Development of an Intelligent Compact Crawler Robot for House Foundation Inspection

Development of House Inspection Robot

Keiichiro Masuda
Graduate School of Science and Engineering
Hosei University
Tokyo, Japan
keiichiro.masuda.6g@stu.hosei.ac.jp

Goragod Pongthanisorn
Graduate School of Science and Engineering
Hosei University, Tokyo, Japan
po.goragod@gmail.com

Genci Capi
Faculty of Science and Engineering
Hosei University
Tokyo, Japan
capi@hosei.ac.jp

***Abstract*— In this work, we propose a mobile robot for house foundation inspection. The robot can operate in user-controlled mode and in autonomous mode. In user-controlled mode, the developed robot exchange information with the user through a GUI system. In addition, the operator can control the robot remotely. The robot moving trajectory is shown in the developed GUI. In autonomous mode the robot utilizes trained Deep Learning models running in the Raspberry Pi for concrete crack and water leaking detection. The results show that the proposed system is functioning well in the experimental environment and can be further expanded for other implementations.**

Keywords—Crawler robot, Deep Learning, House foundation investigation, damage recognition.

I. INTRODUCTION

In recent years, application of in human life environments is increasing, especially in rescue or human dangerous operations.

On the other hand, Japanese house safety is strongly related with the situations in their foundations. To cope with the hot and humid summer climate, it has an open structure with the characteristics of ventilation and heat shielding. Investigating the Japanese house foundation is difficult because the narrow space between the foundation and the floor. Therefore, the expert must move in a dark environment through narrow spaces (height around 60cm). However, since the foundation is generally not a place to be concerned about, the discovery of foundation troubles is discovered not in time. Therefore, regular and easy foundation inspections are really important for the safety of the house.

Two often foundation damages are crucks and water intrusions. Cracks in the foundation can appear for several reasons, including soil movement, water damage, or shoddy construction. Depending on how severe they are, foundation cracks can let water in, weaken the building, and cause additional damage if not repaired. Water collection around or beneath the foundation may be caused by improper drainage near the foundation or plumbing leaks. Long-term water exposure can undermine the foundation, cause erosion, and foster the spread of mold.

Several research works are focused on developing compact robots for investigation or rescue operations. For example, in [1] a crawler robot (MACbot) that has four track modules that can adapt the terrain changes is presented. It can run four tracks at once and crawl tracks as needed because there are four motors at each track. one crucial area for research and development in rescue robot systems is the study of robotic mechanism models with terrain adaptation and operational capabilities ([2]). However, there are some challenges in conquering high obstacles or steps ([4]).

Michael et al. proposed an integrated interface for robots [5]. They primarily use camera images in their system, which also displays the controlled robot's environment map, sensor data, and buttons that may be used to provide commands to the robot. Additionally, a lot of autonomous rescue robots are being created today [6-9].

In this paper, we introduce an intelligent compact crawler robot for house foundation inspection. By using the robots, house foundation inspection can be done more conveniently and efficiently. The developed robot can operate in user-controlled mode and in autonomous mode. We have developed a GUI for a remote control of the crawler robot. Furthermore, the robot can investigate the foundation situation autonomously. The robot utilized the trained deep learning algorithms [10-13], to identify cracks in the pillars and water leaks. In addition, the robot records its own movement route and creates a movement route diagram autonomously shown in the GUI monitor. The robot utilizes the generated route to return to its initial location.

II. HARDWARE DESIGN

A. Developed system

Fig.1 shows the entire crawler robot system. The operation of the crawler robot is performed by remote control by connecting the Raspberry Pi and a PC terminal via WI-FI communication. In user-controlled mode, the operator sends commands to the robot using the GUI. On the other hand, the robot captured camera image is transmitted to the user's PC.

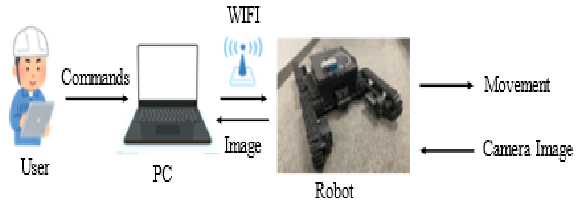
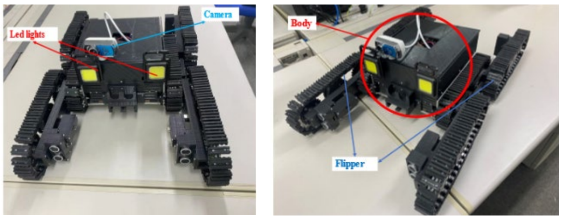


Fig.1. User-robot interface system.



