Article Information

Title	Battery-Powered Wild Animal Detection Nodes with Deep Learning
Authors	Hiroshi SAITO, Tatsuki OTAKE, Hayato KATO, Masayuki TOKUTAKE, Shogo SEMBA, Yoichi TOMIOKA, and Yukihide KOHIRA
Citation	IEICE TRANSACTIONS on Communications Vol.E103-B, No.12, pp.1394-1402
Copyright	copyright@2020 IEICE
IEICE Transactions Online URL	https://search.ieice.org/

PAPER *Special Section on IoT Sensor Networks and Mobile Intelligence*

Battery-Powered Wild Animal Detection Nodes with Deep Learning

Hiroshi SAITO†a), ***Member***, **Tatsuki OTAKE**†, **Hayato KATO**†, **Masayuki TOKUTAKE**†, **Shogo SEMBA**†, ***Nonmembers***, **Yoichi TOMIOKA**†, ***and*** **Yukihide KOHIRA**†, ***Members***

SUMMARY Since wild animals are causing more accidents and damages, it is important to safely detect them as early as possible. In this paper, we propose two battery-powered wild animal detection nodes based on deep learning that can automatically detect wild animals; the detection information is notified to the people concerned immediately. To use the proposed nodes outdoors where power is not available, we devise power saving techniques for the proposed nodes. For example, deep learning is used to save power by avoiding operations when wild animals are not detected. We evaluate the operation time and the power consumption of the proposed nodes. Then, we evaluate the energy consumption of the proposed nodes. Also, we evaluate the detection range of the proposed nodes, the accuracy of deep learning, and the success rate of communication through field tests to demonstrate that the proposed nodes can be used to detect wild animals outdoors.

key words: *wild animal detection, deep learning, camera-trap, microcomputer boards, and power saving*

1. Introduction

Problems caused by wild animals such as crop damages and accidents to people have happened in areas inhabited by people and wild animals. Especially, sudden encounters between people and wild animals result in significant damages to people. To reduce these problems, it is important to safely detect wild animals as early as possible.

Various measures have been taken to detect and report wild animals. The most general measure is the use of eyewitness information which is typically provided to local municipal authorities and polices. Then, the information is distributed to citizens using e-mail and the Internet. However, this measure tends to be ineffective owing to delays in the distribution and reception of the information.

Other traditional measures are traps and electric fences. In order to set traps, cages are usually located where wild animals are likely to pass. When seeking food inside a cage, the door is closed to trap wild animals. However, traps are ineffective if wild animals do not enter the case. Using electric fences, crop and livestock can be protected from wild animals. However, the electric fences are expensive and can be dangerous to people also.

Drones have been used to detect wild animals [1], [2]. Using drones with attached sensors and cameras, wild animals can be detected from the sky. However, the effectiveness of drones depends on weather conditions, geographic conditions, fuel efficiency, and pilot ability.

Recently, camera-traps which can automatically detect wild animals using sensors and cameras are widely used. Trail cameras [3] are one such example. This is because the automatic detection of wild animals is safe and the cost of devices is reasonable. However, commercially available trail cameras do not classify which wild animals are detected. Also, excessive manpower is required to analyze the captured images. To save manpower, [4], [5], [6], [7], [8], and [9] proposed automatic wild animal identification methods based on convolutional neural network (CNN). Using these methods, captured images are automatically classified into species with high accuracy and high speed. [7] and [8] can also perform automatic counting of wild animals in captured images. CNN is also used for other purposes. [10] proposed a method to recognize individual Japanese macaques on location trajectories using the personal traits. [11] proposed a method to track and identify individual Japanese macaques using CNN and particle filter.

Reference [12] proposed an Internet-of-Things (IoT) system called *Where's The Bear*. This system consists of three system components, motion-triggered camera-traps, resource-constrained edge systems located near the camera-traps, and cloud systems. The camera-traps capture images and send the images to the edge systems. The cloud systems construct a trained model based on CNN using a synthetic training set for wild animals. The edge systems implement the trained model and classify wild animals.

The above methods which use camera-traps and deep learning [4], [5], [6], [7], [8], [9], [10], [11], and [12], on the other hand, did not notify the people concerned when wild animals were detected. To reduce accidents and damages, a system which can automatically detect wild animals and notify the detection information to the people concerned as soon as possible is required. Although there are many critical requirements for such a system, energy consumption is one of the significant problems because usually such a system must be used outdoors where power is not available.

In this paper, we propose two battery-powered wild animal detection nodes based on deep learning. The first node is based on a single micro-computer board, while the second node is based on two micro-computer boards. In addition to automatic detection, deep learning based on CNN is useful to reduce the energy consumption of the nodes. If a tar-

Manuscript received January 10, 2020.
Manuscript revised April 13, 2020.
Manuscript publicized July 1, 2020.
†The authors are with the University of Aizu, Aizu-Wakamatsu-shi, 965-8580 Japan.
a) E-mail: hiroshis@u-aizu.ac.jp
DOI: 10.1587/transcom.2020SEP0004

get wild animal is not detected by CNN, operations such as communication with a server to notify the detection of the wild animals are skipped to save energy consumption.

