Sensors and Materials, Vol. 31, No. 6 (2019) 1905–1916
MYU Tokyo

S & M 1906

Small Gardening Robot with Decision-making Watering System

Pikulkaew Tangtisanon*

Department of Computer Engineering, Faculty of Engineering,
King Mongkut's Institute of Technology Ladkrabang,
1 Chalongkrung Rd., Ladkrabang, Bangkok 10250, Thailand

(Received November 1, 2018; accepted February 26, 2019)

Keywords: gardening robot, fuzzy, android, smartphone, small robot

At present, people who live in the countryside tend to move downtown in order to get jobs. As a consequence, only elders and children live in their hometowns with plenty of land left uncultivated. The aim of this research is to build cheap small gardening robots to help people grow plants in small yards as a hobby via a long-distance communication system. In this research, two robots composed of Raspberry Pi and ESP32 microcontrollers, which are low-price controller boards and convenient to buy within the country, were constructed. The robots can be controlled from long distances using Android smartphones. The capacities of the two robots with the two types of microcontroller are compared and discussed. To measure the soil moisture content, two types of moisture sensor, which are capacitive and resistive sensors, were implemented in this project. There are two main functions of the proposed model, which are weed cutting and watering plants. Moreover, a decision-making watering system was implemented and connected to moisture sensors and sprinkler controllers placed in the user's garden. The robots were placed in the northern region of Thailand, while the user stayed in the central region and remotely controlled them with a smartphone for three months. The results show that the automatic watering system is better than a manual watering system since the plant growth rate for the automatic watering system was 20% higher than that for the manual watering system.

1. Introduction

Global warming[1] has become one of the most critical issues in this era. One basic factor that causes climate change is the increase in the level of carbon dioxide in the atmosphere. Many studies have shown that oxygen from plants can help absorb carbon dioxide. However, at present, many teenagers and others of working age have moved to work in downtown areas and left an enormous amount of wasteland in their hometowns.[2] To reduce the global warming crisis, it is vital for everyone who owns wasteland to make use of it. However, when cultivating plants, it is important to eliminate weeds. Normally, people use herbicides to kill weeds, which negatively affect the environment. Such weed killers destroy water resources and natural habitats of animals, and consumers become sick or die because of food products containing herbicide residues. To help solve the above-mentioned problem, this project was carried out to

*Corresponding author: e-mail: pikulkaew.ta@kmitl.ac.th
https://doi.org/10.18494/SAM.2019.2176

ISSN 0914-4935
https://myukk.org/

reduce the amount of labor required to grow vegetables and encourage people who live or work far away from their land to remotely grow home-grown vegetables in small gardens as a hobby. The most important reason why people do not grow vegetables remotely is because commercial gardening robots are expensive. In this project, small gardening robots that are composed of low-price controller boards and sensors that easy to buy within the country were built. In Thailand, there are three main popular microcontroller boards, which are Arduino, ESP32, and Raspberry Pi. In this paper two microcontrollers are compared in order to guide users to buy the controller according to their needs.

Many studies have been carried out to reduce the labor involved in growing vegetables. For example, Blender *et al.*[3] proposed the use of small robots that can work together as a team in a seeding task in large-scale farming. This research approach helps increase the quantity of farm products per acre since the robots can seed plants on land through an appropriate method. Nevertheless, it requires high investment; thus, such an approach is suitable for large-scale farming only. Prema *et al.*[4] proposed a simulation program with a fuzzy logic model that can be applied to seeding and soil moisture sensing. However, they only tested the efficiency of the program through virtual instrumentation. Many researchers[5–7] have also included smartphone technology to improve the productivity of farms.

In my previous study,[8] a small gardening robot was developed. The robot could be controlled remotely within a network segment. It could be used to cut weeds and measure the soil moisture content. The user could determine the soil moisture content in real time since information was sent to a cloud server. Moreover, the algorithm for deciding whether to water the plants or not was implemented using a fuzzy variable set model. However, the cutting weed function was limited, that is, only thin weeds with a stem width of no more than 5 mm could be cut. The previous gardening robot ran smoothly in a garden with short grass (no taller than 5 cm) and on dry soil with a smooth surface but could not move on sloped or rough ground. The forecasting function could predict whether the user should water the plants or not but the user had to manually control the sprinkler.

