

Position Control for Underwater Vehicle Using Image Processing

Chikara Masuzaki,^{1*} Ikuo Yamamoto,^{1,2} Akihiro Morinaga,¹
Koudai Sadano,¹ Yuto Kai,¹ Tubasa Nakano,¹ and Yuya Kato¹

¹Graduate School of Engineering, Nagasaki University, 1-14 Bunkyo-machi,
Nagasaki City, Nagasaki 852-8521, Japan

²Organization for Marine Science and Technology, Nagasaki University, 1-14 Bunkyo-machi,
Nagasaki City, Nagasaki 852-8521, Japan

(Received December 15, 2020; accepted February 22, 2021)

Keywords: underwater vehicle, image processing, PID control, position control

In this paper, we propose a method of position control for an underwater vehicle using a monocular camera. The proposed method does not require an expensive inertial measurement unit (IMU) and uses template matching to track the underwater target and calculate the amount of vehicle movement. The vehicle is then controlled on the basis of the amount of movement. In this paper, we compare the typical similarity calculation methods of template matching, sum of squared differences (SSD), normalized cross-correlation (NCC), and zero-mean normalized cross-correlation (ZNCC), and investigate which has the greatest accuracy underwater. We also compare proportional (P) control and proportional–integral–derivative (PID) control as vehicle control methods. The results showed that ZNCC was the most accurate in detecting the target and that PID control was about 30% more accurate than P control.

1. Introduction

Marine energy projects, such as offshore wind power and tidal power projects, are currently attracting considerable attention. However, there are several challenges in promoting these projects. Underwater facilities are inspected and repaired by divers in the water, but this is highly hazardous and expensive, and only short working periods are possible. Underwater vehicles are being used to address these issues.

In the past, various underwater vehicles have been developed.^(1,2) These vehicles are equipped with a variety of sensors to measure the depth and direction of the vehicle and its position with acoustic positioning devices to stabilize the vehicle and ensure its operability. These devices are generally very expensive, and thus they are not suitable for a small light underwater vehicle utilized as a commercial product for maintaining offshore facilities in shallow water. Control of a vehicle based on images has also been considered.⁽³⁾ Such an underwater vehicle is equipped with laser triangulation optical correlation sensors mounted on the vehicle and facing downward. These measure the horizontal position and velocity and

*Corresponding author: e-mail: bb52119139@ms.nagasaki-u.ac.jp
<https://doi.org/10.18494/SAM.2021.3223>

the distance from the seabed. This control does not have position control relative to the target object, and the amount of movement is calculated from the lights on the vehicle. The Kanade–Lucas–Tomasi (KLT) feature tracker⁽⁴⁾ is often used in a unmanned aerial vehicle (UAV). However, this method requires feature points and is unsuitable in the sea, where there are floating objects. It is also unsuitable in the sea because the brightness value changes and the feature points are vulnerable to large displacements.⁽⁵⁾ A method of control based on the shape of the underwater station visible from the vehicle has also been proposed.⁽⁶⁾

The motivation of this study is to maintain the relative distance from a target in front of the vehicle using a monocular camera equipped on a small underwater vehicle. This allows vehicle pilots to inspect underwater structures to perform their mission even in the presence of tidal currents.

In this study, we propose a control method for underwater vehicles, which uses template matching. The flow of the proposed method is as follows. First, the target is detected by template matching. Second, the displacement between the center of the detected target and the center of the vehicle image is calculated. Third, the vehicle is controlled using the calculated displacement. The proposed method makes it possible to maintain the vehicle in front of the target object. Also, the proposed method uses only a single camera as a sensor, so the vehicle can be made small and inexpensive. The proposed method is tracked by using template matching. Template matching is considered to be a powerful image processing tool that is particularly useful in industrial inspection problems.⁽⁷⁾ The advantage of template matching is that the pilot can freely choose the target.

