Robot Monitoring and Controlling Soybean Field Soil Condition Based On K-Nearest Neighbor Algorithm and Message Queuing Telemetry Transport Protocol

Dania Eridani
Department of Computer Engineering
Diponegoro University
Semarang, Indonesia
email: dania@ce.undip.ac.id

Adian Fatchur Rochim
Department of Computer Engineering
Diponegoro University
Semarang, Indonesia
email: adian@ce.undip.ac.id

Jonathan Imago Dei Gloriawan
Department of Computer Engineering
Diponegoro University
Semarang, Indonesia
email: jonathanidg@student.ce.undip.ac.id

***Abstract*—Soybean production is decreasing every year. The level of soybean production is strongly influenced by soil moisture. The problem is that farmers let soybeans grow without adequate maintenance, including without checking the soil moisture. Therefore, an autonomous robot is built that could replace the role of farmers in caring for soybeans. This robot is built to monitor the conditions of the soybean field and classify the image of soybean field soil using the K-Nearest Neighbor algorithm. The results of soil classification are used to control the watering node for watering plants. This robot uses the Internet of Things concept with the MQTT protocol integrated with ThingsBoard as a display of monitoring information. The robot is built based on the Raspberry Pi 3 Model B+. In this research, with the KNN algorithm, the robot can classify soil moisture accurately and adequately, where it obtained 83.3% accuracy, 90% recall, 81.8% precision, and 85.7% F1 score. The watering node also performed well with a 94.4% success rate. In addition, soybeans in a field with the robot have better growth than soybeans in a field without robot. That is evidenced by the average plant height and the number of leaves in the field with the robot is better than those in the field without robot, that is 17.28 cm and 9 leaves compared to 15.72 cm and 8 leaves. However, plants without robot have a better stem diameter than those in a field with the robot, which is 2.8 mm compared to 2.74 mm.**

Keywords—Internet of Things, K-Nearest Neighbor, MQTT, Robot, Soybean

I. INTRODUCTION

Agriculture is a strategic sector in driving the national economy, namely in realizing food security, increasing competitiveness, expanding employment, and reducing poverty. The agricultural sector recognizes the term "strategic commodity," one of which is soybean commodity. The problem that occurs in soybean commodities is the production rate which has declined every year until 2019. Soybean production only reached 424 thousand tons, or the lowest in 5 years [1]. There are many factors behind low soybean production, but in general natural factors play a significant role in soybean growth and production. The condition of soil moisture, air temperature and humidity affect the growth of soybeans and soybean production [2].

The problem is often farmers plant soybeans by spreading seeds and letting them grow without adequate maintenance, including without checking the soil moisture. This is driven by the fact that many soybean farmers apply the intercropping system, namely planting soybeans and other types of crops (generally corn) simultaneously in the same field, so farmers have to take care of two types of crops at once. In addition, soybeans are just a side crop so farmers are less concerned about caring for soybeans [2].

Many researchers have researched in the field of technology-based agricultural system automation, or commonly referred to as smart farming. Arista Setyawan et al (2018) built a monitoring system for soil moisture, air temperature and humidity, which is integrated with the Internet of Things in the Message Queuing Telemetry Transport (MQTT) protocol which is used to transmit data and information from monitoring results to the ThingsBoard web server [3]. Ipin Prasojo et al (2020) built an automatic watering system based on the level of soil moisture [4]. Then, some studies used wheeled robot media to perform automatic watering based on the level of soil moisture. First, Rizal Isnanto et al (2020) implemented the concept of wall-follower robot and the ESP-NOW protocol to their watering robot [5]. Pengfei Lv et al (2020) built an intelligent watering robot with the NRF24L01 module as a communication communication module [6], and L. Mechsy et al (2017) built a watering robot for lawn maintenance using CPP (Coverage Path Planning) algorithm as robot navigation system [7]. All of those robots used the soil moisture sensor to measure soil moisture. In addition, Djulil Amri (2012) also built agricultural robot but worked to plant peanut seeds by utilizing the concept of image processing [8]. Almost the same as the previous one, Marcin Jasiński et al (2018) built an autonomous agricultural robot with a vision system utilizing image processing for plant/weed classification [9].