In this study, the capacities of two small gardening robots with two types of microcontroller are compared and discussed. A user can control the robots remotely via different network segments with a smartphone. The robots have been improved in various aspects; for example, they can cut thicker weeds, run on sloped or rough ground, and fertilize plants. A fuzzy variable set model is implemented to help the user decide how long the sprinkler should be turned on.

2. Materials and Methods

Figure 1 shows an overall view of the system used in this study. The user can control a sprinkler to water plants in real time through a cloud server. Moreover, the user can automatically set the proposed model to decide whether the sprinkler should be activated or not and how long it should be turned on. The robot can be remotely controlled by the user to move or cut weeds. Two gardening robots with different controllers in the proposed system were built. Figure 2 shows a demonstration plot with the positions of sensors and robots. The

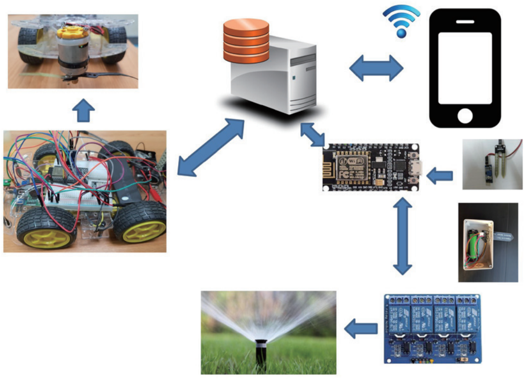


Fig. 1. (Color online) Overall view of the system.

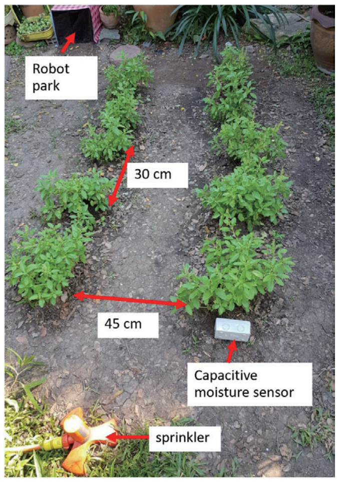


Fig. 2. (Color online) Positions of sensors and robots in an actual plot.

plot size is 3 × 2 m. The resistive soil moisture sensor was attached to one of the robots and the capacitive soil moisture sensor was placed in the plot. Corrosion would occur if the resistive moisture sensor was placed in the soil or water all the time. However, since the sensor was attached to the robot, it mostly stayed in a dry environment. The accuracy of the resistive soil moisture sensor is not so good, but it can be plugged into the soil more easily than the capacitive sensor owing to the shape of the sensor. The resistive soil moisture sensor attached to the robot allows the user to check the soil moisture content in different parts of the garden. When the sensor is not in use, the controller automatically switches to the sleep mode to avoid corrosion. However, the resistive sensor must be changed after every 2 h of operation. An experiment was conducted in Chiang Mai, northern Thailand. Residual soil in this area is mostly derived from tertiary sedimentary rocks or igneous rocks such as granite that break down to produce clay and salty clay soil.[9] The temperature in the garden ranged from 21 to 35 °C. Basil was grown from seedlings produced from seeds of holy basil (*Ocimum basilicum L.*) sold by Chia Tai Company. Plants were grown in 10 holes per plot (five holes per column) with 30 cm space between holes and 45 cm space between rows. Two plots with the same environment were prepared to measure the efficiency of the watering system.

2.1 Software

To control a robot with the ESP32 microcontroller, which has a built-in Wi-Fi module, a user must install an android application in an Android-based smartphone developed and deployed in the Blynk server, which is an IoT cloud server. The minimum requirement for the smartphone is that it is operated with API 15 or higher with an Internet connection. After running the application, the smartphone receives data from the controller attached to the robot. To control a robot with Raspberry Pi, which is a tiny low-cost computer, a user must connect to WebIOPi via a browser such as Chrome. Software was developed to allow the user interface to control Raspberry Pi via Wi-Fi. It can operate GPIO from a browser and run the Python program in Raspberry Pi. Figure 3 shows the user interface for a Raspberry Pi-based robot.

