Image-based navigation of Small-size Autonomous Underwater Vehicle “Kyubic” in International Underwater Robot Competition

Yusuke Mizoguchi, Daiki Hamada, Riku Fukuda, Irmiya R. Inniyaka, Kaito Kuwata, Keisuke Nishimuta, Akihiro Sugino, Rikuto Tanaka, Yoshiki Tanaka, Yuya Nishida and Kazuo Ishii
Kyushu Institute of Technology, 2-4 Hibikino, Wakamatsu, Kitakyushu, Fukuoka 808-0196, Japan
E-mail: yusuke.mizoguchi323@mail.kyutech.jp , ishii@brain.kyutech.ac.jp
www.brain.kyutech.ac.jp/~underwater-robotics/

Abstract

An International underwater robot competition, “RoboSub” is held in USA to demonstrate robot’s autonomy by completing underwater tasks, with a new theme each year. Student project team, “Kyutech Underwater robotics” developed autonomous underwater vehicle (AUV) “KYUBIC” with built-in image processing module for the RoboSub. Sensors on the AUV are connected via ethernet and its automatic navigation program is built using ROS. The AUV moves through gate to the target objects based on self-localization using doppler velocity log and IMU. When approaching the object, the AUV aligns its direction to the object based on the red line detection by image processing as well as the type, position, and altitude of the target objects are detected by deep learning on the image processing module.

Keywords: Autonomous Underwater Vehicle, Robot competition, Image processing, Deep neural network.

1. Introduction

Kyutech Underwater Robotics Team competed in an international robotics competition called RoboSub2022[1]. RoboSub tasks students to provide solutions to simplified versions of challenges facing the maritime industry today. For example, these include marine survey and mapping, pipeline recognition and tracking, and object detection and manipulation[1]. Our team has previously competed in the Okinawa Ocean Robotics Competition[2] and the Techno-Ocean Kobe Competition[3]. Therefore, we decided to compete in RoboSub2022 because it is an international robotics competition and the difficulty level of the image processing tasks is quite different from that of domestic competitions, making it an attractive opportunity to learn new skills. Our team was also the only team from Japan to participate. RoboSub2022 is organized by RoboNation, an organization that organizes 9 other international programs[4]. In this paper, we report on the system configuration of our AUV "KYUBIC" and our strategy and results at RoboSub2022, focusing on image-based navigation.

2. The Hovering Type AUV “KYUBIC” and RoboSub2022

2.1. *Specifications of KYUBIC*

"KYUBIC" as shown in Figure 1 is a hovering AUV

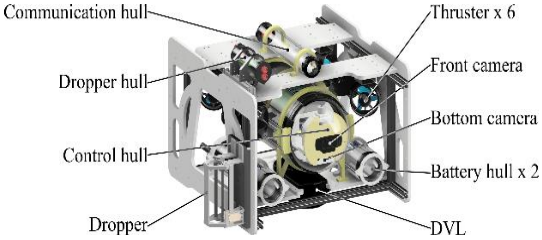

Fig.1. AUV” KYUBIC”

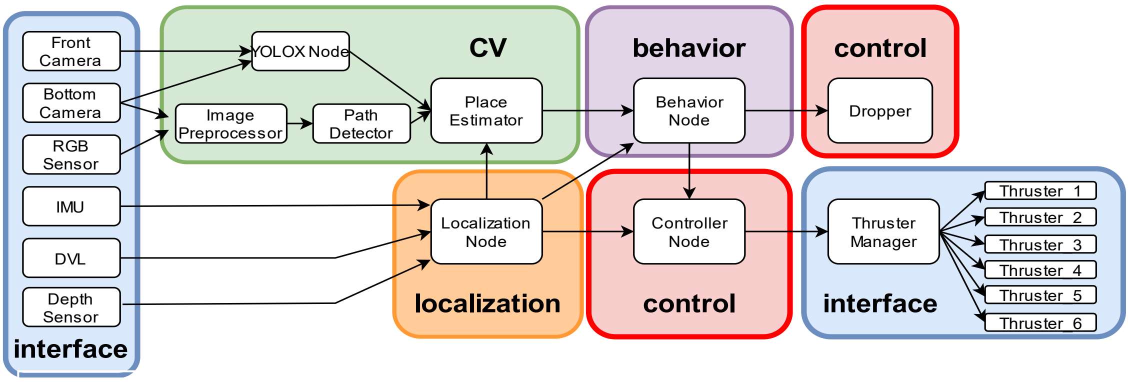

Fig.2. designed KYUBIC's node diagram

Table 1. Specification of AUV KYUBIC

Size (L, W, H)	570, 550, 400 [mm]
Weight (in air)	30 [kg]
Max. depth	15 [m]
Thrusters	Blue Robotics T200 x6
Computer1	ASRock 4X4 BOX-4800U
Computer2	NVIDIA Jetson Nano(4GB)
DVL	Teledyne Marine Pathfinder
IMU	CSM-MG200
Camera	Logitech c920n x2

