Received January 31, 2018, accepted March 18, 2018, date of publication March 28, 2018, date of current version April 23, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2820326

Underwater-Drone With Panoramic Camera for Automatic Fish Recognition Based on Deep Learning

LIN MENG[1], (Member, IEEE), TAKUMA HIRAYAMA [2], AND SHIGERU OYANAGI[3], (Member, IEEE)
[1]College of Science and Engineering, Ritsumeikan University, Kyoto 525-8577, Japan
[2]OhMy DigiFab Company Ltd., Otsu 520-0806, Japan
[3]College of Computer Science and Engineering, Ritsumeikan University, Kyoto 525-8577, Japan

Corresponding author: Lin Meng (menglin@fc.ritsumei.ac.jp)

This work was supported by a grant-in-aid for scientific research from JSPS under Grant 15K00163.

ABSTRACT Highly developed drone technology enables the use of drones in a wide variety of areas. However, those drones are mainly used in the unmanned aerial vehicles. We believe that underwater drones will become a big research topic and find a market in the near future. We developed an underwater drone with a 360° panoramic camera acting as the ''eye'' of the drone. The designs are based on the open-source hardware and will be shared as an open-source for contributing to the innovation of manufacturing including drone. The function of the 360° panoramic camera is generated by correcting the images taken by two fisheye lenses. The underwater drone was designed by extending the Raspberry Pi compute module, the frame was designed by OpenSCAD, and the printed circuit board was designed by MakePro. As for the application of the underwater drone, we focused on fish recognition for investigating fish species in a natural lake to help protect the original environment. Fish recognition is based on deep learning, which is the biggest topic in the artificial intelligence research field today. Experimental results show that the function of the underwater drone achieved at diving in the leak automatically. The 360° panoramic images were generated correctly. Fish recognition achieved 87% accuracy by deep learning.

INDEX TERMS 360-degree panoramic image, underwater-drone, fish recognition, deep learning, open source hardware.

I. INTRODUCTION

Highly developed drone technology enables the use of drones in a wide variety of areas such as in aerial photography for appreciating the beauty of nature, in natural disasters where direct human intervention is impossible, or in agriculture for spraying pesticides to exterminate noxious insects. Furthermore, Amazon is preparing to use drones for delivering packages to customers. In a word, drone technology brings innovation and opens new markets. However, these drones are limited to the unmanned aerial vehicles. We believe that underwater drones, which are autonomous robots capable of moving and operating in the water, will become a big research topic and find a market in the near future [2]–[4].

A camera acting as an ''eye'' is an essential component of a drone. 360-degree panoramic cameras are widely used in various fields, and 360-degree panoramic images have attracted more attention with the increased support of panoramic movies by YouTube and Facebook. We believe that an underwater drone with a function of 360-degree image taking is a state-of-the-art research field which will bring innovations in many areas. For example, it may help to investigate and observes fish species in a lake, check the aging process of the walls of a dam, and so on [9]–[11].

For investigating and observing fish species in a lake, automatic fish recognition is the basis of the research. Recent advances in object recognition have used deep learning for obtaining high-level features in recognizing and classifying the objects.

In manufacturing, open-source software has been widely used in various areas for a long time. Open-source hardware was also defined recently and is making significant contributions into manufacturing innovations [5]–[8].

We developed an underwater drone that combines the hottest keywords in today's drone technology: ''360-degree panoramic camera'',''underwater drone'', ''fish recognition'', ''deep learning'', and ''open-source hardware''. Our model is designed based on open-source hardware, is equipped with a function of 360-degree panoramic camera, and has the ability of fish recognition based on deep learning. The 360-degree panoramic image-taking capability and the underwater drone were developed with an open-source software. The compute modules were extended on a Raspberry Pi compute module. The body was designed using a free software application for creating solid 3D computer-aided design objects (OpenSCAD). The printed-circuit board was designed with MakePro. For taking 360-degree panoramic images, the underwater drone was equipped with two 235-degree fisheye lenses; OpenGL ES2 was used for distorting the 180-degree fisheye images [12], [13].

The goal of this research was to use the underwater drone for investigating and observing the lakes, seas, and so on. We are currently testing it for investigating fish species in a natural lake to help protect the original environment.