It does not stop with the Internet of Things. Smart farming today works more accurately and smarter with machine learning. In relation to machine learning, Zorgani and Ugail (2018) compared the performance of several machine learning algorithms in classifying histological images [10]. The research shows that the SVM (Support Vector Machine) and KNN (K-Nearest Neighbor) algorithms are the algorithms with the best accuracy, namely 99.86%, better than the Naïve Bayes, Binary Decision Tree, and Discriminant Analysis algorithms. Besides, KNN has advantages over other algorithms, namely a simple algorithm, fast training, and robust to noisy training data [11].

978-1-6654-2403-5/21/USB ©2021 IEEE

Of the many studies above, none of them have made soybeans the object of their research. Therefore, this research seeks to provide solutions to problem of neglecting soybean plant care by farmers, in the form of a robot that monitors the conditions of soybean field and classifies images of soybean field soil using the K-Nearest Neighbor algorithm. The results of soil classification are used to control the watering node for watering plants. This robot uses the Internet of Things concept based on the MQTT protocol. MQTT has a smaller payload size [12], lower power consumption, and higher success rate than HTTP [13]. Robot is built based on the Raspberry Pi 3 Model B+. The MQTT protocol is integrated with ThingsBoard as a display of monitoring information. This research is expected to help soybean farmers in caring for soybean fields and increasing soybean production.

II. Methodology

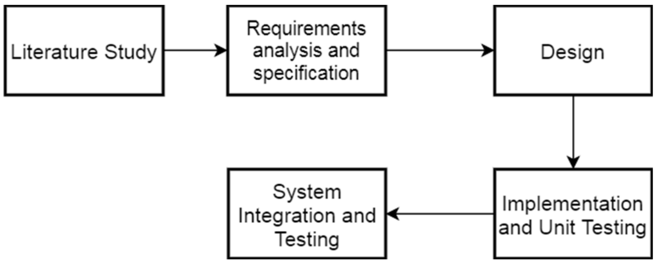

Fig. 1. Flowchart of Methodology

Literature study is the stage of extracting concepts and materials related to the problems raised and the design of the system that is built, both from devices, sensors, and actuators, communication protocols, and methods that can help in realizing the system.

The requirements analysis and specification stage is the stage to describe the needs needed in conducting research. Then these needs are analyzed and used at the design phase.

The design stage is the stage of designing hardware and software or programs needed in conceptual system development. The implementation and unit testing stage aims to implement the system design that was made in the previous stage and test each component used to ensure that the components can work properly.

In the integration and system testing stage, each component that has been tested is connected to form a complete system. Then, a full system test is carried out, as well as re-evaluating errors that can occur when a component is run as a system.

III. Design and Implementation

The working principle of the system is that the robot explores the soybean field while checking temperature and humidity and the processing and classifying soil images. If the soil image is classified as dry, the water pump will water the plants. On the other hand, if the soil image is classified as wet, then the water pump still off. Checking the condition of the land is always followed by sending data to the MQTT broker. When the entire land has been explored, the robot will stop.