platform designed for educational use. KYUBIC's hardware is modular in structure, allowing easy replacement of units (payloads) to suit experiments and robotic competitions. The hardware framework consists of some engineering plastic outer panel and an aluminum frame, providing both durability and portability. It is also equipped with various sensors such as DVL and IMU, six thrusters, and two computers, as shown in Table 1.

2.2. *KYUBIC's software configuration*

As described in the previous section, "KYUBIC" is equipped with various sensors and thrusters. While the AUV is in operation, the CPU continuously acquires information from various sensors, analyzes the AUV status, calculates the target thrust value for each thruster, and sends the target thrust value to the thrusters. Since this system is complex, our team has tried to improve the ease of development and debugging by dividing the system processing into packages for each function. As a result, the system is divided into five packages for interface, localization, CV (Computer Vision), control, and behavior. The relationship between the packages and nodes in them are shown in Figure 2. The behavior package contains the Behavior Node, which analyzes the situation and sends instructions to the Control Node on the next action to be taken, and the interface package contains the sensor drivers such as DVL, depth sensor, and IMU, and the communication interface to the microcontroller that controls the robot's actuators including the thrusters. The Localization package contains the Localization Node that calculates the self-position using dead reckoning from the DVL, IMU, and depth sensor values. The Control package includes a Control Node that calculates the target thrust by P-PID control according to the direction and distance the AUV should move based on the difference between its self-position and target position. This enables waypoint tracking to be performed. The CV package also includes PathDetector, YOLOXNode, PlaceEstimator, and ImagePreprocessor. PathDetector is a program that recognizes paths (orange boards) placed at the bottom of the pool and implemented in a rule-based algorithm. The YOLOXNode is a node that performs image recognition using YOLOX[5], one of the models of DNN (Deep Neural Network). The PlaceEstimator is a node that calculates the position of the target object in the world coordinate system by combining the information of object recognition nodes from the YOLOXNode and PathDetector with the current position of the AUV. Since the size of all objects appearing in this competition are known, the distance to the object can be calculated by the camera equation and the area of the detected object, and that the tasks required in this competition could be accomplished only by a position-based control program. The PathDetector, which is considered to have insufficient object detection performance for bluish images underwater, as such an ImagePreprocessor node is necessary to apply correction processing to the image

Fig.3. Mock objects photographed underwater

using an RGBSensor that can acquire the RGB components of ambient light.

3. Image recognition methods

Previously, rule-based image processing had mainly been used in our student projects, and we had been able to use MATLAB to find single-colored objects with straight lines and spheres[3]. However, to successfully complete the buoy touch and marker drop missions in this competition, our team needed to recognize complex images, which would be difficult to achieve using rule-based algorithms. Therefore, we decided to implement image recognition using DNN as a strategy to accomplish the missions.

Although more and more newer methods of DNN based image recognition are being proposed, our team selected the YOLOX model. YOLOX is chosen because it is licensed under the Apache-2.0 license, there are publicly available examples of its implementation using the "TensorRT SDK" (described below), and other users have also released implementation examples of ROS2 nodes. Therefore, our team decided to use the YOLOX model as it was suitable for integration into the AUV.

3.1. *Custom dataset generation and training*

When using DNNs to recognize their own objects, it is necessary to annotate and create their own dataset. To prepare the data set, our team printed the patterns published by the competition organizer on waterproof fabric and created mock objects by combining PVC pipes and metallic weights. Since the competition is held in an indoor pool underwater, we thought it would be appropriate to train the weights of the model based on images taken of the objects placed in the indoor pool with the same size as in the actual competition to make the conditions as close as possible. However, since we do not have an indoor pool easily accessible to us, we used an outdoor pool. The depth of the outdoor pool was shorter than the vertical length of the buoys for the actual competition. As a result, the mock objects created were reduced to approximately 70% the size of the actual dimensions for the competition. These were placed in an outdoor pool on campus and underwater video was captured using the AUV. The video was divided into frame-by-frame images, and the images were annotated to create a unique data set. The image of the mock objects taken underwater are shown in Figure 3. The annotation targets were 9 labels including "gate", "gun", "badge", "bootlegger", "g-man", "bottle", "barrel", "phone", and "paper" which included patterns that were not used in the buoy touch and marker drop missions. For training, the YOLOX-s weights pre-trained on the COCOtrain2017 dataset[6] were used as initial weights and fine-tuned with a batch size of 16 and a learning rate of 1/64,300 epochs. The learned weights were evaluated using AP (Average precision), AR (Average Recall), and IoU (Intersection Over Union). The AP was 80.7% and the AR was 83.6%. The number of training and validation images for each label, AP and AR are shown in Table 2. IoU was calculated between 0.50~0.95 for both AP and AR.