Contributions of this paper:

- Designing an underwater drone equipped with the function of 360-degree panoramic image generation.
- Using open-source hardware for the design, which will help to drive innovation.
- Challenging fish recognition based on deep learning.

Section 2 describes the background of the 360-degree panoramic image generation and open-source hardware. Section 3 describes the design of the underwater drone with the function of generating 360-degree panoramic images. Section 4 shows the design and the experimental results of the underwater drone. Section 5 summarizes the results and lays out the goals for future work.

II. DESIGN OF THE UNDERWATER DRONE

Figure 1 shows a block diagram of the underwater drone and the peripheral equipment. Picam360-CAM shows the diagram of the main body of the drone. Picam360-FPV is part of the peripheral equipment showing 360-degree images taken by the underwater drone on a 5-inch display. PC/Smartphone shows the 360-degree images on a web browser like the ones used on a PC or a smartphone.

That is to say that the underwater drone can take an image of a 360-degree scene in lake, sea, or an aquarium, and transfer / display the 360-degree scene onto a 5-inch display and a web browser on a PC or a smartphone via network. The reason of using web browser is that Inetnet of things (IoT) are widely used in all kinds of fields [22]–[30]. We want to connect the underwater drone with internet for the research, appreciation too. For example, the user can appreciating the underwater scene of aquarium or sea in realtime, in the case of the people can not in there for some reasons.

A. BASIC COMPUTE MODULE AND EXTENSIONS

We used Raspberry Pi Compute Module as the basic compute module of the underwater-drone.

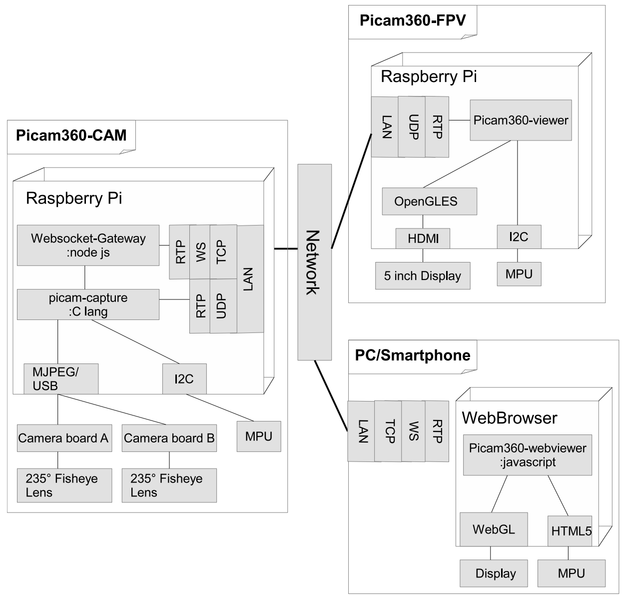

FIGURE 1. Block Diagram of underwater drone.

Raspberry Pi Compute Module is widely used in all kinds of fields, especially in the field of embedded systems like underwater drones. Some versions are equipped with a 700 MHz ARM central processing unit (CPU), 24 FFLOPs graphics processing units (GPUs), and two mobile industry processor interface (MIPI) high-speed cameras. 46 general purpose input/output pins were applied for easy extension. Several kinds of interfaces such as universal asynchronous receiver transmitter (UART), inter-IC (I2C), or serial peripheral interface (SPI) can be applied.

The circuit diagram of Raspberry Pi Compute Module is open and can be easily extended, even if it does not have external interfaces like power circuits and sockets.

We extended the basic compute module of Raspberry Pi by adding a small outline dual in-line memory module (SODIMM) socket, power circuit, two MIPI camera interfaces, two UARTs, USB 2.0, and Ethernet 100Base-T on a printed circuit board. The printed circuit board was designed using EagleCAD.

B. FRAME DESIGN

Two methods for the frame design are currently in use: one is a three-dimensional computer-aided design (CAD) method and another is a two-dimensional CAD method. We used the two-dimensional CAD method to design the frame of our underwater drone. This method relies on basic geometrical shapes like lines, rectangles, circles, etc. to produce flat drawings. These types of software have been first developed back in the 1970s. We used free OpenSCAD [14] software and divided the frame into several hierarchies. The hierarchies were designed by using the two-dimensional CAD as well. Acrylic was processed with a laser cutter and laminated with acrylic adhesive.