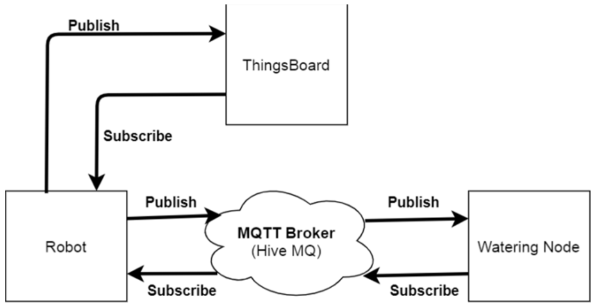

Fig. 2. Block Diagram of The Whole System

A. Hardware Design

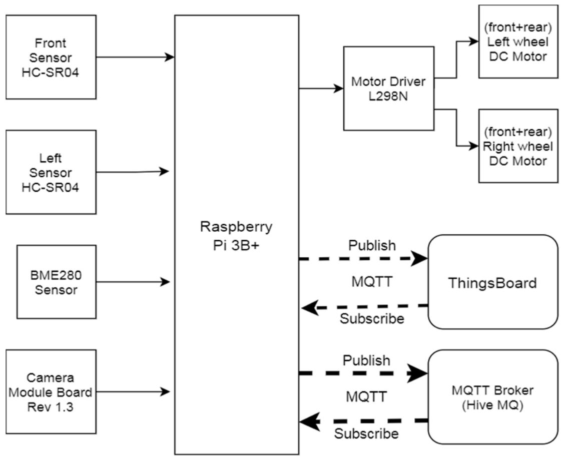

Fig. 3. Block Diagram of The Robot Hardware

Figure 3 shows a block diagram of the robot hardware. The robot is built based on the Raspberry Pi 3 Model B+, which is equipped with sensor and actuator components including 2 (two) HC-SR04 distance sensors, BME280 temperature, and humidity sensor, camera module board Rev 1.3, DC motor driver L298N which controls 4 (four) DC motors as a robot wheel.

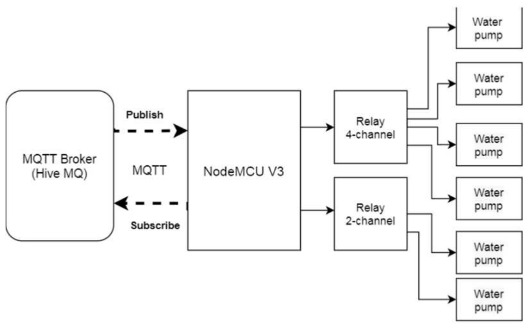

Fig. 4. Block Diagram of The Watering Node Hardware

Figure 4 shows a block diagram of the watering node hardware. The watering node is built based on NodeMCU V3, which is embedded with the ESP8266-12E wireless communication module and is equipped with a logic level converter, a 5V 4-Channel relay module a 5V 2-Channel relay that controls 6 (six) 12V micro water pumps.

B. Software Design: Robot Intelligence

The flowchart of robot intelligence software shown in Figure5. First of all, the used libraries are imported. Next, the program performs GPIO initialization and BME280 sensor initialization. Then, there are defining and allocating GPIO pins to each component and defining global variables. In addition, the MQTT protocol was initialized, which included creating an MQTT client and connecting MQTT to a broker.

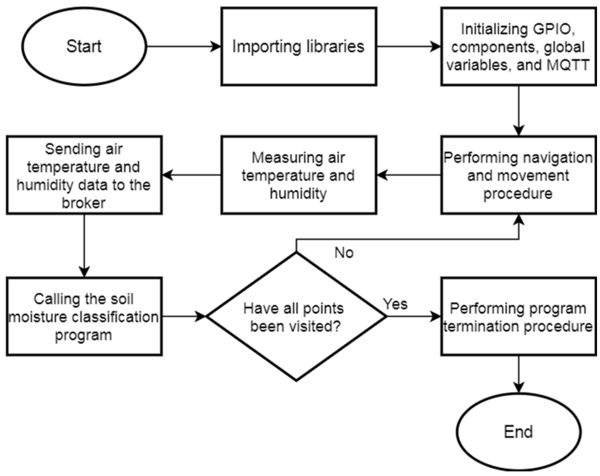

Fig. 5. Flowchart of Robot Intelligence Software

After that, a navigation and movement procedure determines the motion and direction of motion carried out by the robot based on the distance of the wall against the sensor (front and left) so that the robot moves to the desired point. After the robot is at the desired point, the robot will measure the air temperature and humidity values and then send them to the ThingsBoard broker. Still, at the same point, the robot will call the soil moisture classification program, which is tasked with classifying soil moisture at that point. After that, the robot will check whether all points have been visited or not. If the robot has checked the air and soil conditions 6 times and has met the corner (there is a wall in front and on the left side) 3 times, it means that the robot has been in the robot's cage and all points have been visited. If all points have been visited, the robot will stop the program, and if not, then the robot will return to carrying out the navigation and movement procedure.