The proposed node with two micro-computer boards contributes to the power reduction during idle time. Usually, deep learning consumes more energy because of heavy computation. Therefore, a low power micro-computer board is used for sensing and image capturing. Only when something is detected, power is supplied to another micro-computer board for deep learning and the remaining operations. This results in the drastic power reduction during idle time.

In the experiment, we evaluate the operation time and the power consumption for the proposed nodes to clarify the energy consumption of the proposed nodes. We also clarify when the node with two micro-computer boards is useful for energy consumption. Then, we evaluate the sensing range of the proposed nodes, the accuracy of deep learning, and the success rate of communication through field tests.

The rest of this paper is organized as follows. In Sect. 2, we first describe the functional requirements for the proposed nodes. Then, we describe the structure and the operational flow of the proposed nodes. In Sect. 3, we evaluate the proposed nodes in terms of energy consumption. Also, we evaluate the proposed nodes through field tests. Finally, in Sect. 4, we describe the conclusion and future work.

2. Proposed Wild Animal Detection Nodes

The proposed wild animal detection nodes are powered by a battery to be used outdoors where power is not available. In this section, we initially define the functional requirements for the proposed nodes. Then, we describe the structure and the operational flow of the proposed nodes. We also describe how power consumption is reduced by the proposed nodes.

2.1 Functional Requirements

To detect wild animals and notify the people concerned as early as possible, the proposed nodes have the following functions.

- Detection of the target wild animal by applying deep learning to static images captured by a camera
- Connection to the Internet to notify the detection to the people concerned
- Driving away the detected wild animal from the proposed nodes and the notification of the detection to people near the proposed nodes using sound and light

CNN is used to detect the target wild animal in images captured by a camera. The use of CNN allows us to use the proposed nodes for any type of wild animal just by changing the trained models of CNN.

A video recording could be more accurate. However, applying deep learning to multiple frames obtained in a video consumes more power at the proposed nodes. Therefore, the proposed nodes capture static images and infer wild animals from the static images.

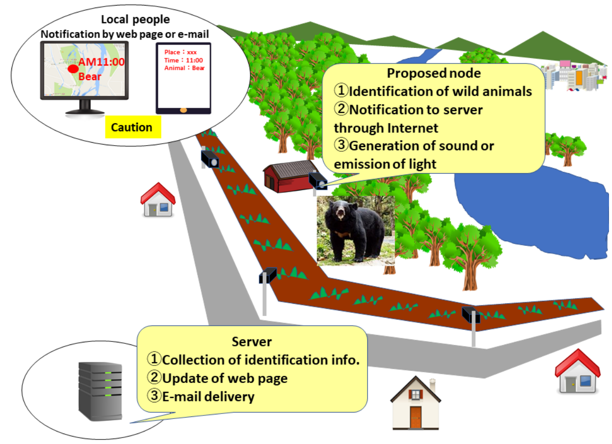


Fig. 1 Application example of the proposed nodes.

Early notification of the detection of a target wild animal is necessary to reduce damages and accidents. The proposed nodes connect to the Internet when the target wild animal is detected by deep learning. The detection information is sent to a server to notify the people concerned using a web page and e-mail.

Also, the proposed nodes generate loud sound and flashing light to notify people near the proposed nodes. The light and sound may drive away the detected wild animal from the proposed nodes.

Figure 1 represents an application example of the proposed nodes. When the proposed nodes are placed in or near a vegetable field, livestock hut, or tourist site, and a target wild animal is detected the proposed nodes generate sound and light to drive away the detected wild animal from the proposed nodes and to notify people near the proposed nodes. Also, the proposed nodes connect to the Internet to send the detection information to a server. The server updates a web page and distributes an e-mail to notify the detection information to the people concerned. Thus, e-mail recipients are made aware immediately.

2.2 Proposed Node Based on a Single Micro-Computer Board

Figure 2 represents the operational flow of the proposed node with a single micro-computer board. First, the operation starts when an auxiliary sensor such as a motion sensor detects something moving. Then, to capture a static image during night, the relay module which supplies power to an infrared projector is turned on when the value of a light sensor is smaller than a threshold value. After capturing a static image by a camera, the inference of the target wild animal for the image is performed using a trained CNN model. When the target wild animal is detected in the image, sound and light are generated to drive away the detected wild animal from the proposed node and to notify people near the proposed node. Also, the proposed node turns on a wireless module to send the detection information to a server through

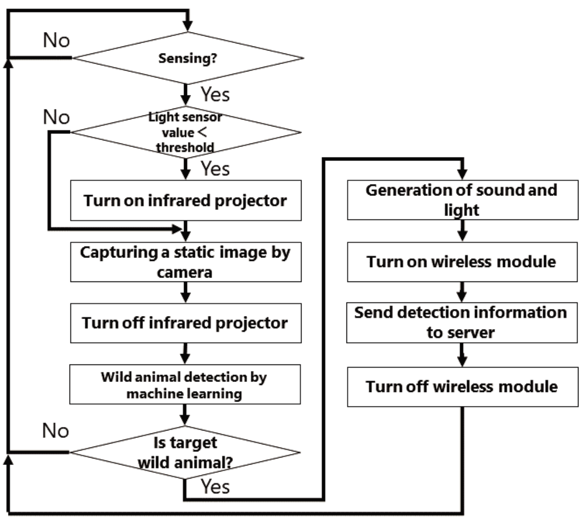


Fig. 2 Operational flow of the proposed sensor node with a single micro-computer board.