C. 360-DEGREE PANORAMIC IMAGE GENERATION

Traditional methods of the panoramic image generation are by distortion of fisheye correction and by image joints correction. Both methods are based on a large computation on the pixel level, hence, some versions have software for generating panoramic images (like RICOH Theta and KODAK SP360) and others are equipped for general-purpose computing on graphics processing units (GPGPU) for generating panoramic images (like Gyroptic 360cam).

Because Raspberry Pi is equipped with a GPU which has OpenGL ES engine, our underwater drone uses the GPU to achieve real-time panoramic image generation by using two fisheye lenses.

In general, the 360-degree panoramic image generation method derives the mapping function from point (i, j) of the target plane to point (x, y) on the fisheye image [12].

First, the target image should be normalized, the normalized $P\_n(u, v)$ is changed into equation (1). It means that a fisheye image in Figure 2(a) should be changed into an image in Figure 2(b).

$$\begin{cases} u = \dfrac{2i}{width - 1} \\ v = \dfrac{2j}{height - 1} \end{cases} \tag{1}$$

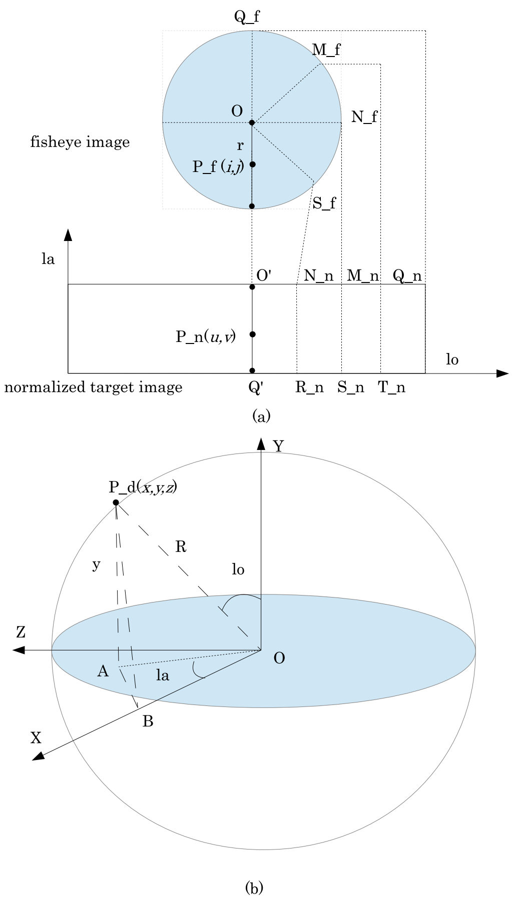

FIGURE 2. 360-degree panoramic image generation.

The lo and la are expressed by equation (2) and are shown in Figure 2 (c).

$$\begin{cases} lo = \dfrac{(1-u)\times \Pi}{2} \\ la = \dfrac{(1-v)\times \Pi}{2} \end{cases} \tag{2}$$

$$y = \overline{OC} = \frac{\sin(lo)}{R} \tag{3}$$

$$\begin{aligned} x &= \overline{OB} = \cos(la) \times \overline{OA} = \cos(la) \times \overline{OP}_d \\ &= \cos(la) \times \frac{\sin(lo)}{R} \end{aligned} \tag{4}$$

Because the ratio of R is 1, we can use the same to calculate the axis Z, the axis of 3-D images can be described in equation (3).

$$\begin{cases} x = \sin(lo) \times cos(la) \\ y = cos(lo) \\ z = \sin(la) \times sin(lo) \end{cases} \tag{5}$$

Finally, the x, y, z are expressed in equation (4) by changing equations (2) and (3).

$$\begin{cases} x = \sin(\dfrac{(1-u)\times \Pi}{2}) \times cos(\dfrac{(1-v)\times \Pi}{2}) \\ y = cos(\dfrac{(1-u)\times \Pi}{2}) \\ z = \sin(\dfrac{(1-v)\times \Pi}{2}) \times cos(\dfrac{(1-u)\times \Pi}{2}) \end{cases} \tag{6}$$

III. AUTOMATIC FISH RECOGNITION BASED ON DEEP LEARNING

A. DEEP LEARNING AND NETWORK ARCHITECTURE

Deep learning-based object recognition is better than the traditional machine learning and traditional feather matching in the case of large data. Deep learning is a kind of convolution neural network (CNN) which has deep layers consisting of convolution layers, polling layers, and fully-connected layers.