The paper is organized as follows. Section 2 describes the underwater vehicle developed, Sect. 3 describes the algorithm created, the template matching employed, the reasoning behind it, and the control method of the underwater vehicle, and Sect. 4 presents the experiment and its results. Section 5 presents a conclusion.

2. Underwater Vehicle

This section describes the hardware and the system configuration of the underwater vehicle, which is named CAIBOT.

2.1 Overview

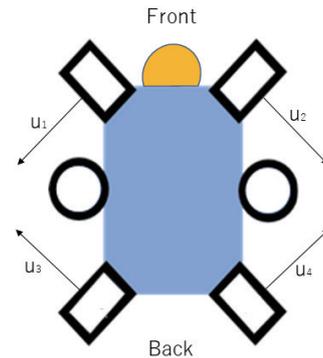
The overall image of the vehicle used is shown in Fig. 1 and the thruster arrangement is shown in Fig. 2. The specifications are given in Table 1.

In this vehicle, four of the thrusters are used to move forward, reverse, and laterally and allow pivoting, and the other two are used for vertical movement. The symmetrical position of the former four thrusters allows for the smooth distribution of the control action of the actuators.⁽³⁾ The body size is 490 mm [L] × 420 mm [W] × 160 mm [H], which is very small compared with that in related studies. The thrusters operate by pulse width modulation (PWM).

The camera used in this study and its specifications are shown below (Fig. 3). The purpose of this research is to develop a vehicle that can inspect underwater structures. In particular, the camera is a very important element because image processing is performed using camera



Fig. 1. (Color online) CAIBOT.

Fig. 2. (Color online) Thruster locations: u_1 , u_2 , u_3 , u_4 are thrust forces.Table 1
Specifications of CAIBOT.

Dimensions	490 [L] × 420 [W] × 160 [H]	mm
Weight	6.9	kg
Cable length	60	m
Material	Aluminum	
Submersible depth	40	m
Sailing speed	1.5	m/s
T100 thruster		
Thruster	Operating voltage	6–16 V
	Max power	135 W
	Maximum forward thrust	2.36 kg f
	Maximum reverse thrust	1.85 kg f
	Rotational speed	300–4200 rpm

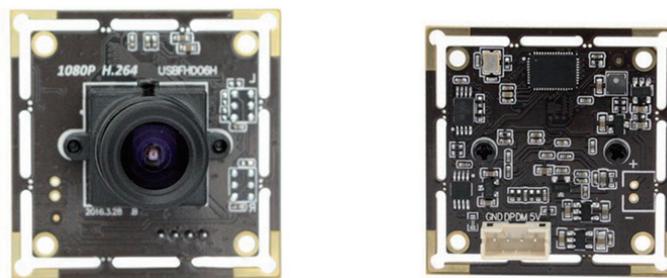


Fig. 3. (Color online) USB camera.

images. The specifications of the camera are H264: 640×480 , 30 fps, 80° (horizontal), 64° (vertical), $5.0 \text{ V/lux-s}@550 \text{ nm}$ (sensitivity).

2.2 System configuration

The system diagram of the vehicle is shown in Fig. 4. The vehicle uses Ethernet communication between the vehicle and the ground station. The communication cable is a two-core cable instead of an Ethernet cable. This is because a two-core cable is smaller in diameter