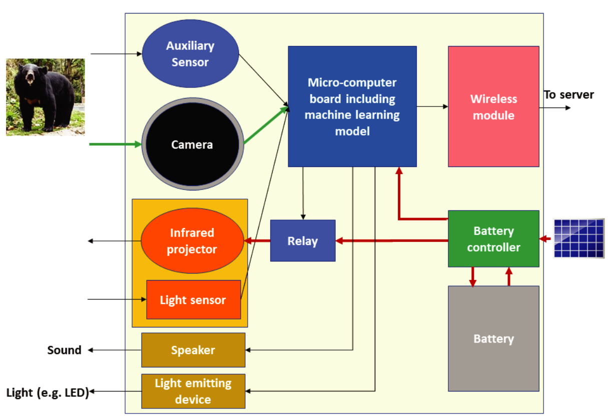


Fig. 3 Structure of the proposed node with a single micro-computer board.

the Internet. After the transmission, the proposed node waits for the next sensing by turning off the wireless module.

To save the power consumption of the proposed node, the operation flow performs the required operations only when the assigned conditions are satisfied. First, the proposed node does not perform anything without detecting something with an auxiliary sensor. Second, the proposed node does not supply power to the infrared projector when the value of the light sensor is over the threshold value. This is very useful because the power consumption of the infrared projector is very large. Furthermore, the proposed node does not generate light and sound and connect to the Internet if a target wild animal is not detected by deep learning.

Figure 3 represents the structure of the proposed node. A battery, infrared projector, and micro-computer board are connected through a battery controller. On the other hand, an auxiliary sensor, camera, light sensor, relay module, wireless module, speaker, and light emission device are connected to the micro-computer board directly to control them from the processor in the micro-computer board. The

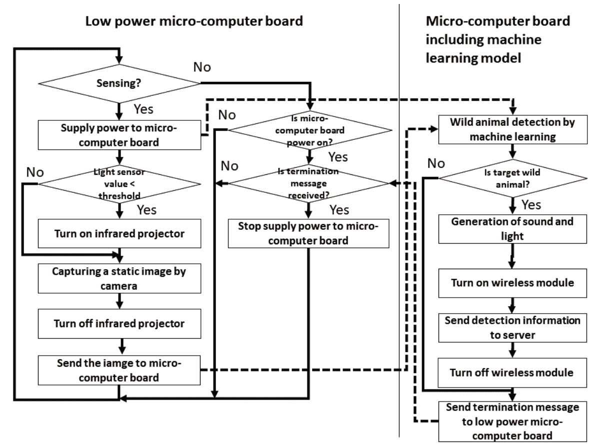


Fig. 4 Operational flow of the proposed node with two micro-computer boards.

relay module is used as a switch to supply power to the infrared projector. The wireless module uses a public line (e.g., 3G/LTE) to connect to the Internet from rural areas and the foot of mountains. The battery is a mobile battery or storage battery with a solar panel.

2.3 Proposed Node Based on Two Micro-Computer Boards

The battery life is the most important when the proposed node is used outdoors where power is not available. It may be possible to use a larger solar panel and a storage battery. However, this increases cost. Moreover, this becomes difficult to place the proposed node because of the increase in weight. To reduce power consumption, we propose a wild animal detection node with two micro-computer boards.

The proposed node with two micro-computer boards separates the required functions. A low power micro-computer board controls an auxiliary sensor, camera, light sensor, and relay modules. The relay modules are used to supply power for the infrared projector and the other micro-computer board. Basically, to control those components, we do not require high performance. On the other hand, a high performance micro-computer board performs deep learning and connection to the Internet. The separation of the functions saves power consumption during idle time. Although the low power micro-computer board must be turned on permanently, the high performance micro-computer board does not consume any power during the time if nothing is detected by the auxiliary sensor.