The most commonly used structure of a neural network (NN) is shown in Figure 3, which is formed in three

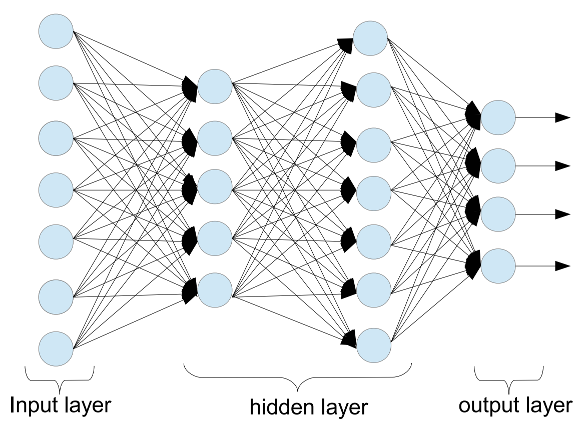

FIGURE 3. Neural network image.

layers called the input layer, hidden layer, and output layer. Each layer consists of one or more nodes represented in this diagram by the small circles. The narrow lines between the nodes indicate the flow of information from one node to the next. The output layers have four nodes, which means that there are four classifications in the case of object classification.

The nodes in the hidden layer and output layer are called active nodes, whereas the nodes in the input layer are called passive nodes.

Each value from the input layer is duplicated and sent to all the hidden nodes. This is called a fully interconnected structure.

As shown in Figure 4, the output of the active nodes is a sigmoid function which consists of input x_i, weight w_i, and the bias b.

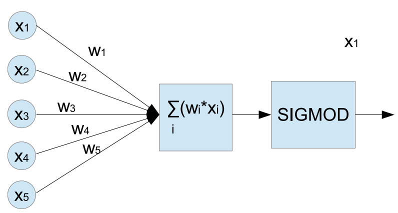

FIGURE 4. Active node.

For our recognition, we used the LeNet [16], AlexNet [17], and GoogLeNet [19]. LeNet was introduced in 1998 to classify images in grayscale. GoogLeNet is based on finding out how an optimal local sparse structure in a convolutional vision network can be approximated and covered by readily available dense components.

$$\sigma(z) = \frac{1}{1 + \exp\left(-\sum_i x_i w_i - b\right)} \tag{7}$$

B. DATA SET AND DATA AUGMENTATION

1) DATA SET

We picked four kinds of fish and 100 images of every kind from the Google search engine. The size and the image quality are not uneven.

The four kinds are the most popular aquarium fish in Japan: Japanese medaka, guppy, snakehead (or channa pleurophthalma), and neon tetra (or paracheirodon innesi). As data collection is an important part in deep learning, we used easily available popular aquarium fish for the early research. We will collect real images by the underwater drone in the future.

2) DATA AUGMENTATION

As 400 images were not enough for image recognition, and because the size and the image quality were not even, we augmented the dataset size to prevent over-fitting. We applied some augmentation parameters as follows:

Blur: Using the Gaussian filtering to blur images, we increased the Gaussian kernel from 0x2 to 0x8 with the 0x1 increase every time.

Rotation: We changed the angle from −180 degrees to 170 degrees with the increasing factor of 10 degrees every time.

After 36 rotations and five blurring, 72,000 blurred images and 14,400 non-blurred images were generated. All 86,400 images were made into 224 × 224 pixel for data training. The data augmentation tool was ImageMagick [1]. The data augmentation took about seven hours.

Figure 5 shows an example of data Augmentation, (a)is the original image, (b)-(g) are rotation images, (h) is the Bulrred image of original image, and (i)-(n) are the Blurred rotation images.