(a) Front view (b) Side view

Fig.2 Developed robot.

Table 2. Robot specification

Length	520 [mm]
Width	320 [mm]
Height	130 [mm]
Weight	4.3 [kg]

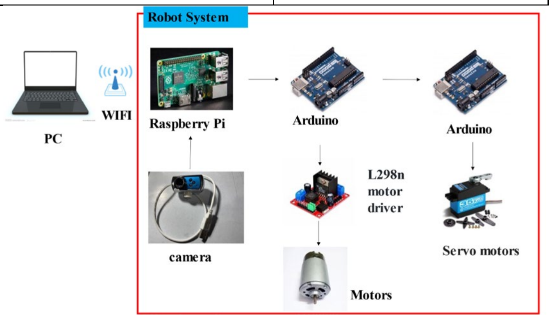


Fig. 3 Robot dataflow.

B. Robot

Based on the operation environments, we developed a compact crawler robot as shown in Fig. 2. The robot consists of the body and four movable flippers. The four flippers are utilized by the robot to climb stairs and steps or pipes that are present in-house foundations. The robot parameters are given in Table1. Main parts such as the flipper part of the robot were manufactured in 3D printer using a PLA to reduce the weight.

There are two main motors for driving the left and right crawlers of the body and four servo motors for driving the four movable flippers [3]. In addition, two LED lights are placed in the robot to lighten the environment.

Fig. 3 shows a schematic diagram of the control system of the robot. Two Arduino boards are used for servo motor control, and main motors. The Raspberry Pi sends the motor commands to the Arduino boards. To operate the robot in user mode, first we connect the Raspberry Pi and the PC via Wi-Fi communication wirelessly.

III. Developed GUI

There are two main parts in the user controlled mode: the operation interface and the image recognition system. To perform wireless remote control, we developed an GUI on the browser, as shown in Fig. 4. The GUI consists of three parts. The buttons in the upper half of the GUI (Fig.4 ①) are the operation buttons for controlling the robot flippers. The buttons on the right (Fig.4 ②) are the buttons for controlling the movement of the robot body. The operator presses buttons on the operation interface to issue commands to the Raspberry Pi. For operation interface we used Flask, which is an operation interface on a browser. Flask is a lightweight customizable framework, which is lighter than other isomorphic frameworks. Using Flask, the GUI pressed buttons are converted into commands and sent to the Raspberry Pi. Raspberry Pi converts commands from the operation interface into signals that Arduino can read. In addition, the robot captured image is shown in the user’s monitor in a separate window.

Because it is a dark enviroment, after some forward, right and left robot motions it is very difficult to know the exact location of the robot. Therefore, we added a window in the GUI (Fig.4 ③) showing the robot motion trajectory. The Canvas element was used to create the movement path. You can draw figures in Html using the Canvas tag. When the operation button of the main motor is pressed, the operation interface autonomously records the operator's instructions and creates a moving path map of the robot based on the instructions. The starting point of the trajectory map is at the origin of the trajectory map. During foundation inspection using a robot, the robot may enter a position that the operator cannot see. It is possible to check the position of the robot even when the robot cannot be seen by moving path map. Also, using this path map, the robot can autonomously returns to its initial location, autonomously. By using the "return" button of the operation interface, the operation interface can change the robot trajectory information into a movement commands, which are sent to the robot. It is also possible to reset the route map by using the "reset Map" button.

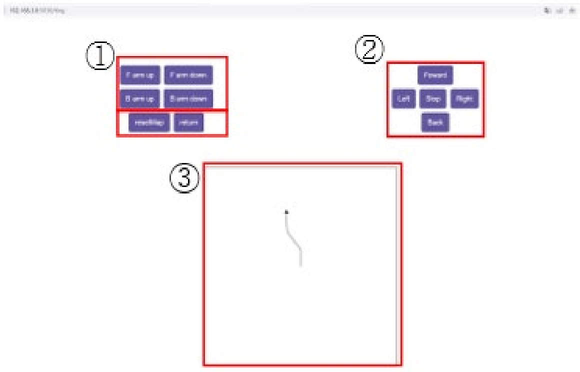
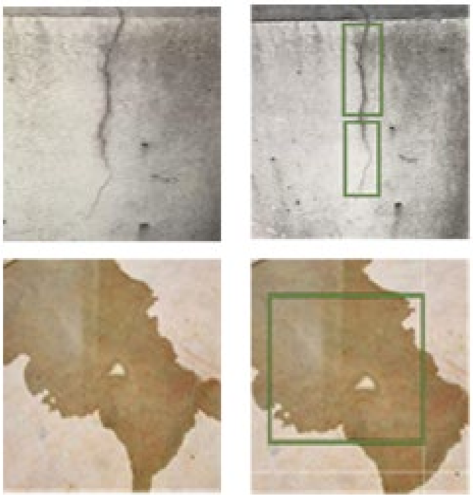


Fig. 4 Developed GUI.