Fig. 3. (Color online) User interface for a Raspberry Pi-based robot.

2.2 Hardware

The robots were composed of Raspberry Pi, ESP32, a camera module, a Pi camera, a motor, motor driver module, a breadboard, a wheel, a plastic cutter, a blade, two Panasonic NCR18650B batteries (3.7 v 3400 mAh) per robot, a power bank, a plate, a soil moisture sensor, and a switch. The cost of the robot is less than 100 dollars for the Raspberry Pi-based robot and 60 dollars for the ESP32-based robot, the former robot is more expensive but its performance is better and it can be used to perform high-level computations such as image processing. Figure 4 shows the robot controlled by Raspberry Pi. Figure 5 shows the robot controlled by ESP32. Both robots are 15 × 26 cm^2, where the one with the Raspberry Pi controller has a weight of 852 g and the one with the ESP32 controller has a weight of 788 g. The sprinkler and soil moisture sensor were controlled by ESP32 via a cloud server.

2.3 Fuzzy logic

Fuzzy logic[10,11] is usually applied in a task whose truth values are composed of real numbers between 0 and 1. Let the x-axis be a universe of discourse and the y-axis be the degree of membership in the [0,1] interval; membership functions can be used to represent a fuzzy set. x represents a real value (Crisp value) within the universe of discourse. In the proposed system, x is the soil moisture content and a, b, c, and d represent the x-coordinates of the four corners of the trapezoidal. Values should be validated under the following condition: $a < b < c < d$.

In this research, triangular and trapezoidal membership functions are used to decide how long the sprinkler should be turned on. The triangular and trapezoidal membership functions are defined by Eqs. (1) and (2), respectively.

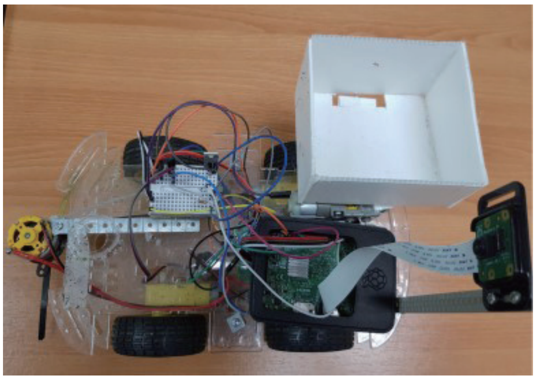


Fig. 4. (Color online) Gardening robot with Raspberry Pi.

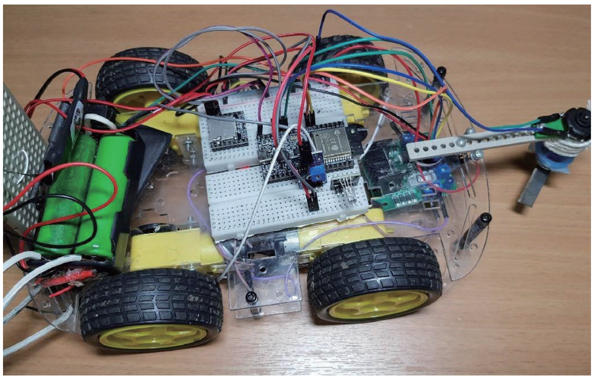


Fig. 5. (Color online) Gardening robot with ESP32.

$$f(x:a,b,c)=\begin{cases}0, & x<a\\ \dfrac{x-a}{b-a}, & a\le x<b\\ \dfrac{c-x}{c-b}, & b\le x\le c\\ 0, & x>c\end{cases} \tag{1}$$

$$f(x:a,b,c,d)=\begin{cases}0, & x<a\\ \dfrac{x-a}{b-a}, & a\le x<b\\ 1, & b\le x\le c\\ \dfrac{d-x}{d-c}, & c\le x<d\\ 0, & d\le x\end{cases} \tag{2}$$