IV. EXPERIMENTAL RESULTS

A. DRONE AND 360-DEGREE PANORAMIC CAMERA

We designed the underwater drone and achieved real-time image streaming at 2048 × 1024, 5*fps*. Two 235-degree panoramic images taken by the fisheye lenses were corrected into 360-degree panoramic images.

Figure 6 shows the 360-degree panoramic images taken in an office. Figure 7 shows the drone diving in a lake: (a) shows the image of our drone, (b) shows the drone floating in the water, (c) shows the drone diving in the water, and (d) shows the drone achieving a certain level of depth. The drone of figure 7 (b),(c) and (d) is the old version which is taken in the Biwa Lake. The figure 7 (a) is the lasted released version. Hence the color of the frames are different.

B. FISH RECOGNITION RESULTS

We used four kinds of fish and three kinds of network to perform fish recognition. The test data was picked from the internet.

The machine used for fish recognition had four Geforce GTX-1080 Ti GPU boards; every GPU board had 3584 CUDA cores and 11GB GDDR5x memory.

Figure 8 shows the test images of four kinds of fish and Figure 9 shows the fish recognition results. More than 80% recognition was achieved for almost all kinds of fish, with AlexNet and GoogLeNet achieving 87 % and 85% recognition rate, respectively.

However, the recognition rates are different for different kinds of network. The recognition accuracy is only about 67% in LeNet due to the fact that LeNet recognizes the objects in grayscale and the image size is very small.

The recognition accuracy of guppy, medaka, and neon tetra using AlexNet is more than 5% higher than with GoogLeNet, especially for medaka (18%). However, the recognition accuracy of AlexNet is about 13% lower than GoogLeNet. It may be because AlexNet is better than the newest network of GoogLeNet.

Because LeNet is a simpler network than AlexNet and GoogLeNet and because the image size was very small, the training time was much shorter than with AlexNet and GoogLeNet.

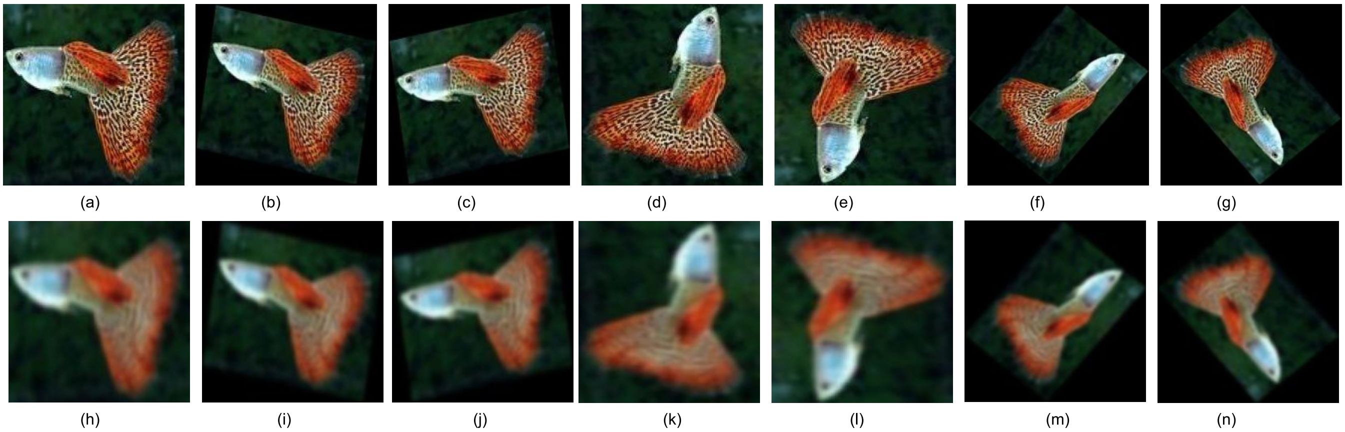

FIGURE 5. Data augmentation examples. (a) original image. (b) 10-degree rotation. (c) −10-degree rotation. (d) 90-degree rotation. (e) −90- degree rotation. (f) 130-degree rotation. (g) −130-degree rotation. (h) original image Gaussions 0 × 2. (i) 10-degree rotation Gaussian 0 × 2. (j) −10-degree rotation Gaussion 0 × 2. (k) 90-degree rotation Gaussion 0 × 2. (l) −90-degree rotation Gaussion 0 × 2. (m) 130-degree rotation Gaussion 0 × 2. (n) −130-degree rotation Gaussion 0 × 2.