(a) Original mage (b) Labeling image
Fig. 5. Sample images of crack and water database.

IV. Demage recognition

In this work, two demeges of the house oundations are considered: cracks and water leaks. For cracks and water leaks recognition we trained deep neural networks. DNNs are machine learning methods that uses multi-layer networks. In our implementation we used ensorFlow Lite which is an open source software library for machine learning in mobile devices such as Raspberry Pi.

For training the DNNs we labeled cracks and water leaks to create the dataset, as shown in Fig. 5. A total of 165 crack images and 151 water leak images were created. 252 are used as training images, 31 validation images, and 33 test images. The model performance is evaluated by the precision and the recall equations as follows:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

where TP is the true positive rate, FP is false positive rate, and FN is False Negative.

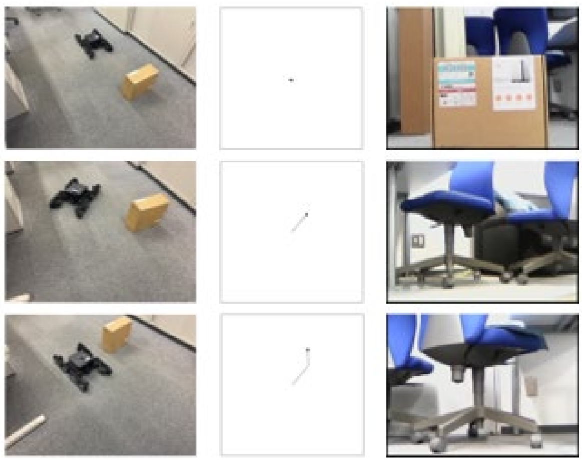
V. EXPERIMENTAL RESULTS

Initially we evaluated the developed GUI for different robot trajectories. Fig. 6 shows the robot motion controlled by the user relying only in the robots cuptured image.

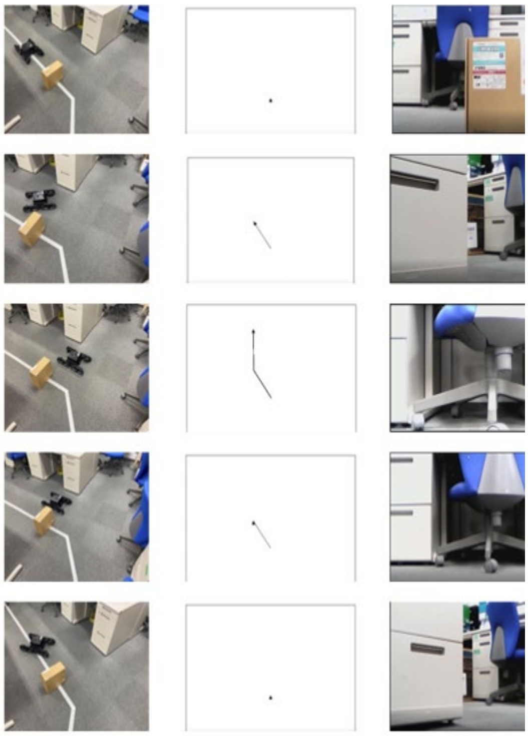
We also tested experimentally the ability of the robot to return to its initial location after moving autonomously in the enviroment. The robot automatically returned to the initial position using the "return" button (Fig. 7). Then, the robot's actual path, the map path and the initial and final positions were compared. The experiments were performed three times. The experimental results showed that the distance error between robot nitial and returned final locations were in a range of 33-37 mm.

Relation between precission and recall are shown in Fig. 8. Using the TensorFlow Lite's image recognition program, the DNNs achieved an accuracy of 76% and 63% for cracks and water leaks, respectively. In this experiment, we used 10 images of cracks and 10 images of water leaks to evaluate the actual performance of. The experiment is shown in Fig. 9. This performance evaluation is an experiment of recognition accuracy of the program. The evaluation criterion for recognition accuracy is the accuracy of program judgment.