The input to the model comes from the soil moisture sensor values. The time until the plants are to be watered is divided into six functions. Let W+ be the membership function of superwet, W the membership function of wet, W− the membership function of slightly wet, D+ the membership function of superdry, D the membership function of dry, and D− the membership function of slightly dry. However, the soil moisture content can vary and depend on a controller board and input voltage. In this experiment, the soil moisture content was calibrated to output a value between 0 and 1023 as shown in Fig. 6. After defining the membership function, the output can be calculated according to the rule based on Eq. (3). Table 1 shows an example of each membership function. To determine the relationship between the input and output, a rule-based table was built.

$$\text{Soil state} = \text{weight membership } [\mathrm{MAX}(W, W+, W-, D, D+, D-)]. \tag{3}$$

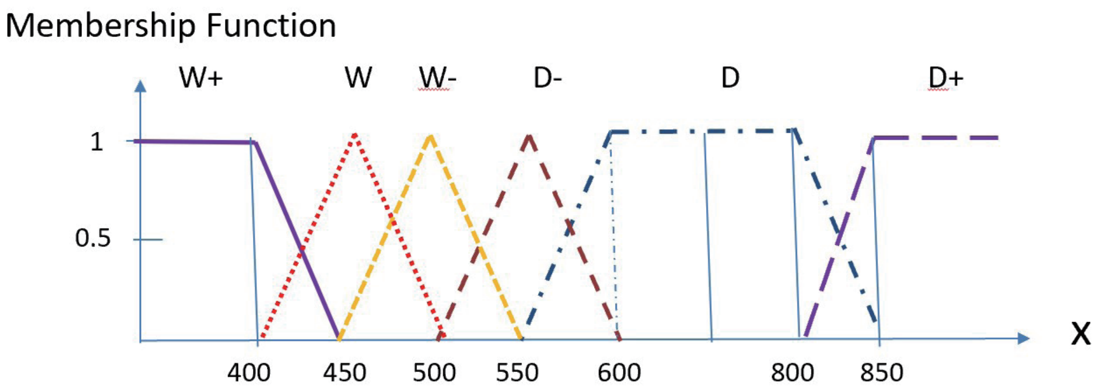


Fig. 6. (Color online) Membership functions.

Table 1
Sample outputs from the fuzzy model.

Input from sensor	Weight membership function						Fuzzy
	W+	W	W−	D−	D	D+	
200	1	0	0	0	0	0	W+
900	0	0	0	0	0	1	D+
440	0.2	0.8	0	0	0	0	W
510	0	0	0.8	0.2	0	0	W−
560	0	0	0	0.8	0.2	0	D−
700	0	0	0	0	1	0	D

W+ (x;0,0,400,450) = If $0 \leq x \leq 400$ then 1 (True)
If $400 < x < 450$ then $(450 - x)/(450 - 400)$
If $450 \leq x$ then 0 (False)

D+ (x;800,850,1023,1023) = If $850 \leq x \leq 1023$ then 1 (True)
If $800 < x < 850$ then $(850 - x)/(850 - 800)$
If $x \leq 800$ then 0 (False)

D (x;550,600,800,850) = If $x < 550$ then 0 (False)
If $550 \leq x \leq 600$ then $(x - 550)/(600 - 550)$
If $600 \leq x < 800$ then 1 (True)
If $800 \leq x \leq 850$ then $(850 - x)/(850 - 800)$
If $850 \leq x$ then 0 (False)

W (x;400,450,500) = If $x \leq 400$ then 0 (False)
If $400 < x < 450$ then $(x - 400)/(450 - 400)$
If $450 \leq x < 500$ then $(500 - x)/(500 - 450)$
If $500 \leq x$ then 0 (False)

$$
\begin{aligned}
W-(x;450,500,550) = \; & \text{If } x \le 450 \text{ then } 0 \text{ (False)} \\
& \text{If } 450 < x < 500 \text{ then } (x - 450)/(500 - 450) \\
& \text{If } 500 \le x < 550 \text{ then } (550 - x)/(550 - 500) \\
& \text{If } 550 \le x \text{ then } 0 \text{ (False)} \\
D-(x;500,550,600) = \; & \text{If } x \le 500 \text{ then } 0 \text{ (False)} \\
& \text{If } 500 < x < 550 \text{ then } (x - 500)/(550 - 500) \\
& \text{If } 550 \le x < 600 \text{ then } (600 - x)/(600 - 550) \\
& \text{If } 600 \le x \text{ then } 0 \text{ (False)}
\end{aligned}
$$

