IoT Based Smart Plant Irrigation System with Enhanced learning

Kemal Cagri Serdaroglu, Cem Onel, Sebnem Baydere
Yeditepe University
Istanbul, TR
{kserdaroglu,sbaydere}@cse.yeditepe.edu.tr

***Abstract*—In this study, we propose a smart plant irrigation IoT system that autonomously adapts itself to a defined irrigation habit. The automated plant irrigation systems generally make decisions based on static models derived from the plant's characteristics. In contrast, in our proposed solution, irrigation decisions are dynamically adjusted based on the changing environmental conditions. The learning mechanism of the model reveals the mathematical connections of the environmental variables used in the determination of the irrigation habit and progressively enhances its learning procedure as the irrigation data accumulates in the model. We evaluated the success of our irrigation model with four different supervised machine learning algorithms and adapted the Gradient Boosting Regression Trees(GBRT) method in our IoT solution. We established a test bed for the sensor edge, mobile client, and the decision service on the cloud to analyze the overall system performance. The early results from our prototype system that is tested with two indoor plants; namely Sardinia and Peace-lily are very encouraging. The results reveal that the proposed system can learn the irrigation habits of different plants successfully.**

***Index Terms*—IoT, machine learning, smart plant irrigation, GBRT, testbed.**

I. INTRODUCTION

In today's society, automated irrigation systems have become a mandatory need in many places and continue to spread rapidly [1]. From personal plant growing areas to large-scale farms, people no longer have the time and material for manually watering their plants as required [1]. Reducing the cost and increasing the efficiency of irrigation can be possible with remotely controlled automated irrigation systems having capability of making smart decisions [2]. Internet of things (IoT) makes remote and smart irrigation possible as it organizes agricultural things for the creation of necessary observation media [3]. Hence, IoT systems become an essential part of precision agriculture like in other parts of our society [4] [5].

The general approach used for a human-controlled conventional irrigation system consists of two main processes [4]. Firstly, agronomists create a weekly or monthly irrigation plan by evaluating the data that they get from the plant's environment. Secondly, they transfer this plan into the system via a user interface, and every week or month, they follow this procedure for every single plant they have in the environment. The main difference between its smart and autonomous counterpart is that the smart system can self-learn the environmental conditions which affect the irrigation. Besides, it can define and schedule an irrigation plan by itself without any human intervention, resulting in increased efficiency [6] [7].

In this study, we develop a smart irrigation model that mimics the unstructured decision process of the human-controlled irrigation, done by agronomists. This model autonomously adapts itself to a defined irrigation habit with an enhanced learning method by considering the plant's characteristics. It uses a machine learning approach for revealing the mathematical connections of the environmental variables needed in the determination of the irrigation habit [8] and enhances its learning procedure as the amount of irrigation data accumulates in the system. In the development stage of our approach, we have evaluated four basic supervised machine learning algorithms namely, KNN, LR, GBRT and GNB on the collected irrigation data. Since it is suitable for additive enhanced learning methods and give better results in the regression as it iteratively improves the decision model, we adapted GBRT algorithm to our learning model. Besides, we have developed the components of an IoT based irrigation system which uses the proposed learning model for the data analytics. Overall, the system first gathers data via sensors used to measure environmental conditions such as soil moisture, air temperature, and humidity sensor. Then it performs decision-making process with the machine learning algorithm based on the collected data and makes a mathematical correlation of different plant environment variables. As a result, the system is actuated on the decision for irrigation.

The organization of the paper is as follows: In Section II, we examine the state of art solutions related to our study. In Section III, we introduce our irrigation model and its steps. In Section IV, we describe the components of our smart irrigation system. Section V presents the test results. In Section VI, we conclude our study and elaborate on the future work.

II. RELATED WORK

Meher et al. [9] studied on plant irrigation and water logging system for edge-cloud based IoT. This system transmits the sensor data to the cloud for decision making. Rather than using an automated irrigation method, this system requires a client reaction for the start of irrigation. Additionally, it does not contain an automated decision mechanism using a learning approach. Though, this system is useful for the control of water shortages in the land.

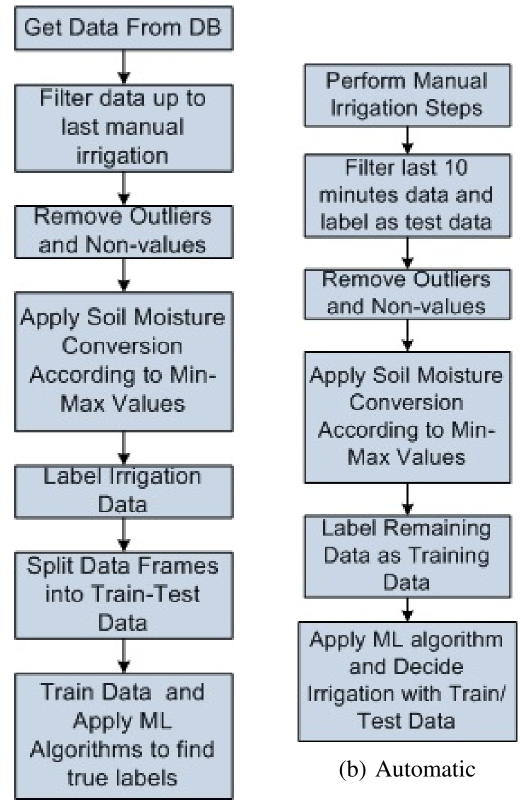
Goldstein et al. [4] and Vij et al. [10] developed an irrigation decision support model that applies ML methods with a readily prepared irrigation data set. They applied different ML algorithms and evaluated their precision for irrigation decisions. Rather than the need for a ready data set, in our study, we aim to develop such a model that progressively learns the irrigation needs of any plants. Our solution needs initial manual irrigations only a couple of times before making accurate decisions. Also, since our model is fully dynamic and based on the processing of data progressively collected from the environment, it can be adapted to different plants with different irrigation conditions easily.