C. Software Design: Soil Moisture Classification

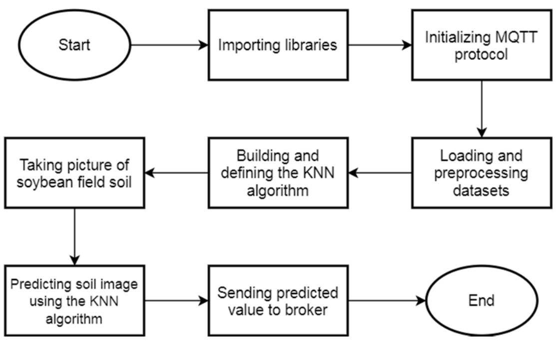

Fig. 6. Flowchart of Soil Moisture Classification Software

In the soil moisture classification program, we first imported the used libraries and initialized the MQTT protocol. Then, the dataset that has been in the CSV (Comma-separated values) file is loaded. This CSV file contains numbers ranging from 0 to 1, representing each pixel in each soil image. This CSV will be converted into a NumPy array which will be used in the classification process. Converting an image dataset into a CSV file is carried out outside of this program, where the process is described in Figure 7.

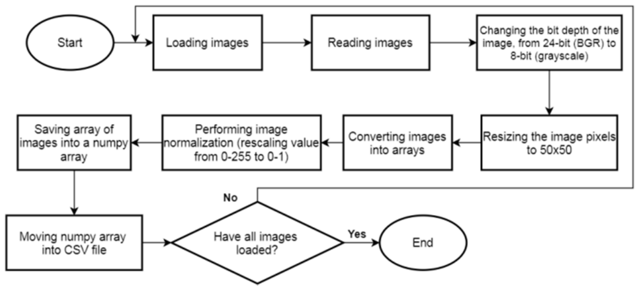

Fig. 7. Flowchart of Processing Image Dataset Into CSV File

Then, a KNN algorithm class is built, which contains methods for loading training data and test data, calculating the Euclidean distance between training data and test data, and predicting or classifying test data. Next, the program will capture the soil image where the robot is located and predict that soil image. Before making the prediction, the soil image is first processed into an array, which is the same process as converting an image dataset into a CSV file. Then, the prediction results are sent to the broker.

IV. Results and Discussion

A. KNN Algorithm Performance Test

Tests were carried out with a soil images dataset consisting of 143 images divided into 2 classes, namely "Wet" (63 pictures) and "Dry" (80 pictures). Soil conditions are stated as wet when the soil moisture is as desired, above or equal to 70%. Meanwhile, the soil is declared dry if the soil moisture is below 70% [2].

TABLE I. KNN Algorithm Performance on Training Data

Training-Testing Data Ratio	Training Set	
	Accuracy	**F1 score**
70 : 30	98.00%	98.15%
80 : 20	98.24%	98.36%
85 : 15	98.35%	98.48%

The dataset is split into 85% for training data and 15% for testing data in this test. That ratio is chosen because it produces the best accuracy and F1 score on training data compared to the others. Then, perform the calculation of accuracy, precision, recall, and F1 score for each k value. Tests are carried out in the range k = 1 to 20.

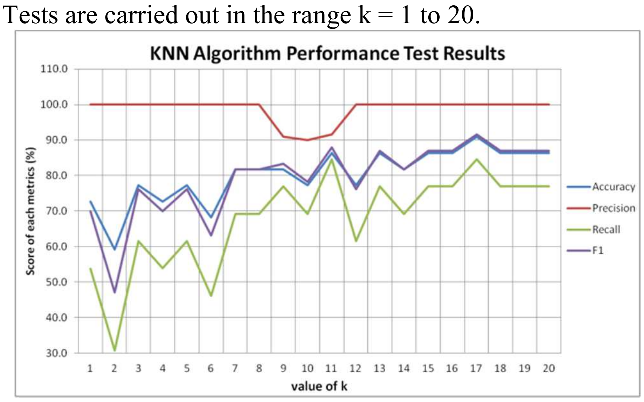

Fig. 8. Graph of KNN Algorithm Performance Test Results

The best k is 17 with 90.9% accuracy, 100% precision, 84.6% recall, and 91.7% F1 score.

B. Throughput Test

Throughput is the rate of data transmission [14]. The throughput can be formulated as follows:

$$Throughput = \frac{\sum packets\ received\ (in\ bits)}{total\ transmission\ time} \quad (1)$$

Throughput test is performed by sending a specific number of packets to the destination and taking note of the length of transmission time. The destination devices are ThingsBoard with the ThingsBoard server broker and watering node with Hive MQ broker. This test is carried out by varying the number of packets, which is 25 bytes/packet.