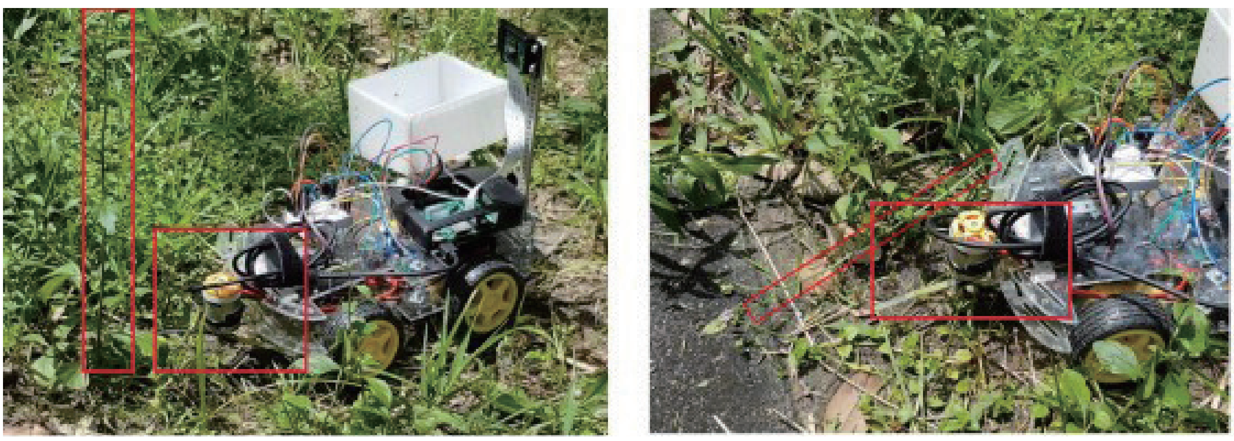
3. Real-world Experiment and Discussion

3.1 Android-based gardening robot

The two robots were placed in the northern region of Thailand, while the user sent commands to control the robots from the central region for three months. The commands were sent from a smartphone or computer, such as cut the weeds or move forward, to control the robots remotely. A video stream was sent from a camera attached to the robot to a server then accessed by the user through a smartphone.