FIGURE 6. 360-degree panoramic image of an office.

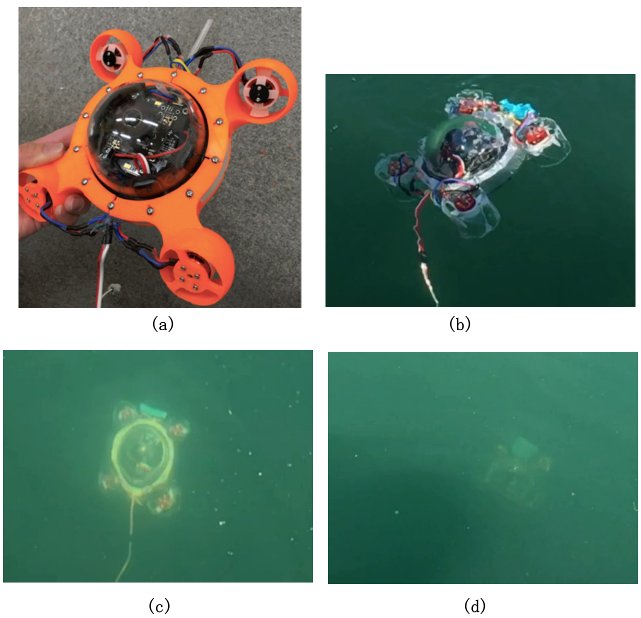

FIGURE 7. Images of diving underwater drone. (a) Drone image. (b) Preparation step. (c) Diving example one. (d) Diving example two.

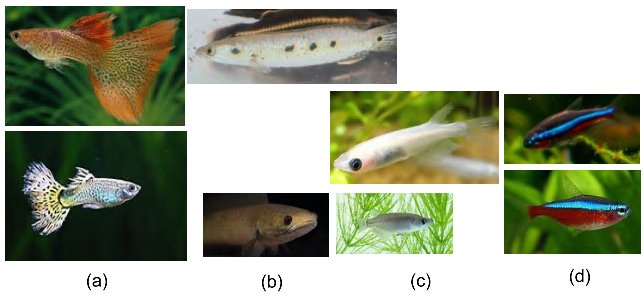

FIGURE 8. Fish examples. (a) Guppy. (b) Snakehead. (c) Medaka. (d) Neontetora.

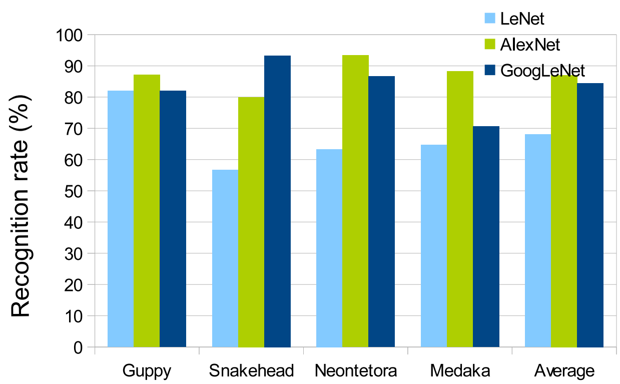

FIGURE 9. Fish recognition results.

TABLE 1. Time.

	Training Time	Recognition time
LeNet	2 minutes	3 seconds
AlexNet	32 minutes	6 seconds
GoogLeNet	2 hours	5 seconds

Table 1 shows the training time and recognition time of 3 kinds network. The recognition time of 115 test images was only about 5 seconds with AlexNet and GoogLeNet. AlexNet is therefore suitable for a real-time application.

C. DISCUSSION WITH OTHER DRONES

Table 2 lists the several underwater drones production information including name, company HP, cost, speed depth rating

TABLE 2. Comprision table of underwater drone.