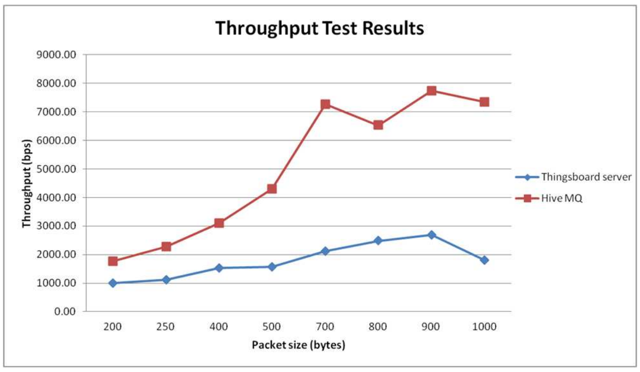

Fig. 9. Graph of Throughput Test Results

The ThingsBoard server broker has an average throughput of 1784.45 bps while the Hive MQ is 5040.45 bps.

C. Packet Loss Test

Packet loss (loss rate) is the percentage of packets dropped compared to the number of packets sent by the sender [14]. Packet loss can be formulated as follows:

$$Packet\ loss = \frac{\sum packets\ drop}{\sum\ packets\ sent} \times 100\% \quad (2)$$

Packet loss testing is performed by sending a specific number of packets to the destination with variations in packets and the gap between transmission. The destination devices are ThingsBoard with the ThingsBoard server broker and watering node with Hive MQ broker. The gap between transmission for the ThingsBoard server broker is 0.2 and 0.3 seconds, while the gap between transmission for the Hive MQ broker is 0.2 seconds. In this test, 1 packet is 15 bytes.

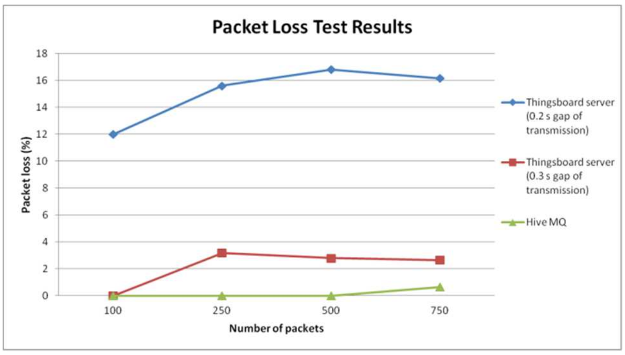

Fig. 10. Graph of Packet Loss Test Results

The ThingsBoard server broker has an average packet loss of 15.1% for 0.2 seconds gap of transmission and 2.2% for 0.3 second gap of transmission while the Hive MQ is 0.2%.

D. Delay Test

Delay is a latency that arises in the process of sending a packet [15]. Delay is the time it takes for a packet sent by the sender to arrive at the destination device [14]. Delay can formulated as follows:

$$Delay = \frac{total\ transmission\ time}{\sum packets\ received} \quad (3)$$

A delay test is performed by sending a specific number of packets to the destination and taking note of the length of transmission time. The destination devices are ThingsBoard with the ThingsBoard server broker and watering node with Hive MQ broker. In this test, on the ThingsBoard server broker, 1 packet is 15 bytes, while on Hive MQ broker, 1 packet is 25 bytes.

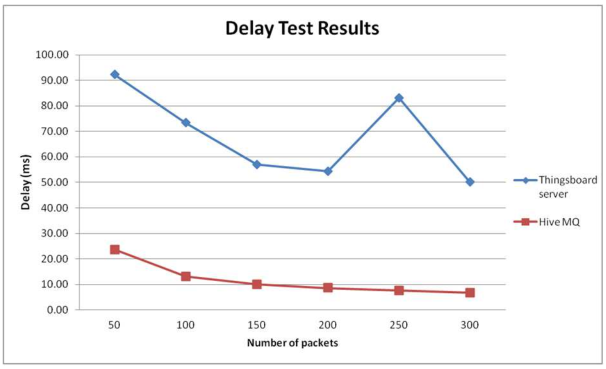

Fig. 11. Graph of Delay Test Results

The ThingsBoard server broker has an average delay of 68.31 ms while the Hive MQ is 11.59 ms.