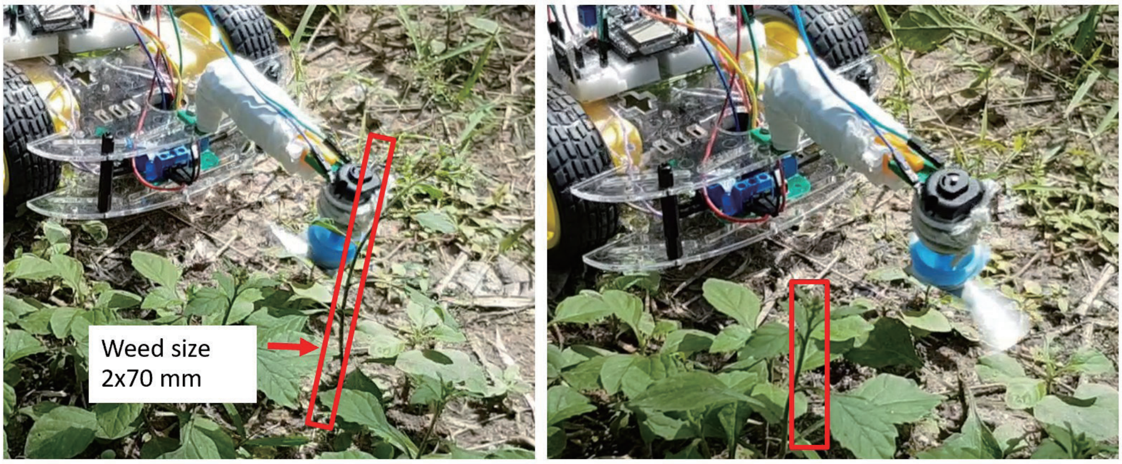
3.1.1 Weed cutting

There are two tools attached to the robots that were used to cut the weeds, namely, a trimmer line and a cutter blade, as shown in Figs. 7 and 8, respectively. The two tools were used to cut the same species of weeds in the size range of 2–10 mm three times per size. The results of the experiment show that the trimmer line is more powerful than the cutter blade since the trimmer line can cut weeds with a stem width of more than 8 mm, while the cutter blade can cut only those with a stem width of up to 5 mm as shown in Table 2.



(a) (b)

Fig. 7. (Color online) Weeds (a) before cutting with a trimmer line and (b) after cutting with a trimmer line.



(a) (b)

Fig. 8. (Color online) Weeds (a) before cutting with a cutter blade and (b) after cutting with a cutter blade.

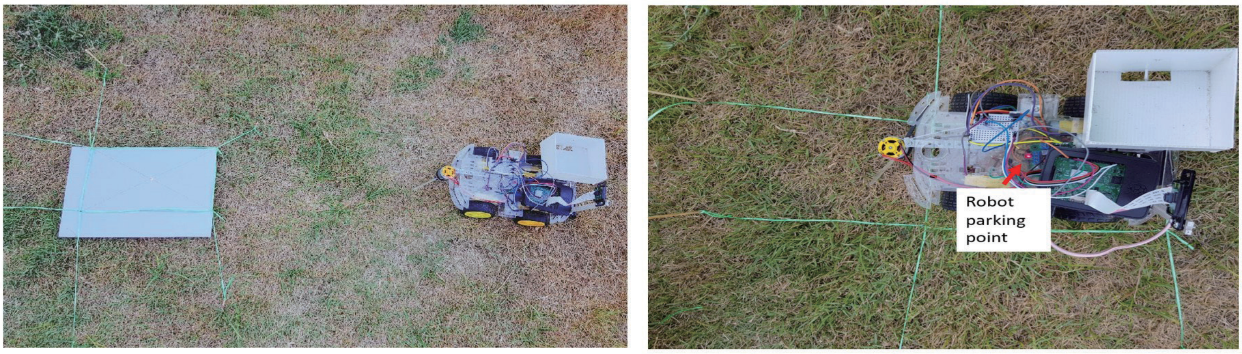
Table 2
Result of experiment on cutting weeds.

Size of weed (mm)	A cutter blade #1	A cutter blade #2	A cutter blade #3	A trimmer line #1	A trimmer line #2	A trimmer line #3
2	yes	yes	yes	yes	yes	yes
3	yes	yes	yes	yes	yes	yes
4	yes	yes	yes	yes	yes	yes
5	yes	no	yes	yes	yes	yes
6	no	no	no	yes	yes	yes
7	no	no	no	yes	yes	yes
8	no	no	no	yes	yes	yes
9	no	no	no	no	yes	no
10	no	no	no	yes	no	no

3.1.2 Robot movement

The experiment was performed three times by letting the robots move forward 50 cm in a straight line along different surfaces (grass, wet grass, soil, and wet soil) then park in a rectangular check point (15 × 26 cm^2). Figure 9 shows the results of the experiment. A laser pointer was used to mark the robot parking position based on the center point of the robot. The accuracy of the robot's movement was computed as the Euclidean distance. Let a be the center point of the marker and b be the center point of the robot. If $a = (a_1, a_2)$ and $b = (b_1, b_2)$ then the distance is given by

$$d(a, b) = \sqrt{(b_1 - a_1)^2 + (b_2 - a_2)^2}. \quad (4)$$



(a) (b)

Fig. 9. (Color online) (a) Set up position and (b) parking position.

Table 3
Average Euclidean distance on different surfaces.