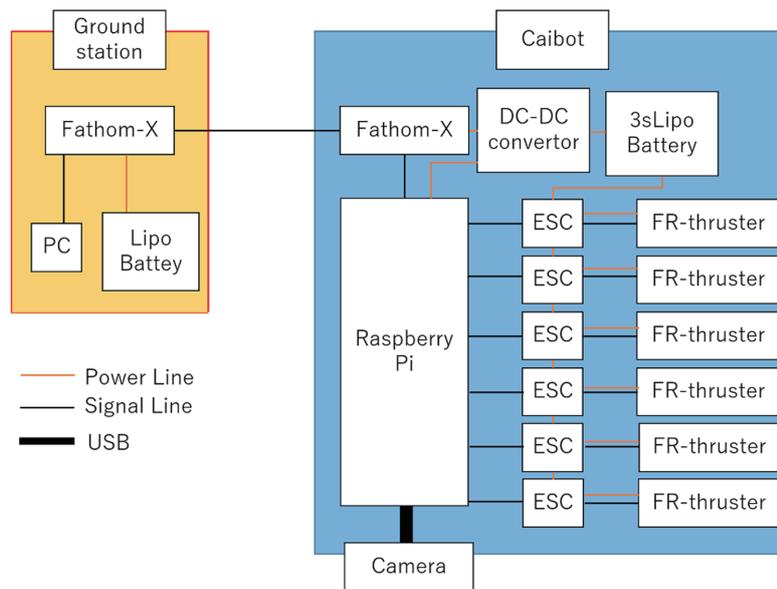


Fig. 4. (Color online) System configuration of CAIBOT and ground station.

than an Ethernet cable, so it is less susceptible to the effects of waves and currents and does not affect the thrust, control, and operation of the vehicle when moving through the water. This vehicle uses FATHOM-X-SINGLE-R1-RP as the Ethernet connection to a two-core conversion module.

3. Proposed Control System

This section describes the proposed positioning control method for the underwater vehicle.

3.1 Control flow

The purpose of the position control is to maintain the target in front of the camera in a tidal stream. A flowchart of the proposed position control method is shown in Fig. 5.

The flowchart in Fig. 5 is divided into three stages.

- (a) Capture images from a monocular camera and transmit them to a ground station.
- (b) Image processing stage: In this phase, the target is selected and detected by template matching, and the calculated movement of the target object is sent to a Raspberry Pi device.
- (c) Convert the displacement to the pulse width and output.

In (a), the images were captured in the vehicle and transmitted to the ground station. The images captured by the camera were encoded into bytes by a Raspberry Pi 4 and decoded by the base station. This was done by installing OpenCV on the Raspberry Pi 4 and the base station.

In (b), the amount of movement is calculated from the displacement between the center point of the recognized target object and the center point of the captured image by template matching. This is used to control the thruster so that the target object is always displayed in the center of the image.