Drone name	Company(HP)	Cost ($)	Speed (knot)	Depth rating (m)	Dimension (mm^3)	weight (kg)
openrov	https://www.openrov.com/	1200	2	100	150x200x300	3.0
Powerray	http://www.powervision.me/en/html/pv/powerray.html	2000	3	30	465x270x126	3.8
BlueRov	https://www.bluerobotics.com/store/rov/bluerov2/	3000	4	100	457x338x254	11
Rouv360 (our drone)	http://www.ohmydigifab.com/ja/rouv360/	1000	2	100	150x150x120	1kg

dimension and weigh for comprision. According the table, we found that our drone is the most small, light and low cost. However it can diving in the depth of 100 m with the 2 knot speed. Furthermore, only our drone can realize the 360-degree image.

V. CONCLUSION

This paper presented an underwater drone equipped with fisheye lenses and with the function of a 360-degree panoramic camera for taking panoramic images by using an image generation algorithm. The 360-degree panoramic image generation and the underwater drone were developed with open-source software; the compute modules were extended on a Raspberry Pi compute module. We implemented an automatic underwater drone and conducted experiments in a lake. The 360-degree panoramic images were generated correctly. The experimental results showed that almost all fish species were recognized with a recognition rate higher than 85% with AlexNet and GoogLeNet (AlexNet achieved 87%). The recognition time for 115 images was 6 seconds. Hence, AlexNet may be used in a real-time application with high accuracy. In the future, we aim to improve the underwater drone to a practical level.

REFERENCES

[1] *ImageMagick*. Accessed: Aug. 20, 2017. [Online]. Available: http://www.imagemagick.org/

[2] M. Dumiak, "Lice-hunting underwater drone protects salmon," *IEEE Spectr.*, vol. 54, no. 4, pp. 9–10, Apr. 2017.

[3] K. Grythe, T. A. Reinen, and A. A. Transeth, "Autonomy levels versus communication in an underwater environment," in *Proc. OCEANS Genova*, May 2015, pp. 1–8.

[4] S. Bhattacharyya and H. H. Asada, "Single jet impinging verticial motion analysis of an underwater robot in the vicinity of a submerged surface," in *Proc. OCEANS Shanghai*, Apr. 2016, pp. 1–8.

[5] *Opensource*. Accessed: Jun. 6, 2017. [Online]. Available: https://opensource.org/

[6] S. Heunis, Y. Paichard, and M. Inggs, "Passive radar using a software-defined radio platform and opensource software tools," in *Proc. IEEE RadarCon (RADAR)*, May 2011, pp. 879–884.

[7] A. Ribière, "Emulation of obsolete hardware in open source virtualization software," in *Proc. 8th IEEE Int. Conf. Ind. Inform.*, Jul. 2010, pp. 354–360.

[8] B. Mohammed and M. Kiran, "Analysis of cloud test beds using open-source solutions," in *Proc. 3rd Int. Conf. Future Internet Things Cloud*, Aug. 2015, pp. 195–203.

[9] *SP360*. Accessed: Mar. 29, 2018. [Online]. Available: http://www.maspro.co.jp/products/pixpro/sp360/

[10] *Theta360*. Accessed: Mar. 29, 2018. [Online]. Available: https://theta360.com/ja/

[11] Y. K. Ho, "Panorama interpolation for novel view composition," M.S. thesis, Dept. Comput. Sci., Univ. Hong Kong, Hong Kong, 2000.

[12] X. Shouzhang and W. Fengwen, "Generation of panoramic view from 360∘ fisheye images based on angular fisheye projection," in *Proc. 10th Int. Symp. Distrib. Comput. Appl. Bus., Eng., Sci.*, Oct. 2011, pp. 187–191.

[13] P. Dhane, K. Kutty, and S. Bangadkar, "A generic non-linear method for fisheye correction," *Int. J. Comput. Appl.*, vol. 51, no. 10, pp. 58–65, 2012.

[14] *Openscad*. Accessed: Jun. 6, 2017. [Online]. Available: http://www.openscad.org/

[15] S. W. Smith, *The Scientist & Engineer's Guide to Digital Signal Processing*. San Diego, CA, USA: California Technical Publishing, 1997.

[16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 25. 2012, pp. 1097–1105.