E. The Whole System Test

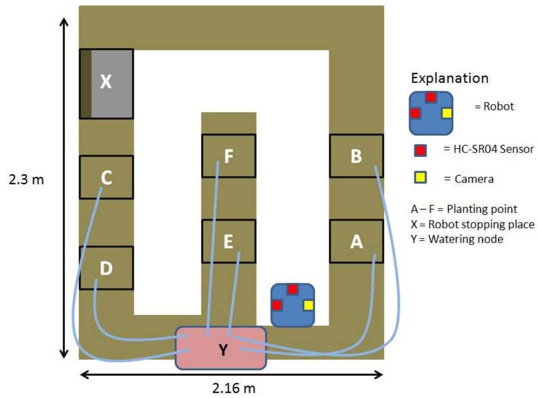

Fig. 12. Illustration of Soybean Field Prototype

This test examines the overall work of the system, which consists of a robot and a watering node when it is run in the soybean field. The soybean field prototype has a size of 4.96 m^2 which is illustrated in Figure 12. The test is carried out 3 times where each test consists of 6 checking points, resulting in 18 points or test result data.

Out of 18 test data, 3 times the error in predicting soil moisture conditions occurs. Then, of the 18 test points, there is 1 point that has not reached the desired soil moisture condition after watering. From the result of this test, a confusion matrix can be generated, which is shown in Table II.

TABLE II. CONFUSION MATRIX OF ROBOT TEST RESULT

	Actual Value: Dry	Actual Value: Wet
Predicted Value: Dry	9 (TP)	2 (FP)
Predicted Value: Wet	1 (FN)	6 (TN)

From the confusion matrix in Table II, we can calculate accuracy, recall, precision, and F1 score. Accuracy is the proportion of correct predictions divided by the number of predictions [17]. Accuracy is formulated as

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{4}$$

The recall is the percentage of positive cases that are correctly predicted. Precision is the ratio of the correct positive predictions among the positive predictions. Meanwhile, the F1 score is the harmonic mean between precision and recall [18]. Recall, precision, and F1 score are formulated as

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$F1\ score = 2 \times \frac{Precision \ \times Recall}{Precision + Recall} \tag{7}$$

where:
TP = true positive
TN = true negative
FP = false positive
FN = false negative

From equations (4), (5), (6), and (7), accuracy, recall, precision, and F1 score are obtained as follows:

$$Accuracy = \frac{9 + 6}{9 + 6 + 1 + 2} = 0.833 = 83.3\%$$

$$Recall = \frac{TP}{TP + FN} = \frac{9}{9 + 1} = \frac{9}{10} = 0.9 = 90\%$$

$$Precision = \frac{TP}{TP + FP} = \frac{9}{9 + 2} = \frac{9}{11} = 0.818 = 81.8\%$$

$$F1\ score = 2 \times \frac{0.818 \times 0.9}{0.818 + 0.9} = 0.857 = 85.7\%$$

So, from this test, the robot has 83.3% accuracy, 90% recall, 81.8% precision, and 85.7% F1 score. In addition, to calculate the success rate of the watering unit, the following equation can be used:

$$Success\ Rate = \frac{\sum successful\ trial}{\sum trial} \tag{8}$$

$$Success\ Rate = \frac{17}{18} = 94.4\%$$

So, in this test, the success rate of the watering node was 94.4%. In this test, the time needed by the robot to check at each planting point is 20-40 seconds.

F. Comparison of the Growth of Soybean in Field with the Robot and Field without Robot

In this test, performed a comparison of the growth of soybean in the field treated using robot and soybean in a field without robot. Comparisons were only carried out on 5 plants in each field. Observations were made up to 18 days after sowing (DAS). The parameters observed for plant growth were plant height, number of leaves, and stem diameter.