As an experimental environment, we reproduced a dark house foundation situation in the Human Assistive Robot Laboratory on the Koganei Campus of Hosei University. We also added columns with cracks and ground with water leaks. The planned movement route is shown in Fig. 10(a). The experimental environment is shown in Fig. 10(b). Only the light of the robot was used in the experiments, as shown in Fig. 10(c).

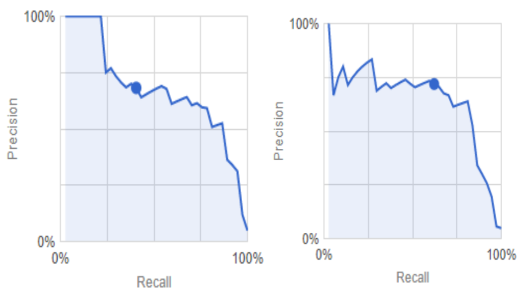


(a) Robot image (b) GUI image (c) Robot camera image
Fig. 6 Generated robot route in the GUI monitor.



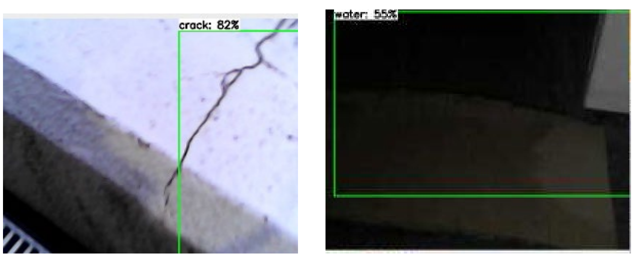
(a) Robot image b) GUI image (c) Robot camera image

Fig. 7. Robot autonomous motion.



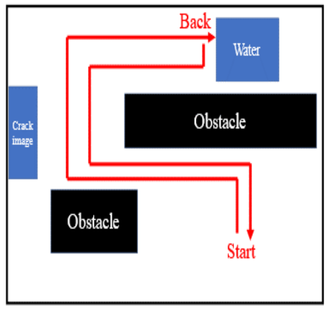
(a) Crack model (b) Water model

Fig. 8. Model evaluation results.

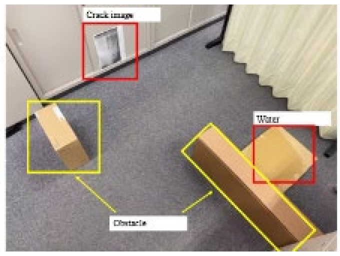


(a) Crack recognition by the robot (b) Water recognition by the robot

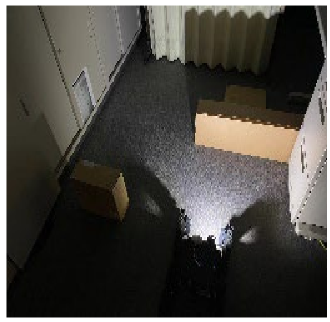
Fig. 9. Object detection result.



(a) Environment layout



(b) Real environment



(c) Dark environment

Fig. 10. Experimental environment.

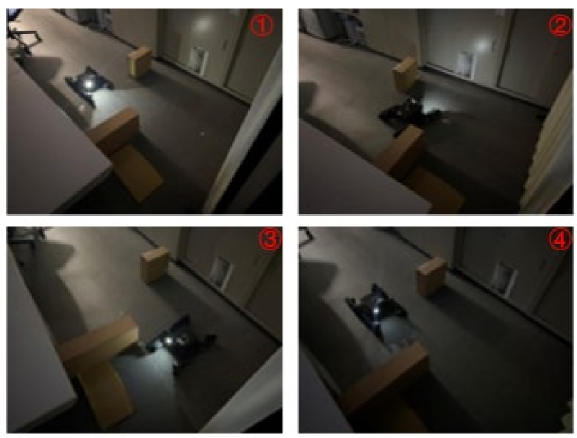
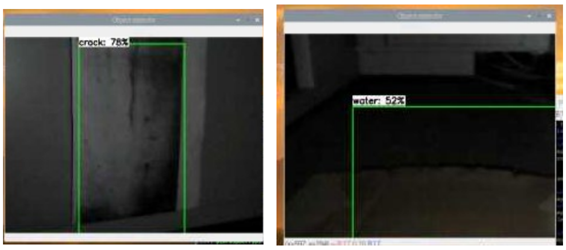


Fig. 11. User controlled robot using GUI and autonomous navigation.



(a) Crack (b) Water leak

Fig. 12. Damage detection.