[18] G. Lee, Y. W. Tai, and J. Kim, "ELD-net: An efficient deep learning architecture for accurate saliency detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 8, pp. 1–13, 2015.

[19] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[20] J. Jäger, E. Rodner, J. Denzler, V. Wolff, and K. Fricke-Neuderth, "SeaCLEF 2016: Object proposal classification for fish detection in underwater videos," in *Proc. 7th Int. Conf. CLEF Initiative*, 2016, pp. 1–9.

[21] P. Zhuang, L. Xing, Y. Liu, S. Guo, and Y. Qiao, "Marine animal detection and recognition with advanced deep learning models," in *Proc. Conf. Labs Eval. Forum (CLEF)*, 2017, pp. 1–9.

[22] Z. Lv, A. Tek F. Da Silva, C. Empereur-Mot, M. Chavent, and M. Baaden, "Game on, science—How video game technology may help biologists tackle visualization challenges," *PLoS ONE*, vol. 8, no. 3, p. e57990, 2013.

[23] Z. Lv, A. Halawani, S. Feng, H. Li, and S. U. Rehman, "Multimodal hand and foot gesture interaction for handheld devices," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 11, no.1s, 2014, Art. no. 10.

[24] Z. Lv, A. Halawani, S. Feng, S. U. Rehman, and H. Li, "Touch-less interactive augmented reality game on vision-based wearable device," *Pers. Ubiquitous Comput.*, vol. 19, nos. 3–4, pp. 551–567, 2015.

[25] Z. Lv *et al.*, "Managing big city information based on WebVRGIS," *IEEE Access*, vol. 4, pp. 407–415, 2016.

[26] Y. Sun, H. Song, A. J. Jara, and R. Bie, "Internet of Things and big data analytics for smart and connected communities," *IEEE Access*, vol. 4, pp. 766–773, 2016. [Online]. Available: http://dx.doi.org/10.1109/ACCESS.2016.2529723

[27] Y. Sun, C. Lu, R. Bie, and J. Zhang, "Semantic relation computing theory and its application," *J. Netw. Comput. Appl.*, vol. 59, pp. 219–229, Jan. 2016. [Online]. Available: http://dx.doi.org/10.1016/j.jnca.2014.09.017

[28] Y. Sun, J. Zhang, and R. Bie, "Measuring semantic-based structural similarity in multi-relational networks," *Int. J. Data Warehousing Mining*, vol. 12, no. 1, pp. 20–33, 2016, doi: http://dx.doi.org/10.4018/IJDWM.2016010102

[29] Y. Sun and A. J. Jara, "An extensible and active semantic model of information organizing for the Internet of Things," *Pers. Ubiquitous Comput.*, vol. 18, no. 8, pp. 1821–1833, 2014.

[30] Y. Sun, H. Yan, C. Lu, R. Bie, and Z. Zhou, "Constructing the Web of events from raw data in the Web of things," *Mobile Inf. Syst.*, vol. 10, no. 1, pp. 105–125, 2014.

LIN MENG (M'17) received the B.S., M.S., and Ph.D. degrees in science and engineering from Ritsumeikan University in 2006, 2008, 2012, respectively.

He was a Research Associate with the Department of Electronic and Computer Engineering, Ritsumeikan University from 2011 to 2013, where he was also an Assistant Professor from 2013 to 2018. From 2015 to 2016, he was a Visiting Scholar with the Department of Computer Science and Engineering, University of Minnesota at Twin Cities. In 2018, he became a Lecturer with the Department of Electronic and Computer Engineering, Ritsumeikan University. His research interests include computer architecture, parallel processing, and image processing. He is a member of IEICE and IPSJ.

TAKUMA HIRAYAMA received the B.S. and M.S. degrees in science and engineering from Ritsumeikan University in 2004 and 2006, respectively.

He is currently the COE of OhMy DigiFab Company Ltd.

SHIGERU OYANAGI (M'17) received the Ph.D. degree in engineering from Kyoto University in 1979.

He is currently a Professor with the College of Information Science and Engineering, Ritsumeikan University. His research interests include computer architecture, parallel processing, and data mining. He is a member of IEICE, IPSJ, and ACM.