Table.2 The number of images of teacher data per label and AP and AR

Label	Train data	Validation data	AP [%]	AR [%]
gate	101	35	89.2	90.5
gun	165	49	83.2	86.3
badge	187	44	84.5	86.3
bootlegger	69	21	85.1	86.1
g-man	50	15	79.6	82.6
bottle	55	9	79.0	82.2
barrel	68	14	70.6	77.1
phone	58	14	75.2	80.7
paper	57	14	80.0	81.4
average	90	23.8	80.7	83.6

3.2. *Inference*

In determining the size of the network model, a policy was first determined. The policy was to select a model that could be inferred in about 100 [ms] and that was as accurate as possible. The inference time was determined since the information acquisition cycle of the AUV's onboard DVL is 3.5 [Hz] and that the AUV's motion has a large time constant. As a result, a network model of size YOLOX-s was selected within the YOLOX model. As mentioned above, the computational resource for inference built into "KYUBIC" is the "NVIDIA Jetson Nano (4GB)," which is inexpensive, small, and easily accessible, but not powerful. When inference was performed using Pytorch weight files and python, the median inference time on the Jetson Nano was 10.3 [sec] out of 10 experiments. This was far from satisfying our requirements. However, after optimization with the TensorRT SDK provided by Nvidia, the median inference time on the ROS node implemented in C++ was 0.110 [sec] out of 10 experiments, which was a significant reduction. This enabled us to infer YOLOX-s, calculate the center position of the buoy, and judge whether the relationship between the buoy and the AUV is stable, all in the target period of approximately 0.1 [sec].

4. Results

4.1. *Results of image recognition*

To evaluate the weights trained on the home-made dataset, a test dataset was created by annotating videos taken in the competition environment in the same way as when the home-made dataset was created. We executed inference on that dataset and calculated the Average Precision (AP) and Average Recall (AR). The results along with AP and AR on the dataset at training are shown in Table 3. This time, only the "badge" and "gun" labels were calculated due to the videos that could be taken in the competition environment. The number of test data images for each label was 170 and 106. Both AR and AP decreased in the test data set, suggesting that overlearning occurred in the training data set. Quantitative evaluation on the test data confirmed that recognition accuracy decreased. However, inference in the competition environment confirmed that when an object is in the image, it is always recognized as one object in succession without causing False Positive or False Negative, which recognizes it as a separate object.

Table.3 Comparison of AP and AR on training dataset and competition dataset

Label	AP(@train) [%]	AR(@train) [%]	AP(@test) [%]	AR(@test) [%]
gun	83.2	86.3	73.5	78.5
badge	84.5	86.3	74.4	78.9
Average	83.8	86.3	74.4	78.7

4.2. *Result of Competition*

Our team was not able to fully execute our strategy in this competition.

There were three main reasons for this. First, the Place Estimator could not be implemented. Second, the Image Preprocessor could not be implemented, so the Path Detector could not fully demonstrate its performance. Finally, the integration test between the "YOLOX Node" and the "Control Node" was not sufficiently completed. However, during the competition, the strategy was reconfigured and reimplemented, focusing on the position controller, which yielded high performance for the applied missions. As a result, our team successfully completed the missions our team concentrated on (gate pass and surfacing missions) and were ranked 8th out of 19 teams in the qualifying round.

Conclusion

In this paper, we have introduced the international robotics competition RoboSub2022, as organized by RoboNation and AUV "KYUBIC". We have also described the strategy of the competition, our implementation, and the image recognition module using DNN that was necessary to accomplish the strategy. Regarding the image recognition module, it was able to recognize pictures, but was not able to score points. This was because the integration test between the image recognition module and the position controller had not been sufficiently performed. However, we confirmed that the image recognition module’s performance was adequate. In the future, our team would like to expand the integration test, develop a target tracking program based on the image recognition results, and develop a program to calculate three-dimensional positions using an RGBD camera instead of an RGB camera.