	Raspberry Pi (cm)	ESP32 (cm)
Soil	3.2	5.4
Wet soil	4.5	5.8
Grass	3.1	6.2
Wet grass	4.1	6.7

The results in Table 3 indicated that the Raspberry Pi-based robot can be moved more accurately to every point on each surface than the ESP32-based robot. This is caused by a delay in the command sent to ESP32 since the user must send a command to move the robot to an external cloud server, and also the external camera attached to the robot presents a delayed video stream; thus, it is very difficult to precisely control the robot. The Raspberry Pi-based robot is equipped with a Raspberry Pi camera that streams a video of good quality; thus, the user can easily control the position of the robot.

3.2 Watering system

In the previous proposed work, a user can request to view a forecasting system model based on the fuzzy variable set model to support his or her decision whether to water the plants or not. However, the user must manually water the plants in the final step. The proposed system received a value from a soil moisture sensor as an input to the fuzzy model and automatically turned on and off the sprinkler in an appropriate range of time. The received voltage was converted to a value range of 0−1023 (10 bit number) as shown in Fig. 10. The crop coefficient (K_c) values for basil are as follows: growth stage 0.45 ± 0.02 (25 days after transplant, dat); maturity stage 0.59 ± 0.02 (26–50 dat); and senescence stage, 0.42 ± 0.03 (51–71 dat) according to Daza *et al.*[12] The fuzzy model output six soil states according to K_c and the sprinkler was turned on according to the output from the fuzzy model as shown in Table 4. The experiment was performed by firstly preparing two plots for holy basil with 10 holes in each plot. Subsequently, the user manually sent a command to water the holy basil plot, whereas an automatic watering

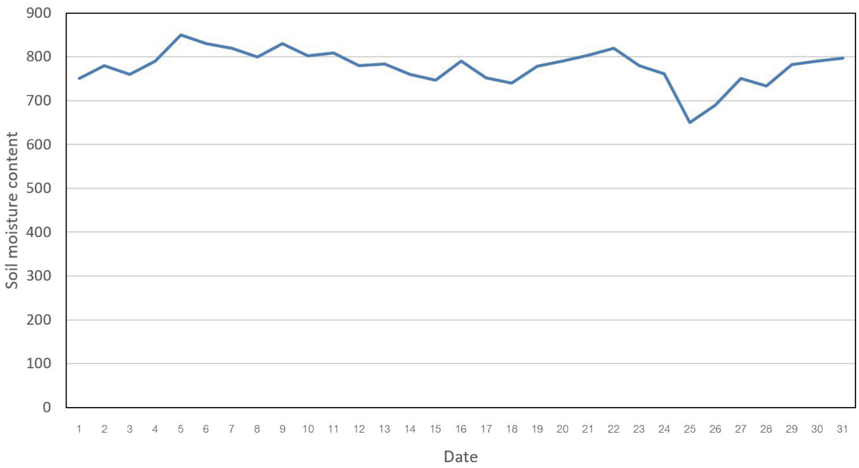


Fig. 10. (Color online) Soil moisture content at 6 am in one month.

Table 4
Range of time to operate a sprinkler based on the output from the fuzzy model.

Fuzzy	Turn on sprinkler (min)
W+	2
D+	15
W	4
W−	6
D−	8
D	10

system was used to water the other plot. Watering was carried out for three months. To save battery power, the soil moisture sensor sends soil moisture content to the server at 6 a.m. for 10 s every day, then automatically enters the sleep mode. The server calculates the average soil moisture content, sends it to the decision-making watering system, waits for an output, and sends commands to turn the sprinkler on or off according to the output.

The growth rate of basil was measured as the average mass of fresh basil. In the manually watered plot, the average mass of the fresh basil was 39 g. In the automatically watered plot, the average mass of fresh basil was 49 g. The result shows that the automatic watering system is better than the manual watering system since the plant growth rate for the automatic watering system is 20% higher than that for the manual watering system.