Figure 4 represents the operational flow of the proposed node with two micro-computer boards. First, the low power micro-computer board controls the relay module for the high performance micro-computer board to supply power when something is detected by an auxiliary sensor. Then, the low power micro-computer board controls the relay module for the infrared projector when the value of the light sensor is smaller than a threshold value. The low power micro-computer board also captures a static image using a

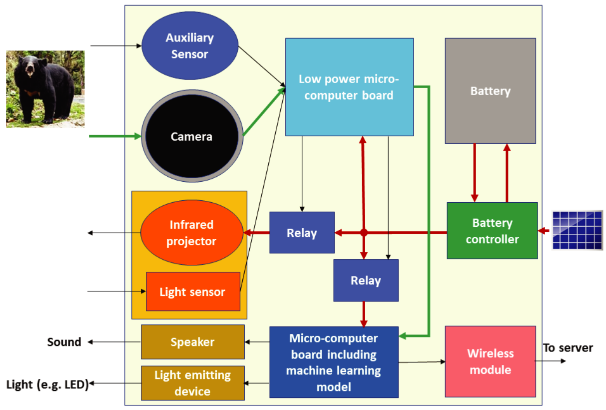


Fig. 5 Structure of the proposed node with two micro-computer board.

camera. After the high performance micro-computer board is turned on, the low power micro-computer board sends the image to the high performance micro-computer board and waits a fixed time or a completion message to complete its operations at the high performance micro-computer board. The high performance micro-computer board performs the inference of the target wild animal by deep learning when the static image is received. Then, the high performance micro-computer board generates light and sound to drive away the detected wild animal from the proposed node and to notify people near the proposed node. The high performance micro-computer board also connects to the Internet to send the detection information to a server. Finally, the high performance micro-computer board sends a completion message to the low power micro-computer board if required. After all operations for detection complete, if the low power micro-computer does not detect, from the auxiliary sensor outputs, anything within some period, it controls the relay module to stop the supply of power to the high performance micro-computer board.

Figure 5 represents the structure of the proposed node with two micro-computer boards. The main difference from the proposed node with a single micro-computer board (Fig. 3) is that an additional relay module is used to manage power for the high performance micro-computer board. Note for the image data transfer between the two micro-computer boards, we use serial communication through the general purpose I/Os (GPIOs) of the boards.

3. Evaluation

To demonstrate that the proposed nodes can be used outdoors, we evaluate the proposed nodes in terms of energy consumption through the evaluation of the operation time and the power consumption. Then, we evaluate the detection range of the proposed nodes, the accuracy of deep learning, and the success rate of communication through field tests. The target wild animal in these evaluations is the Japanese black bear (*Ursus thibetanus japonicus*).

Table 1 represents the specification of three developed prototypes (*oneboard_v0*, *oneboard_v1*, *twoboard*).

Fig. 6 Developed prototype (*twoboard*).

oneboard_v0 is a simplified version of the proposed node with a single micro-computer board powered by a mobile battery. *oneboard_v1* is a full version of the proposed node with a single micro-computer board powered by a lead storage battery. *twoboard* is the proposed node with two micro-computer boards. The used micro-computer boards are Raspberry Pi [13]. The low power micro-computer board in *twoboard* is Lazurite developed by LAPIS semiconductor [14]. The CNN model is a trained model of Inception-v3 provided by Google [15]. Note that we did not implement a speaker and a light emission device in the evaluation to avoid stress to the bears in a bear zoo where we visited to test the developed prototypes. Also, we did not use a solar panel in this evaluation. Figure 6 represents *twoboard*. The upper middle component is the infrared projector, the component under the projector is the camera, and the components under the camera are the motion sensor and the light sensor.

There are differences in the used Raspberry Pi. We used Raspberry Pi Zero for *oneboard_v0* to reduce power consumption. However, because the performance of the ARM processor in Raspberry Pi Zero was low and the size of the main memory was small (single-core CPU, 1GHz clock frequency, and 512MB memory), it took about five minutes for the load of the trained model and the inference of the target wild animal. We evaluated that it is difficult to detect wild animals in real time when we use Raspberry Pi Zero. Therefore, in *oneboard_v1* and *twoboard*, we used Raspberry Pi 3 (quad-core CPU, 1.2GHz clock frequency, and 1GB memory).

There are differences in the used components between *oneboard_v1* and *twoboard* too. The components represented by brackets are components used in *twoboard*. The differences come from the difference which Raspberry Pi or Lazurite is used for the control of the components. Raspberry Pi supports only digital components while Lazurite supports both digital and analog components.

3.1 Operation Time, Power Consumption, and Energy Consumption

Table 2 represents the operation time when we held a picture of the Japanese black bear in front of *oneboard_v1* and *twoboard*. The time is an average value of the ten time

Table 1 Specification of the developed prototypes.

Name	*oneboard_v0*	*oneboard_v1* *twoboard*
Micro-computer	Raspberry Pi Zero	Raspberry Pi 3 Model B
Low power micro-computer	-	Lapis Lazurite
deep learning model	Google inception-v3	Google inception-v3
Auxiliary sensor	Seeedstudio PIR Motion Sensor	Seedstudio PIR Motion Sensor
Camera	Kuman Raspberry Pi Camera SC15	Kuman Raspberry Pi Camera SC15 (ArduCam OV5642)
Infrared projector	Broadwatch SEC-IRLED-2B	Broadwatch SEC-IRLED-2B
Light sensor	TSL2561	TSL2561 (cds cell 5mm GL5537-2)
Wireless module	NTT DoCoMo L-02C	NTT DoCoMo L-05A
Battery controller	-	Indoor Corgi Elec. E32-SolarCharger
Battery	REV...POWER Mobile Battery 30,000 mAh	Long 12 V 7.2 Ah

Table 2 Operation time of *oneboard_v1* and *twoboard* [s].