In the experiments, the robot was controlled using the GUI by the user based on the image shown in the monitor. The robot detects cracks and water leaks in the experimental environment, and finally returns to its initial position autonomously.
Fig. 14 shows the video capture of the robot operation in the environment. After avoiding two obstacles and reaching the target point, the robot autonomously returns to the initial position. The error between the final position and initial position, even in this complicated environment, was 55 mm.

While the robot moves in the environment, a crack on the wall was detected with a recognition rate of 78%, as shown in Fig. 12. Finally, a water leak was detected with a recognition rate of 52% (Fig.15). In average, the average recognition accuracy for water leaks and cracks exceeded 60%.

VI. Conclusion

In this work, we presented an intelligent crawler robot for house foundation inspection. The operator can remotely control the robot. For remotely operation of the robot, the Raspberry Pi and a personal computer were connected via WIFI communication. The remote-control system operated on the browser using Flask. In addition, the movement path map was recorded by the robot, returning autonomously to its initial location even in a complicated route. For the damage recognition, such as cracks and water leaks, we used TensorFlow Lite using images streamed from the webcam. As a result, it was found that water leakage and cracks were recognized even in dark environments.

However, some issues must be considered in the future works, such as: 1) Improving the recognition accuracy of deep learning algorithms by collecting more data and testing other DL architectures; 2) Increasing the kind of damages recognized by the robot; 3) Improving the algorithm for robot localization in the environment.

References

[1] Quy-Hung Vu, Byeong-Sang Kim and Jae-Bok Song, "Autonomous stair climbing algorithm for a small four-tracked robot," 2008 International Conference on Control, Automation and Systems, Seoul, Korea (South), 2008, pp. 2356-2360.

[2] Zhenzhong Yu, Kaiti Cai, Wei Liu, Weicou Zheng, Review of Rescue Robot Technology, Journal of Jiangnan University (Natural Science Edition), 2015,14(04): 498-504.

[3] Hui Zhang, Xiangdong Cai, Dan Hai, Dengke Zhu, Shaoke Qian, Xun Li, Zhiqiang Zheng, NuBot rescue robot overall design , Robot Technique and Application, 2010(04): 17-19.

[4] Homma, K., Yamada, Y., Matsumoto, O., Ono, E., Lee, S., Horimoto, M., & Shiozawa, S. (2009, June). A proposal of a method to reduce burden of excretion care using robot technology. In *2009 IEEE International Conference on Rehabilitation Robotics* (pp. 621-625). IEEE.

[5] Michael Baker, Robert Casey et al. " Improved interfaces for human-robot interaction in urban search and rescue.", SMC (3) 2004: 2960-2965.

[6] Yugang liu, Goldie Nejat, "Robotic urban search and rescue: A survey from the control perspective", Journal of Intelligent & Robotics Systems, Vol. 72, Issue 2, pp. 147-165, 2013.

[7] David Erdos, Abraham Erdos, Steve E. Watkins, "An Experimental UAV System for Search and Rescue Challenge", IEEE Aerospace and Electronic Systems Magazine, Vol.28, Issue 5, pp.32-37, 2013.

[8] Teodor Tomic, Korbinian Schmid, Philipp Lutz, Darius Burschka,Toward a Fully Autonomous UAV: Research Platform for Indoor and Outdoor Urban Search and Rescue", IEEE Robotics & Automations Magazine, Vol.19, Issue.3, pp. 46-56, 2012.

[9] Ivan Vasilyev, Alla Kashourina, Maxim Krasheninnikov, Ekaterina Smirnova, "Use of Mobile Robots Groups for Rescue Missions in Extreme Climatic Conditions", 25th DAAAM International Symposium on Intelligent Manufacturing and Automation, pp. 1242 –1246, 2014.

[10] T. Duc Dung, and G. Capi, Application of Neural Networks for Robot 3d Mapping and Annotation Using Depth Image Camera, International Journal of Robotics and Automation, Vol. 37, Issue 6, 2022.

[11] 5. S. Nilwong, D. Hossain, S. Kaneko, G. Capi, Deep Learning-Based Landmark Detection for Mobile Robot Outdoor Localization. Machines 2019, 7, 25.

[12] D. Hossain, G. Capi, "Multiobjective Evolution of Deep Learning Parameters for Robot Manipulator Object Recognition and Grasping", Advanced Robotics, 32(20): 1090-1101.

[13] W. Qi, Y. Chun, Y. Sheng, G. Zhao and L. Wang, "Stability Analysis of Obstacle Avoidance Ability and Environment Adaptation Modeling of Snake Robot based on Deep Learning and Binocular Vision," 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2022, pp. 299-3024.