4. Conclusions

In this study, the functions of two small gardening robots that can help users work on a small garden are described. The robots and an automatic watering system can be controlled remotely. A limitation of the robots is that when the Wi-Fi signal is weak or if the robots move too far from the hotspot, then the user cannot control them to move back to the base station via a smartphone. Moreover, the batteries of the robot must be charged every after 3 h of operation. In a small garden, weeds should be cut at least three times per week, taking 20 min each; thus, the batteries must be charged every three weeks. When applying the decision-making watering system based on a fuzzy variable set model the automatic watering strategy, the plant growth rate for the automatic watering system was 20% higher than that for the manual watering system. The Raspberry Pi-based robot can work with higher efficiency than the ESP32-based robot, since it can move more accurately with the Raspberry Pi camera module; nevertheless, its investment cost is higher than that of the ESP32-based robot. To protect sensors and controllers from water, moisture sensors were placed in the plot connected with ESP32 in waterproof plastic boxes, and controllers were shielded with epoxy resin. Since these robots are prototype versions that are designed to be adjustable upon demand, the wires that connect the controller boards with the driving circuits are not soldered. Thus, they could fall off when the robot runs on a bumpy surface, so it would be better to solder all wires. To protect robots from rain, they were stored in a box with a plastic cover and placed at a dry location. However, this protection can only protect the robots from normal rain and not a storm. The robots were composed of cheap and easy-to-buy sensors and controllers so their replacement would be very easy and convenient. To remove dust and clay from the wheels, it is recommended that every two weeks the user pulls the wheels out of the robot and washes them with water. To remove dust from the controller board, the user should use a low-pressure blower to blow dust out of the controller board before using a clean fabric to remove the clay from the robot by wiping. The user should use epoxy resin or a silicone adhesive to protect the sensors and controller boards from water. The user can also plug more sensors into the existing controller board according to their needs. In future studies, the images from the camera could be used as an input to a machine learning model to detect infected plants and determine how to treat them.

Acknowledgments

The research presented in this paper is supported by the Coordinating Center for Thai Government Science and Technology Scholarship Students (CSTS) and the National Science and Technology Development Agency (NSTDA). The author thanks to H. Kawase, Y. Onishi, T. Veerakul, and R. Sarayut for their contribution to this project.

References

1 Intergovernmental Panel on Climate Change (2007). Climate Change 2007: Synthesis Report (SYR): Intergovernmental Panel on Climate Change.
2 International Labour Organization (2015): ILO Global Estimates of Migrant Workers: Results and Methodology. Geneva: International Labour Office.
3 T. Blender, T. Buchner, B. Fernandez, B. Pichlmaier, and C. Schlegel: IECON 2016 - 42nd Annu. Conf. IEEE Industrial Electronics Society (IECON, 2016) 6879.
4 K. Prema, N. Senthil Kumar, S. S. Dash, and S. Chowdary: IEEE Int. Conf. Advances in Engineering, Science and Management (ICAESM, 2012) 196.
5 A. Arturo, M. Borja, G. Daniel, D. María-Paz, and T. Javier: Sensors **15** (2015) 21204.
6 J. Qian, X. Yang, X. Wu, B. Xing, B. Wu, and M. Li: Comput. Electron. Agric. **166** (2015) 101.
7 P. Han, D. Dong, X. Zhao, L. Jiao, and Y. Lang: Comput. Electron. Agric. **123** (2016) 232.

8 P. Tangtisanon: 14th Int. Conf. Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON, 2017) 722.
9 Geological Survey Division (1983). Geological map of Thailand. Department of Mineral Resources.
10 R. Singh and A. D. Dwivedi: Int. J. Sci. Eng. Res. **7** (2016) 860.
11 L. Xu, R. Zhao, and Y. Ning: Two-Person Zero-Sum Matrix Game with Fuzzy Random Payoffs: Lecture Notes in Computer Science (Springer, Berlin, 2006) p. 809.
12 M. C. Daza, P. C. Arias-Prado, A. Reyes, and N. U. Cobo: Ingeniería e Investigación **37** (2017) 8.

About the Author

Pikulkaew Tangtisanon received her B. Eng. and M. Eng. degrees in information technology engineering from King Mongkut's Institute of Technology Ladkrabang, Thailand, in 2003 and 2005, respectively. She continued her study and received her Ph.D. degree from Tokai University, Japan, in 2009. Since 2009, she has been a professor at King Mongkut's Institute of Technology Ladkrabang. Her research interests are in the Internet of Things, bioengineering, and sensors.
(pikulkaew.ta@kmitl.ac.th)