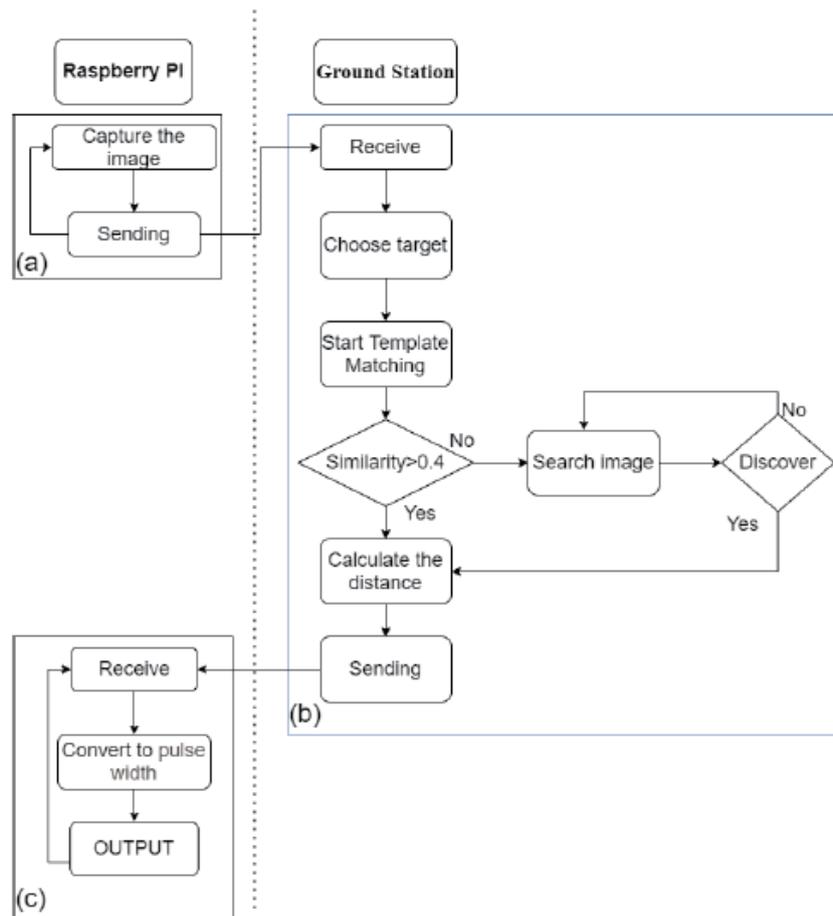


Fig. 5. Flowchart of proposed control method.

3.2 Template matching

Template matching is a type of pattern recognition⁽⁸⁾ and has been used for vehicle type recognition in automated driving.⁽⁹⁾ Normally, the main issues of template matching are illumination, scale changes, robustness, high computational cost, and accuracy.⁽¹⁰⁾ The reason for the high computational cost is that the template image is probed from the entire frame. Therefore, we improved the processing speed by performing a partial search in the region of the outer frame of the two frames, as shown in Fig. 6. The inner frame is the detected target object. We focus on robustness and accuracy because the luminance values and color of the input image change underwater. Therefore, we tried to improve them by investigating similarity calculations for template matching that are effective underwater. Typical similarity calculations are the sum of squared differences (SSD), normalized cross-correlation (NCC), and zero-mean normalized cross-correlation (ZNCC). The features of each algorithm and the calculation formulae are described below. In this study, template matching was performed using OpenCV. The following equations are used to calculate the similarity measures compared in this study.⁽¹¹⁾



Fig. 6. (Color online) Vehicle perspective. The outer blue square is the search area and the inner green square is the search window.

- SSD: This method performs matching by squaring the difference. Therefore, the value is 0 for a match, and the value increases with increasing displacement.

$$R(x, y) = \sum \sum [T(x + x', y + y') - I(x', y')]^2 \quad (1)$$

- NCC: Variations in luminance values can be absorbed by calculating the cross-correlation between the pixel values of the input image and the template image.

$$R(x, y) = \frac{\sum \sum [T(x + x', y + y') I(x', y')]}{\sqrt{T(x + x', y + y')^2 \sum \sum I(x', y')^2}} \quad (2)$$

- ZNCC: The cross-correlation coefficients are calculated as a statistic by considering the pixel values of the input image and the template image as the density distribution in the image area. Because the average value is subtracted, it is more robust to changes in brightness than NCC.

$$R(x, y) = \frac{\sum \sum [(T(x + x', y + y') - \bar{T})(I(x', y') - \bar{I})]}{\sqrt{\sum \sum (T(x + x', y + y') - \bar{T})^2 \sum \sum (I(x', y') - \bar{I})^2}} \quad (3)$$

In Eqs. (1)–(3), R denotes the result, T denotes the input image, and I denotes the template image. Also, \bar{T} and \bar{I} represent average luminance values in the region.

3.3 Experiment on template matching speed

We carried out an experiment to determine which of the template matching algorithms was best suited for use underwater. In this experiment, 10 frames were randomly extracted from

the input video and evaluated. In these 10 frames, the positions of the target and the camera were not moved but the way the light hit the target was changed. The target object was cut out from the first input image, and template matching was performed using it as a template image. The original target object is shown in Fig. 7. Examples of input images and template images are shown in Fig. 8. In this experiment, four patterns were tested: [1] no light illuminated the target object, [2] part of the target object was illuminated, [3] light illuminated the target object from overhead, and [4] light illuminated the target object from the front. Figure 9 shows the positions of the camera, light, and target. All lights were turned off except for those used in the experiment, and the experimental pool was covered.