Name	*oneboard_v1*	*twoboard*
Sensing to detection	26.5	116.1
(communication between boards)	-	70.3
(inference)	19.6	19.8
Detection to completion	48.6	46.5
(communication)	14.6	11.5
Sensing to completion	75.1	162.6

executions. “sensing to detection” is the time from sensing to the completion of the inference through communication between boards and capturing a static image by the camera. “communication between boards” in “sensing to detection” is the time by serial communication of the image between Raspberry Pi and Lazurite in *twoboard*. “inference” in “sensing to detection” is the inference time by Inception-v3. “detection to completion” is the time from the completion of the inference to the turn off of the wireless module through turning on of the wireless module and the transmission of the detection information to our server. “communication” in “detection to completion” is the actual communication time by the wireless module. “sensing to completion” is the sum of “sensing to detection” and “detection to completion”. In other words, this time corresponds to the required time for the next sensing. Note that we implemented the control programs using the Python language. For the measurement of the times, we inserted time() functions to the programs. The average image size was 159 KB in *oneboard_v1* and 28 KB in *twoboard*. This difference was the difference of the cameras used for Raspberry Pi and Lazurite. The used picture was a color image with dimensions 640×480.

oneboard_v1 detected the Japanese black bear about 26.5 s after sensing. It implies that *oneboard_v1* can drive away the detected bear from *oneboard_v1* and notify the detection to people near *oneboard_v1* after 26.5 s using light and sound. On the other hand, *oneboard_v1* required 75.1 s to complete all operations after sensing. Within 75.1 s, the inference time was 19.6 s and the communication time was 14.6 s. The longest time except for the inference time and the communication time was the time to turn on the wireless module (about 23 s). The reason for this result was the setting of a long wait time to ensure a stable connection to the Internet.

In applying the proposed nodes, the duration of “sensing to detection” must be shortened as much as possible. If we can reduce this time, the proposed nodes can immediately generate sound and light to drive away the detected wild animals and to notify the detection information to people near the proposed nodes. As a result, damages and accidents by the detected wild animals will be reduced. Since currently the inference time dominates the time “sensing to detection” (19.6 s), we are going to reduce this time in our future work considering the re-generation of the trained model.

twoboard required more than twice of *oneboard_v1*. In particular, *twoboard* required 116.1 s from sensing to the completion of detection. The reasons for this result are that *twoboard* required 70.3 s for the serial communication between Lazurite and Raspberry Pi and 25 s for the launching time of Raspberry Pi after power is supplied by the relay module. Because there is no large difference for the inference time and the communication time between *oneboard_v1* and *twoboard*, we are going to reduce the launching time of Raspberry Pi and the serial communication time between Lazurite and Raspberry Pi in our future work. Also, we will consider to generate sound and light by Lazurite regardless of the inference result when something is detected by the motion sensor.

Table 3 represents the average voltage, the current in both operation time and idle time, and the power consumption in both operation time and idle time in *oneboard_v1* and *twoboard*. Similar to the evaluation of the operation time, we evaluated them when we held a picture of the Japanese black bear in front of *oneboard_v1* and *twoboard* ten times. For the evaluation, we used Indoor Corgi Elec. E32-SolarCharger [16]. E32-SolarCharger can control charging power from a solar panel to a lead storage battery and supplying power from the lead storage battery to a micro-computer board. Also, E32-SolarCharger can store the average voltage, current, and power consumption once every minute to a log file.

The difference between the current in the operation time and the current in the idle time in *oneboard_v1* was 0.08A. The reasons why the difference was small are that 1)

Table 3 Average voltage, current, and power consumption in *oneboard_v1* and *twoboard*.

Name	*oneboard_v1*	*twoboard*
Voltage [V]	12.50	12.50
Current in the operation time [A]	0.28	0.33
Power consumption in the operation time [W]	3.50	4.13
Current in the idle time [A]	0.20	0.05
Power consumption in the idle time [W]	2.50	0.63

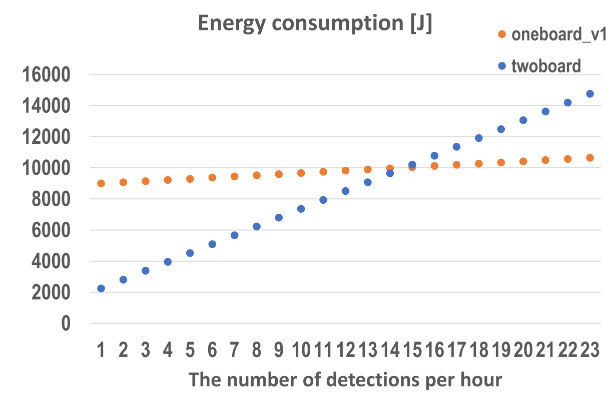


Fig. 7 Energy consumption of *oneboard_v1* and *twoboard* when the number of detections per hour is changed.

