016/j.marpolbul.2017.12.061.

Kiuru P, D'auria MV, Muller CD, Tammela P, Vuorela H, Yli-Kauhaluoma J. 2014. Exploring marine resources for bioactive compounds. Planta Med. 80:1234–1246. doi:10.1055/s-0034-1383001.

Li X, Qin Y, Wang F, et al. 2020. Pitaya detection in orchards using the MobileNet-YOLO model2020 39th Chinese Control Conference (CCC). IEEE. P. 6274–6278.

Li Y, Huang H, Xie Q, Yao L, Chen Q. 2018. Research on a surface defect detection algorithm based on MobileNet-SSD. Appl Sci. 8:1678. doi:10.3390/app8091678.

Mackenzie K. 2002. Parasites as biological tags in population studies of marine organisms: an update. Parasitology. 124:153–163. doi:10.1017/S0031182002001518.

Patek SN, Korff WL, Caldwell RL. 2004. Deadly strike mechanism of a mantis shrimp. Nature. 428:819–820. doi:10.1038/428819a.

Patrón P, Petillot Y. 2008. The underwater environment: A challenge for planning. Edinburgh: UK PlanSIG.

Porte C, Janer G, Lorusso LC, Ortiz-Zarragoitia M, Cajaraville MP, Fossi MC, Canesi L. 2006. Endocrine disruptors in marine organisms: approaches and perspectives. Comp Biochem Physiol C Toxicol Pharmacol. 143:303–315. doi:10.1016/j.cbpc.2006.03.004.

Prasad B, Agrawal A, Viswanathan V, et al. 2015. A visually guided spherical underwater robot. 2015 IEEE Underwater Technology (UT). IEEE. P. 1–6.

Redmon J, Farhadi A. 2017. YOLO9000: better, faster, stronger. Proceedings of the IEEE conference on computer vision and pattern recognition, 20177263-7271.

Sattar J, Dudek G. 2006. On the performance of color tracking algorithms for underwater robots under varying lighting and visibility. Proceedings 2006 IEEE International Conference on Robotics and Automation. ICRA 2006. IEEE. P. 3550–3555.

Shafi I, Mazahir A, Fatima A, Ashraf I. 2022. Internal defects detection and classification in hollow cylindrical surfaces using single shot detection and MobileNet. Measurement (Mahwah NJ). 202:111836. doi:10.1016/j.measurement.2022.111836.

Shi L, Bao P, Guo S, Chen Z, Zhang Z. 2022. Underwater formation system design and implement for small spherical robots. Ieee Syst J. 17:1259–1269. doi:10.1109/JSYST.2022.3199574.

Steinhardt E, Hyun NP, Koh J, Freeburn G, Rosen MH, Temel FZ, Patek SN, Wood RJ. 2021. A physical model of mantis shrimp for exploring the dynamics of ultrafast systems. Proc Natl Acad Sci USA. 118:e2026833118. doi:10.1073/pnas.2026833118.

Teng B, Zhao H. 2020. Underwater target recognition methods based on the framework of deep learning: A survey. Int J Adv Robot Syst. 17:1738133683. doi:10.1177/1729881420976307.

Wang X, Liu A, Zhang Y, Xue F. 2019. Underwater acoustic target recognition: a combination of multi-dimensional fusion features and modified deep neural network. Remote Sens-Basel. 11:1888. doi:10.3390/rs11161888.

Wei Y, Zhu D, Chu Z. 2018. Underwater dynamic target tracking of autonomous underwater vehicle based on MPC algorithm. 2018 IEEE 8th International Conference on Underwater System Technology: Theory and Applications (USYS). IEEE. p. 1–5.

Weitbrecht V, Kühn G, Jirka GH. 2002. Large scale PIV-measurements at the surface of shallow water flows. Flow Meas Instrum. 13:237–245. doi:10.1016/S0955-5986(02)00059-6.

Williams DP, Fakiris E. 2014. Exploiting environmental information for improved underwater target classification in sonar imagery. Ieee T Geosci Remote. 52:6284–6297. doi:10.1109/TGRS.2013.2295843.

Yuan X, Guo L, Luo C, Zhou X, Yu C. 2022. A survey of target detection and recognition methods in underwater turbid areas. Applied Sciences. 12:4898. doi:10.3390/app12104898.

Yuh J, West M. 2001. Underwater robotics. Adv Robotics. 15:609–639. doi:10.1163/156855301317033595.

Zhao S, Yuh J. 2005. Experimental study on advanced underwater robot control. Ieee T Robot. 21:695–703. doi:10.1109/TRO.2005.844682.