The method used to evaluate the tracking performance was based on the difference between the actual center point of the target and the detected center point of the frame. The processing time of each pattern was also measured and compared. The results are shown in Tables 2 and

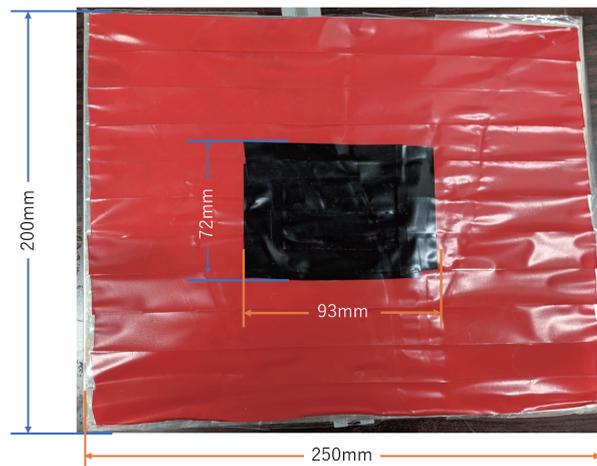


Fig. 7. (Color online) Target object.

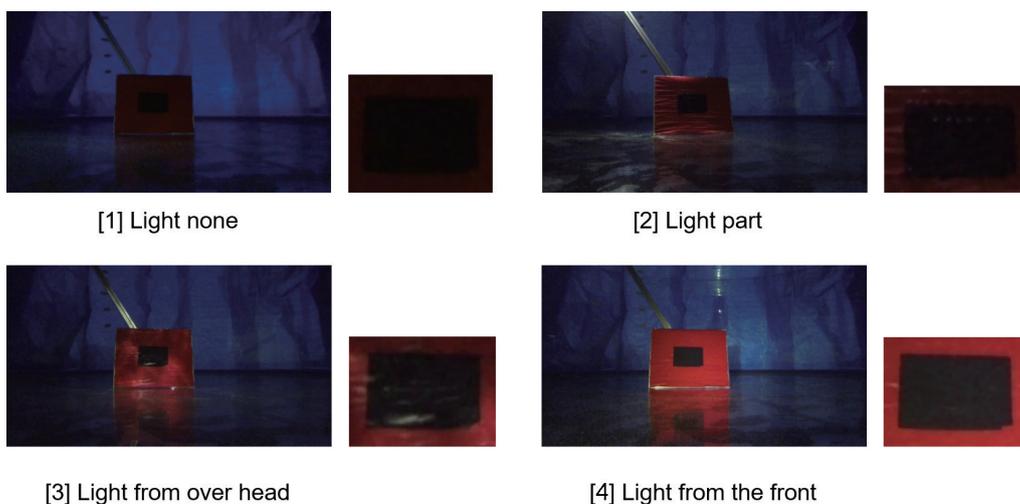


Fig. 8. (Color online) Experimental setup used in this study.

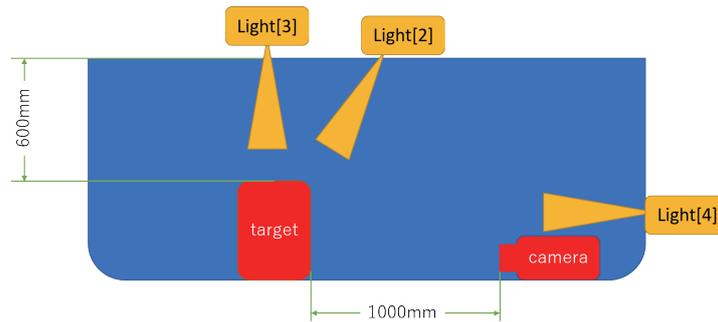


Fig. 9. (Color online) Positions of the camera, light, and target.

Table 2
Average processing times.

	[1] (ms)	[2] (ms)	[3] (ms)	[4] (ms)
SSD	43.532	45.852	45.304	43.742
NCC	42.768	44.887	43.926	47.418
ZNCC	53.809	56.584	52.638	56.657

Table 3
Average displacements between center points.