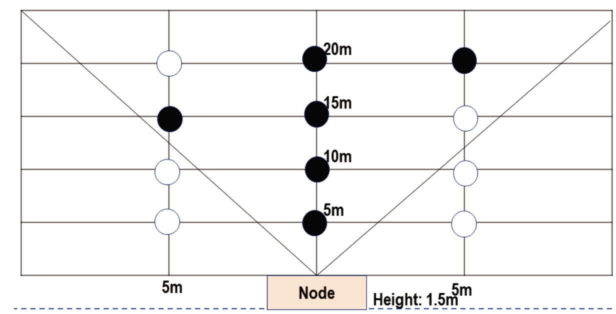


Fig. 8 Detection range of the developed prototypes.

the power was always supplied to Raspberry Pi and 2) only the power supply to the wireless module could be turned on or off.

In *twoboard*, current in the operation time was increased 18% while current in the idle time was decreased 75% compared to *oneboard_v1*. Because power was supplied to both Lazurite and Raspberry Pi in the operation time, the current in the operation time was increased. On the other hand, power was not supplied to Raspberry Pi during the idle time, hence the current in the idle time was largely decreased.

Figure 7 represents the energy consumption of *oneboard_v1* and *twoboard* when the number of detections per hour is changed. Note that this evaluation does not include sensing except Japanese black bears during one hour. In other words, this evaluation is based on the condition that a Japanese black bear is always detected whenever the motion sensor senses. The energy consumption is obtained by the sum of (the power consumption in the operation time * the operation time) and (the power consumption in the idle time * (3600 - the operation time)). If the number of detections per hour is less than 14 times, *twoboard* is better. Since the detection of the Japanese black bear in fields will be irregular, *twoboard* is better in the energy consumption than *oneboard_v1*. Especially, if the number of detections per hour by *twoboard* is less or equal to four times, the energy consumption is less than half of *oneboard_v1*. In such a case, we can use a smaller solar panel and a smaller lead storage battery.

In the evaluation of the operation time and the power consumption, capturing a static image by the camera with the infrared projector was not included. The operation time in both *oneboard_v1* and *twoboard* will be increased 3s when the infrared projector is turned on. This is the time required for the projection of the infrared projector to become stable after power is supplied. The current in the operation time in both *oneboard_v1* and *twoboard* will be increased 0.32A. Therefore, the energy consumption will be increased by 0.96J (3*0.32). Note that the power consumption and the energy consumption in both *oneboard_v1* and *twoboard* in idle time are not changed because power is not supplied to the infrared projector.

3.2 Detection Range, Accuracy of Deep Learning, and Success Rate of Communication through Field Tests

In addition to energy consumption, the detection range, the accuracy of deep learning, and the success rate of communication are important to use the proposed nodes outdoors. We evaluated them through field tests.

Figure 8 represents the detection range by the developed prototypes. Since the same sensor was used, there is no difference among *oneboard_v0*, *oneboard_v1*, and *twoboard*. In the athletic field of the University of Aizu, we placed *oneboard_v0* to 1.5 m higher position than the ground. A man who is 1.7 m in height stood 5 m in front of *oneboard_v0* and then parallel to the baseline, 5 m to the left and right. This was done every 5 m up to the distance of 20 m. We evaluated whether the motion sensor detected the man. In Fig. 8, black circles represent the detected places while white circles represent the places not detected. In the case of the front, the motion sensor detected until 20 m away from *oneboard_v0*.

Japanese black bears walk fields on all fours. In such a case, the height from the ground will be tens of centimeters. Since the difference of the height between man and *oneboard_v0* in this evaluation was 0.2 m (1.7 m–1.5 m), we may obtain the same result when we place the developed prototypes 0.2 m lower height than the height of walking

Table 4 Inference result by deep learning. The number in the table represents the number of captured images.

Bears in images / Detection	Yes	No
Yes	40 (53.3%) ①	22 (29.3%) ③
No	0 (0%) ②	13 (17.3%) ④

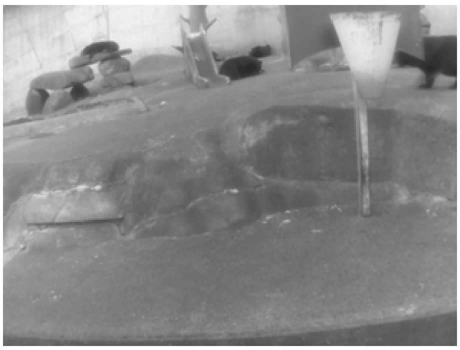


Fig. 9 Example of images where bears were not detected.

Japanese black bears.

We tested the operations of *oneboard_v0* and *twoboard* in a bear zoo located at Kita-Akita City. In the rest of the section, we show the accuracy of deep learning, the success rate of communication between Lazurite and Raspberry Pi, and the success rate of communication between *twoboard* and our server.