TABLE III. DEPENDENT VARIABLE

Variable	Value
Type of soil	Entisol
Type of seed	Willis F1 Varieties
Number of seeds per point	5 – 6 seeds
Hole depth	5 – 7 cm
Watering interval	2 days
Length of observation	18 days after sowing

TABLE IV. INDEPENDENT VARIABLE

Variable	Field with Robot	Field without Robot
Number of points	6	3
Soil moisture measurement	Performed by robot	Not performed
Water Volume	±210 ml	200 – 300 ml
Watering	Performed by watering node	Performed by human
Watering treatment	The robot checks every 2 days and is watered or not determined by the robot	Watering every 2 days

TABLE V. COMPARISON OF SOYBEANS HEIGHT

Plants	Height Plants (cm)					
	6 DAS		12 DAS		18 DAS	
	With Robot	Without Robot	With Robot	Without Robot	With Robot	Without Robot
1	6	5	13	12.4	15	14.8
2	7.5	6	13.5	13.3	16.4	15.6
3	7.5	5	13.5	12.6	16	14
4	9	6	14.9	13	18.2	16.2
5	8	8	16.4	15	20.8	18
Average	7.6	6	14.26	13.26	17.28	15.72

TABLE VI. COMPARISON OF SOYBEANS NUMBER OF LEAVES

Plants	Number of Leaves					
	6 DAS		12 DAS		18 DAS	
	With Robot	Without Robot	With Robot	Without Robot	With Robot	Without Robot
1	3	4	7	7	8	8
2	3	4	7	7	8	8
3	4	4	8	7	8	8
4	4	4	9	7	10	8
5	4	4	10	7	11	8
Average	3.6	4	8.2	7	9	8

TABLE VII. COMPARISON OF SOYBEANS STEM DIAMETER

Plants	Stem Diameter (mm)					
	6 DAS		12 DAS		18 DAS	
	With Robot	Without Robot	With Robot	Without Robot	With Robot	Without Robot
1	1.27	1.59	2.23	2.55	2.86	2.86
2	1.59	1.27	2.23	1.91	2.86	2.86
3	1.59	1.27	2.55	2.23	2.86	2.86
4	1.59	1.59	2.23	2.23	2.55	2.86
5	1.59	1.59	1.91	2.23	2.55	2.55
Average	1.53	1.46	2.23	2.23	2.74	2.80

After 18 days after sowing, the plants in the field with the robot had a better average plant height and number of leaves, namely 17.28 cm and 9 leaves, compared to plants in the field without robot, which are 15.72 cm and 8 leaves. Meanwhile, plants in the field without robot have a better stem diameter than those in the field with robot, which is 2.8 mm compared to 2.74 mm.

V. CONCLUSION AND RECOMMENDATION

A. Conclusion

In this research, with the help of an autonomous robot and the soil moisture classification method using the KNN algorithm, farmers were able to increase soybean growth. This is evidenced by the average plant height and the number of leaves in the field with the robot is better than those in the field without robot, that is 17.28 cm and 9 leaves compared to 15.72 cm and 8 leaves. It can be achieved because the robot can classify soil moisture accurately and adequately, where it obtained 83.3% accuracy, 90% recall, 81.8% precision, and 85.7% F1 score. The watering node also performed well with a 94.4% success rate. Whereas the KNN algorithm achieves optimal performance when k = 17, where using a dataset of 143 images and the dataset is split into 85% training data and 15% test data, and it is obtained 90.9% accuracy, 100% precision, 84.6% recall, and 91.7% F1 score. Even so, plants in the field without robot have a better stem diameter than those in the field with the robot, which is 2.8 mm compared to 2.74 mm.

B. Recommendation

This research still uses the relatively simple KNN algorithm, so further research is recommended to use an algorithm that is able to produce better accuracy but is still practical and lightweight. This research also only has 2 soil classifications, namely "Dry" and "Wet", besides that the dataset used is still relatively small. So, the number of classes can be augmented and detailed in the future, and the number of datasets can be enlarged. In addition, in this research the robot takes about 20-40 seconds to check each point. So, in the future, the waiting time can be minimized.

ACKNOWLEDGMENT

This research was financially supported by The Faculty of Engineering, Diponegoro University, Indonesia through Strategic Research Grant 2021 number: 3178/S/komputer/4 /UN7.5.3.2/PP/2021.

REFERENCES

[1] Planning Bureau, Secretariat General, Ministry of Agriculture RI, "Strategic Plan of the Indonesian Ministry of Agliculture 2020 - 2024," Ministry of Agriculture Republic Indonesia, Jakarta, 2020.