	[1] (pixel)	[2] (pixel)	[3] (pixel)	[4] (pixel)
SSD	184.26	279.21	326.33	194.85
NCC	105.11	2.38	29.40	40.56
ZNCC	105.07	2.20	2.20	0.49

Table 4
Variances of displacements in Table 3.

	[1]	[2]	[3]	[4]
SSD	3.25	2059.52	1796.25	12454.24
NCC	0.23	11.7076	7546.04	14467.77
ZNCC	0.29	11.67	43.65	0.31

3, where the units in Table 2 are in seconds and those in Table 3 are in pixels. The top left corner of the image is the origin of the coordinates, the rightward direction is the x -axis, and the downward direction is the y -axis. In this experiment, template matching was done by a full survey rather than a partial survey.

From Table 2, ZNCC has the longest processing time. This may be because the calculation method of ZNCC is more complicated than the other similarity calculation methods. This problem can be solved by using a partial survey instead of a full survey. It can be seen that the detection performance of all the similarity calculation methods is low in the case of no light. Even under the lighted conditions, the displacement between the center of the detected target and the center of the camera frame was the greatest for SSD. Table 4 shows the variance of the displacements in Table 3. There is no significant difference in the detection performance and the variance of the displacement to the center of the target between NCC and ZNCC when there is no light or when the light hits a part of the target. However, when there is a lot of light [3], [4], the detection performance of ZNCC is better than that of NCC. The variance for ZNCC is also smaller than that for NCC, indicating that the rate of false detection is less than that of NCC. Therefore, ZNCC is used as a similarity calculation method in this study.

3.4 Tracking control

In Fig. 5(c), the pulse width is converted from the displacement using Eqs. (4)–(7), where Eqs. (4) and (5) represent proportional (P) control and Eqs. (6) and (7) represent proportional–integral–derivative (PID) control. In this paper, the gains are K_P , K_I , and K_D and the displacement is d . $T_{Neutral}$ represents the neutral value of the thruster used in the experiment, and T_{u1} , T_{u2} , T_{u3} , and T_{u4} are the pulse widths of thruster outputs u_1 , u_2 , u_3 , and u_4 , respectively.

$$T_{u1}, T_{u3} = T_{Neutral} + K_P d \quad (4)$$

$$T_{u2}, T_{u4} = T_{Neutral} - K_P d \quad (5)$$

$$T_{u1}, T_{u3} = T_{Neutral} + \left(K_P d + K_I \dot{d} + K_D \int_t^{t_0} d dt \right) \quad (6)$$

$$T_{u2}, T_{u4} = T_{Neutral} - \left(K_P d + K_I \dot{d} + K_D \int_t^{t_0} d dt \right) \quad (7)$$

Note that d varies in value depending on the location of the thrusters. For the thrusters at the corners of the body, the horizontal displacement is assigned, and for the thrusters in the center of the body, the vertical displacement is assigned.

4. Experimental Results and Discussion

On the basis of the results in Sect. 3.3, ZNCC was implemented on the vehicle. K_P in Eqs. (4) and (5) was set to 0.5 for the horizontal thrusters and to 0.7 for the vertical thrusters. K_P , K_I , and K_D in Eqs. (6) and (7) were set to 0.2, 0.1, and 0.2 for the horizontal thrusters and to 0.3, 0.3, and 0.3 for the vertical thrusters, respectively. These values were determined by trial and error.