References

1. “ RoboSub 2022 official website ” ,2022.(Accessed Dec.19,2022)
2. Irmiya R.I,et al.,"Underwater Acoustic Positioning Based on MEMS Microphone for a Lightweight Autonomous Underwater Vehicle "Kyubic"", Proceedings of

International Conference on Artificial Life & Robotics (ICAROB2022),pp. 349-353, 2022.
3. K.Harada, et al.," Autonomous Underwater Vehicle with Vision-based Navigation System for Underwater Robot Competition", Proceedings of International Conference on Artificial Life & Robotics (ICAROB2022),pp. 354-359, 2022.
4. "RoboNation official website",2022.(Accessed Dec.19,2022)
5. Ge, et al., "YOLOX: Exceeding YOLO Series in 2021", arXiv preprint arXiv:2107.08430,2021.
6. Lin, TY. et al.," Microsoft COCO: Common Objects in Context", European Conference on Computer Vision, pp.740-755,2014.

Authors Introduction

Mr. Yusuke Mizoguchi

He received his bachelor's degree from the Department of Electronics, Kyoto Institute of Technology, Japan in 2021. He is currently a Master's course student at the Department of Human Intelligence Systems, Graduate School of Life Science and Systems Engineering, Kyushu Institute of Technology, Japan.

Mr. Daiki Hamada

He received his master's degree from Department of Human Intelligence Systems, Kyushu Institute of Technology, Japan, in 2022. He is pursuing the PhD at Kyushu Institute of Technology, Department of Life and Systems Engineering, under the supervision of Prof. Kazuo Ishii.

Mr. Riku Fukuda

He received his bachelor's degree from the Department of Engineering, Electronic Information Engineering, Nishinippon Institute of Technology, Japan in 2021. He is currently a Master's course student at the Department of Human Intelligence Systems, Graduate School of Life Science and Systems Engineering, Kyushu Institute of Technology, Japan.

Mr. Irmiya R. Inniyaka

He received his master's degree in Automotive Mechatronics from the Department of Mechatronics, Cranfield University, England in 2016. He is currently pursuing his Doctoral program at Kyushu Institute of Technology, Japan. His research interests are in Control, Underwater Robotics navigation and localization.

Mr. Kaito Kuwata

He completed the production and electrical system technology course at Kyushu Polytechnic College in 2022. Currently enrolled in a master's course in Human Intelligence Systems, Graduate School of Life Science and Technology, Kyushu Institute of Technology, Japan.

Mr. Keisuke Nishimuta

He received his bachelor's degree from the Department of Mechanical and Control Engineering, the Kyushu Institute of Technology School of Engineering, Japan in 2022. He is currently a Master's course student in Kyushu Institute of Technology, Japan.

Mr. Akihiro Sugino

He received his bachelor's degree from the Faculty of Science and Engineering, Kyushu Sangyo University, Japan in 2021. He is currently a Master's course student at the Department of Human Intelligence Systems, Graduate School of Life Science and Systems Engineering, Kyushu Institute of Technology, Japan.

Mr. Rikuto Tanaka

He completed the production and electrical system technology course at Shikoku Polytechnic College in 2021. Currently enrolled in a master's course in Human Intelligence Systems, Graduate School of Life Science and Technology, Kyushu Institute of Technology, Japan.

Mr. Yoshiki Tanaka

He received his master's degree from Department of Human Intelligence Systems, Kyushu Institute of Technology, Japan, in 2019. He is pursuing the PhD at Kyushu Institute of Technology, Department of Life and Systems Engineering, under the supervision of Prof. Kazuo Ishii. His research area is underwater robots, its application.

Dr. Yuya Nishida

He is currently an Associate Professor at the Department of Human Intelligence Systems, Kyushu Institute of Technology, Japan. He obtained his master's degree in engineering in 2008 and his D. Eng. degree in 2011 at the same university. His research interests are in the field of Underwater Robotics, Field Robotics, and Intelligent Systems.

Dr. Kazuo Ishii

He is currently a Professor at the Department of Human Intelligence Systems of Kyushu Institute of Technology, Japan. He obtained his M.S.degree in 1993 and his D. Eng. degree in 1996 at The University of Tokyo. His research interests are in the fields of Underwater Robotics, Field Robotics, Neural Networks and Intelligent Systems.