[2] T. Adisarwanto, Subandi and Sudaryono, "Soybean Production Technology," Indonesian Legumes and Tuber Crops Research Institute, Malang, 2016.

[3] A. B. Setyawan, M. H. lchsan and G. Setyawan, "Monitoring System for Soil Moisture, Air Humidity, and Temperature on Agricultural Land Using the MQTT Protocol," *Jurnal Pengembangan Teknologi Informasį dan Ilmu Komputer,* vol. 2, no. 12, pp. 7502-7508, 2018.

[4] I. Prasojo, A. Maseleno, O. Tanane and N. Shahu, "Design of Automatic Watering System Based on Arduino," *Journal of Robotics and Control,* vol. 1, no. 2, pp. 55-58, 2020.

[5] R. R. Isnanto, Y. E. Windarto, J. I. D. Gloriawan and F. N. Cesara, "Design of a Robot to Control Agricultural Soil Conditions using ESP-NOW Protocol," *2020 Fifth International Conference on Informatics and Computing (ICIC),* pp. 1-6, 2020.

[6] P. Lv, S. Wang, J. Li, X. Gao, H. Liu, J. Lv, Y. Shi, P. Zhang, D. Luo, H. Che, J. Niu and J. Wang, "Design of Intelligent Watering Robot," *2020 IEEE International Conference on Mechatronics and Automation (ICMA),* pp. 1499-1504, 2020.

[7] L. S. R. Mechsy, M. U. B. Dias, W. Pragithmukar and A. L. Kulasekera, "A mobile robot based watering system for smart lawn maintenance," *2017 17th International Conference on Control, Automation and Systems (ICCAS),* pp. 1537-1542, 2017.

[8] D. Amri, "Design of Agricultural Robot Based on ATMega128 Microcontroller," *National Seminar "Science, Engineering and Technology",* pp. 1-6, 2012.

[9] M. Jasiński, J. Mączak, P. Szulim and S. Radkowski, "Autonomous Agricultural Robot – Testing of the Vision System for Plants/Weed Classification," in *Automation 2018*, Warsaw, 2018.

[10] M. A. Zorgani and H. Ugail, "Comparative Study of Image Classification using Machine Learning Algorithms," *The 2nd Annual Innovative Engineering Research Conference,* no. 332, 2018.

[11] N. Bathia and Vandana, "Survey of Nearest Neighbor Techniques," *International Journal of Computer Science and Information Security,* vol. 8, no. 2, pp. 302-305, 2010.

[12] T. Yokotani and Y. Sasaki, "Comparison with HTTP and MQTT on Required Network Resources for IoT," *The 2016 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC),* pp. 1-6, 2016.

[13] S. Nicholas, "Power Profiling: HTTPS Long Polling vs. MQTT with SSL, on Android," Stephendnicholas.com, 31 May 2012. [Online]. Available: http://stephendnicholas.com/posts/power-profiling-mqtt-vs-https. [Accessed 15 October 2020].

[14] R. Banno, J. Sun, M. Fujita, S. Takeuchi and K. Shudo, "Dissemination of Edge-Heavy Data on Heterogeneous MQTT Brokers," *2017 IEEE 6th International Conference on Cloud Networking (CloudNet),* pp. 1-7, 2017.

[15] V. M and S. Sankaranarayanan, "Publish/subscribe based multi-tier edge computational model in Internet of Things for latency reduction," *Journal of Parallel and Distributed Computing,* vol. 127, pp. 18-27, 2019.

[16] A. Zheng, "Evaluation Metrics," in *Evaluating Machine Learning Models*, Sebastopol, O'Reilly Media, Inc, 2015, p. 8.

[17] S. Shalev-Shwartz and S. Ben-David, "Multiclass, Ranking, and Complex Prediction Problems," in *Understanding Machine Learning: From Theory to Algorithms*, New York, Cambridge University Press, 2014, pp. 244-245.

[18] J. Bartnitsky, "HTTP vs MQTT performance tests," flespi, 23 January 2018. [Online]. Available: https://flespi.com/blog/http-vs-mqtt-performance-tests. [Accessed 15 October 2020].