The target was placed in the upper right corner of the camera frame and the algorithm was started when the vehicle reached the bottom of the test pool. The experiment is shown in Fig. 10, and Fig. 11 shows the template images used in the experiment. The relationship between

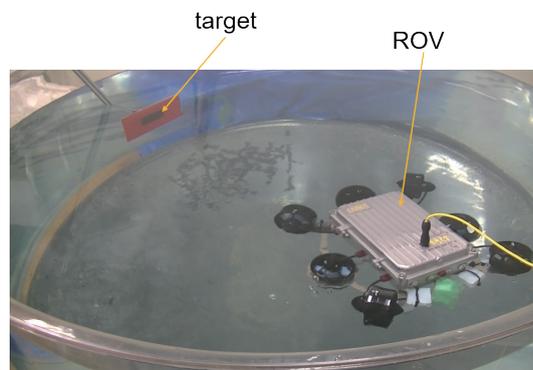


Fig. 10. (Color online) Scene of the experiment.

the vehicle, target, and the experiment pool is shown in Fig. 12. The results are shown in Fig. 13. The average horizontal and vertical displacements were derived from Eqs. (8) and (9), respectively. The results are shown in Table 5.

$$\frac{\sum_{N=1}^N |Horizontal|}{N} \tag{8}$$

$$\frac{\sum_{N=1}^N |Vertical|}{N} \tag{9}$$

Figure 13 shows the center point of the object to be tracked in a 640 × 480 camera frame and Table 5 shows the average displacement between the center point of the object and the center of the camera frame. In Fig. 13, the closer the plot is to the center, the more accurate the tracking is. Figures 13(a) and 13(b) show the results for PID control and P control, respectively, revealing

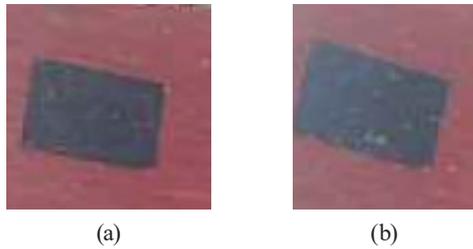


Fig. 11. (Color online) Template images: (a) PID-control template image. The image size is (76, 77). (b) P-control template image. The image size is (68, 77).

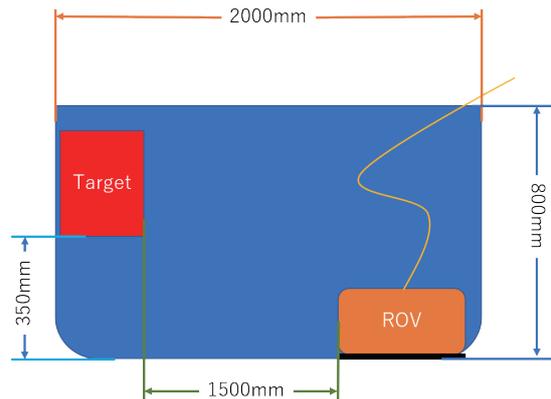


Fig. 12. (Color online) Relationship between the vehicle, the target, and the experiment pool.

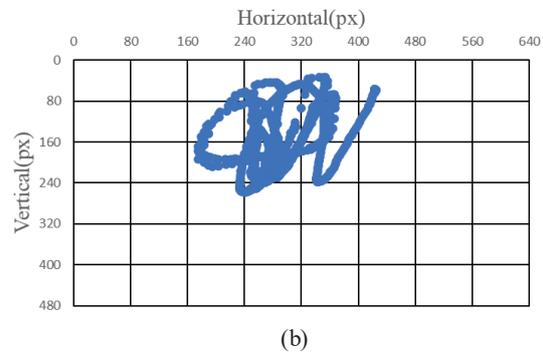
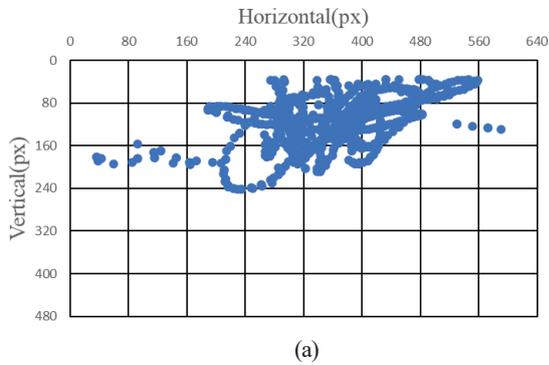


Fig. 13. (Color online) Trajectories of the target in the captured frame: (a) PID-control trajectory and (b) P-control trajectory.