Table 4 represents the confusion matrix which results in the inference result by Inception-v3. This result was obtained by placing *oneboard_v0* about one day at the zoo (Sept. 24–25, 2018). We used the trained model of Inception-v3 to check how general CNN models can detect Japanese black bears. Since Inception-v3 has a class category for black bear and American black bear, we regarded that Japanese black bears were detected when the class category was included in the Top-5 result by image classification. ①, ②, ③, and ④ represent "true-positive (TP)", "false-positive (FP)", "false-negative (FN)", and "true-negative (TN)". The accuracy, precision, recall, and f1-score of Inception-v3 in Table 4 were 70.6%, 100%, 64.5%, and 78.4%. Note that as we used the same trained model there is no difference in deep learning among the developed three prototypes.

The reason for lowering accuracy and f1-score came from that 22 times bears were not detected although bears were in the images (i.e., the item ③ in Table 4). Figure 9 displays such an image where bears slept in the background and the image was a gray color due to operation at night. It was difficult for the trained Inception-v3 to detect Japanese bears in such images.

To improve accuracy and f1-score, it is required to re-develop a trained model using such images. In addition, in real cases, the proposed nodes will be used not at the zoo but instead in vegetable fields, livestock huts, and tourist sites. Therefore, we are going to re-develop a trained model by using images captured under various environments in our future work.

We evaluated the communication at the zoo on a different day using *twoboard* (Sept. 25–26, 2019). In *twoboard*, captured images by the camera attached to Lazurite were sent to Raspberry Pi using serial communication. Raspberry Pi in *twoboard* received 105 images for 117 images sent by Lazurite. Therefore, the success rate of communication was about 90%. We consider that 10% failures were caused by the connection between the GPIOs of Lazurite and Raspberry Pi using jumper wires. Data loss might occur if a loose wiring connection existed. To solve this problem, we are going to consider a solid connection between Lazurite and Raspberry Pi.

For the evaluation of the communication between Raspberry Pi and our server using the wireless module, we sent not only the detection information by text, but also static images. Our server received 77 images for 105 images sent by Raspberry Pi. Therefore, the success rate of communication was 73.3%. In addition to the location of the zoo which is located in the mountain area of Kita-Akita City, the used wireless module was located inside the case of *twoboard* due to being non-waterproof. As a result, the sensitivity of the wireless module was low. We are going to solve this problem by using a wireless module with an external antenna.

4. Conclusion

In this paper, we proposed two battery-powered wild animal detection nodes with deep learning. To avoid damages and accidents by wild animals, the proposed nodes automatically detect wild animals using deep learning and notify the detection information to the people concerned. We evaluated the operation time and the power consumption of the proposed nodes to clarify the energy consumption of the proposed nodes. Also, we evaluated the detection range of the proposed nodes, the accuracy of deep learning, and the success rate of communication through field tests to demonstrate that the proposed nodes can be used to detect wild animals outdoors.

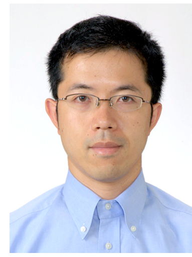
As future work, we are going to shorten the operation time from sensing to detection including the inference time by deep learning. Also, we are going to reconstruct a trained model to improve the accuracy of deep learning. Moreover, we will evaluate the effect of light and sound generation to drive away detected wild animals from the proposed nodes and to notify people near the proposed nodes.

Acknowledgments

The authors would like to thank Dr. Santiprapan in the Prince of Songkla University, Thailand who supported the preparation of the evaluation. Also, the authors would like to thank Dr. J. Brine in New Zealand who checked the English throughout the paper. Finally, the authors would like to thank Kita-Akita City office which manages the visited bear zoo where we tested the developed prototypes.