Table 5

Average displacement between the center point of the object and the center of the camera frame.

	P control (pixel)	PID control (pixel)
Average horizontal displacement	74.18	52.32
Average vertical displacement	124.87	86.92

that PID control is more tightly distributed. However, in both cases, the target remains above the vehicle. This may be related to the fact that the vehicle started from the bottom of the pool. It is considered that this problem would be resolved if the vehicle started in a neutral buoyant state.

Table 5 shows the average displacement between the center of the target object and the center of the camera frame. It shows that PID control is clearly superior for both horizontal and vertical displacements, which have been reduced by about 30%.

5. Conclusion

We have studied the position control of a small underwater robot using template matching. We have confirmed that ZNCC, which is robust to changes in luminance, is the most suitable similarity calculation method for template matching. Using this method, the target object is detected, and position control is possible by controlling the output of the thrusters based on the displacement between the detected target object and the center point of the camera frame. As a result of comparing P control and PID control as the control method, we confirmed that the accuracy of PID control is about 30% better than that of P control.

The problem in the experiment was the control in the depth direction and the attitude. When the vehicle was moving, there was a limit to how much the buoyancy force could be adjusted to maintain the attitude, and the vehicle sometimes moved in a tilted position. In this case, the vehicle was often too close to the target object. Therefore, we are considering solving this problem by acquiring depth direction information from images and incorporating it into the control system. It is known that there are various restrictions on image processing in water.⁽¹²⁾ We would like to develop an algorithm that helps overcome these restrictions. We will also consider a more robust approach to the scaling of the target image, which is an issue in template matching.

References

- 1 H. Yoshida, T. Hyakudome, S. Ishibashi, T. Sawa, and T. Shimura: IEICE Commun. Soc. Mag. **5** (2011) 127. <https://doi.org/10.1587/bplus.5.127>
- 2 R. IWASE: Chigaku Zasshi **109** (2000) 885. https://doi.org/10.5026/jgeography.109.6_885
- 3 M. Caccia: Control Eng. Practice **15** (2007) 703. [https://doi.org/10.1016/S1474-6670\(17\)31764-0](https://doi.org/10.1016/S1474-6670(17)31764-0)
- 4 B. D. Lucas and T. Kanade: Proc. Imaging Understanding Workshop (1981) 121–130.
- 5 S. Tanathong and I. Lee: The Improvement of KLT for Real-time Feature Tracking from UAV Image Sequence: https://www.researchgate.net/publication/267692948_The_improvement_of_KLT_for_real-time_feature_tracking_from_UAV_image_sequence (accessed February 2021).

- 6 S. Ghosh, R. Ray, S. R. Krishna, S. N. Shome, and S. Nandy: *Mach. Vision Appl.* **27** (2016) 221. <https://doi.org/10.1007/s00138-015-0736-4>
- 7 B. R. Meijer: *Proc. 11th IAPR Int. Conf. Pattern Recognition* (1992) 760–763.
- 8 L. Zheng and X. He: *Classification Techniques in Pattern Recognition*: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.853.5313&rep=rep1&type=pdf> (accessed January 2021).
- 9 Thiang, A. T. Guntoro, and R. Lim: *Proc. 2001 Seminar of Intelligent Technology and Its Applications (SITIA, 2001)*.
- 10 D. Lee, G. Kim, D. Kim, H. Myung, and H. T. Choi: *Ocean Eng.* **48** (2012) 59. <http://doi.org/10.1016/j.oceaneng.2012.04.006>
- 11 A. Kaehler and G. Bradski: *Learning OpenCV, Computer vision with the OpenCV library* (O'Reilly Media Japan, Tokyo, 2009) 1st ed., Chap. 7.
- 12 H Lu, Y. Li, and Y. Zhang: *Mobile Networks Appl.* **22** (2017) 1204. <http://doi.org/10.1007/s11036-017-0863-4>