References

[1] Y. Oishi, "The use of remote sensing to reduce friction between human and wild animals and sustainably banage biodiversity," J. Remote Sensing Society of Japan, vol.36, no.2, pp.152–155, 2016 (in Japanese).
[2] S. Miyasaka, A. Tamura, S. Unome, and N. Yoshida, "Development of the wildlife detection method by using airborne sensor system," J. Remote Sensing Society of Japan, vol.36, no.5, pp.493–500, 2019 (in Japanese).
[3] Campark, "Campark trail camera," https://www.campark.net/collections/frontpage, accessed April 4, 2020.
[4] H. Nguyen, S.J. Maclagan, T.D. Nguyen, T. Nguyen, P. Flemons, K. Andrews, E.G. Ritchie, and D. Phung, "Animal recognition and identification with deep convolutional neural networks for automated wildlife monitoring," Proc. International Conference on Data Science and Advanced Analytics, pp.40–49, 2017.
[5] M.S. Norouzzadeh, A. Nguyen, M. Kosmala, A. Swanson, C. Packer, and J. Clune, "Automatically identifying wild animals in camera trap images with deep learning," Proc. National Academy of Sciences, vol.115, no.25, pp.1–12, 2017.
[6] R. Thangarasu, V.K. Kaliappan, R. Surendran, K. Sellamuthu, and J. Palanisamy, "Recognition of animal species on camera trap images using machine learning and deep learning models," Int. J. Sci. Technol. Res., vol.8, no.10, pp.2613–2622, 2019.
[7] M. Ando, S. Nakatsuka, H. Aizawa, S. Nakamori, T. Ikeda, J. Moribe, K. Terada, and K. Kato, "Recognition of wildlife using deep learning in images taken by camera traps," The Mammal Society of Japan Mammalian Science, vol.59, no.1, pp.49–60, 2019 (in Japanese).
[8] M.S. Norouzzadeh, A. Nguyen, M. Kosmala, A. Swanson, M.S. Palmer, C. Packer, and J. Clune, "Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning," Proc. National Academy of Sciences, vol.115, no.25, pp.E5716–E5725, 2018.
[9] A.G. Villa, A. Salazar, and F. Vargas, "Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks," Ecological Informatics, vol.41, pp.24–32, 2017.
[10] T. Morita, A. Toyoda, S. Aisu, A. Kaneko, N. Suda-Hashimoto, I. Matsuda, and H. Koda, "Animals exhibit personality traits in their movement: a case study on location trajectories of primates," bioRxiv, doi:http://dx.doi.org/10.1101/775486, pp.1–5, 2019.
[11] H. Hayashi, R. Kabata, K. Terada, M. Ueno, and K. Yamada, "Tracking and individual identification of Japanese macaque using deep learning and particle filter," IEICE Technical Report, PRMU2017-84, 2017 (in Japanese).
[12] A.R. Elias, N. Golubovic, C. Krintz, and R. Wolski, "Where's the bear? - Automating wildlife image processing using IoT and edge cloud systems," Proc. IEEE/ACM 2nd International Conference on Internet-of-Things Design and Implementation, pp.247–258, 2017.
[13] Raspberry Pi Foundation, "Raspberry Pi," https://www.raspberrypi.org/, accessed March 18, 2020.
[14] LAPIS Semiconductor, "Lazurite," http://www.lapis-semi.com/lazurite-jp/, accessed March 18, 2020 (Japanese page).
[15] Google Inc., "Inception-v3," https://cloud.google.com/tpu/docs/inception-v3-advanced, accessed March 18, 2020.
[16] Indoor Corgi Elec., "E32-SolarCharger," https://www.indoorcorgielec.com/products/e32-solarcharger/, accessed March 18, 2020 (Japanese page).

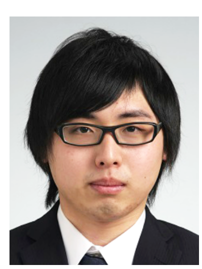


Hiroshi Saito received his B.S. and M.S. degrees in Computer Science and Engineering from the University of Aizu in 1998 and 2000, respectively. In 2003, he received his D.E. degree from the University of Tokyo. He is currently a senior associate professor of the University of Aizu. His research interests include asynchronous circuit design, wireless sensor network applications, and IoT applications.

Tatsuki Otake received his B.S. degree in Computer Science and Engineering from the University of Aizu in 2019. He is currently a Master student at the University of Aizu. His research interests include asynchronous circuit design for FPGAs.

Hayato Kato received his B.S. degree in Computer Science and Engineering from the University of Aizu in 2019. He is currently a Master student at the University of Aizu. His research interests include asynchronous circuit design.

Masayuki Tokutake received his B.S. degree in Computer Science and Engineering from the University of Aizu in 2018. He is currently pursuing a master's course at the University of Aizu. His research interests include image recognition.



Shogo Semba received his B.S. and M.S. degrees in Computer Science and Engineering from the University of Aizu in 2017 and 2019. He is currently a Doctoral student at the University of Aizu. His research interests include asynchronous circuit design.

Yoichi Tomioka received his B.E., M.E., and D.E. degrees from the Tokyo Institute of Technology, Tokyo, Japan, in 2005, 2006, and 2009, respectively. He was a research associate with the Tokyo Institute of Technology until 2009. He was an assistant professor with the Division of Advanced Electrical and Electronics Engineering, Tokyo University of Agriculture and Technology until 2015. He was an associate professor with the School of Computer Science and Engineering, the University of Aizu until 2018. Since 2019, he has been a senior associate professor with the university. His research interests include image processing, hardware acceleration, high-performance computing, electrical design automation, and combinational algorithms.

Yukihide Kohira received his B.E., M.E., and D.E. degrees from the Tokyo Institute of Technology, Tokyo, Japan, in 2003, 2005, and 2007, respectively. He had been a researcher of Department of Communications and Integrated Systems in the Tokyo Institute of Technology from 2007 to 2009. In 2009, he joined the School of Computer Science and Engineering in the University of Aizu, where he is currently as a senior associate professor. His research interests include VLSI design automation and combinational algorithms.