Kwok et al. [1] present an automated plant irrigation system that uses deep learning for the detection of the type of plant. The system determines the water need of the plant with a deep-learning-based plant type recognition process using a prepared plant image data set and images obtained from the farm. After the recognition of the plant, it uses the irrigation information of the corresponding plant, stored in a database. The model training process might take a long time because it needs storage of a large number of images in the system. Different from this method, our learning model does not require plant type recognition, which uses images. So, the training stage does not take a long time interval, since our learning model trains itself by a relatively lighter learning process that uses environmental parameters which do not require larger storage in the system.

III. Proposed Irrigation Model

The proposed irrigation model has two alternative mechanisms for learning the watering nature of the plant. In the first method, the model learns the irrigation needs without any pre-prepared data provided to it. In the learning process of this option, initially, it is required to train the system with manual irrigations a couple of times according to recommendations pre-defined by the agronomists. The system collects the necessary data and trains itself, and after an adequate number of manual irrigations, the proposed model begins to grasp the watering needs of the plant. After that, it can make an autonomous watering decision with the collected data using machine learning methods and does not require any further manual irrigation. In the second method, the model can decide on irrigation automatically using machine learning methods without any pre-prepared data provided for its operation. In each option, as the number of accurate irrigation points provided to the model increase, the proposed model enhances its learning process progressively.

We can divide the decision-making procedure of these options in two main steps. The first one is the model training step with manual irrigation, named as Step M. The second one is the automatic irrigation decision making step, named as Step A. The flow of these steps are illustrated in Fig. 1. The proposed model has three stages in performing these steps;data collection, data preparation and model application.



(a) Manual

(b) Automatic

Fig. 1: Model Flow for Irrigation Decision

A. Data Collection

In the data collection stage, which is performed for both options, the system periodically acquires necessary air temperature, humidity, and soil moisture data needed for the watering decision from the physical environment. After the acquisition of these data, they are timestamped and collected into a database. Besides, in the data collection stage, the time when automatic or manual irrigation performed by the system is recorded on the database. So with this information, the decision-making system knows the time of irrigation.

B. Data Preparation

At the data preparation stage, firstly, for each learning options, anomalies in the irrigation data set are eliminated.

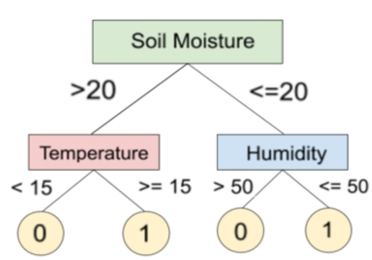


Fig. 2: Decision Tree.

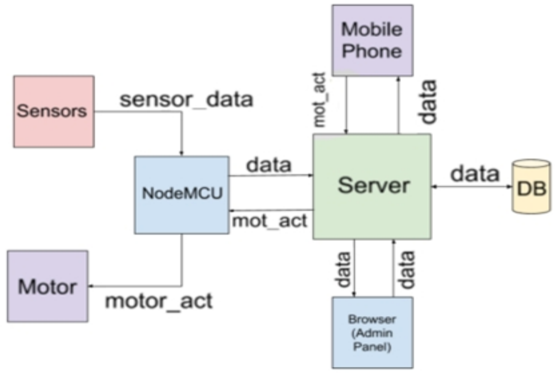


Fig. 3: Overall Architecture

This elimination process removes the broken data caused by the instant problems of the sensors or data transmission. Secondly, again for each options, soil humidity sensor values are converted to soil humidity percentages. In this step, the system determines the maximum and minimum soil moisture values in the data set. The system converts all raw values to a percentage value by considering the maximum raw sensor value maps to 0% and the minimum one maps to 100%. In the third and the most important step, the system labels the candidate irrigation points in the data set with the Algorithm 1. Our model performs these steps to data up to last manual irrigation for Step M. For Step A, these steps are performed to all data in the database. After that, all data are ready to be processed in ML methods.

Algorithm 1 Labeling Irrigation Data

$avg \leftarrow averageofmanualirrigations()$
$stddev \leftarrow standarddeviationofmanualirrigations()$
for each $dat \in dataset$ **do**
  $distance \leftarrow calculateeuclediandistance(dat, avg)$
  if $distance < stddev$ **then**
    labeldata(dat,irrigation)
  else
    labeldata(dat,non-irrigation)
  end if
end for

C. Model Application

For the learning and decision making stages, in each option of the model, we use supervised machine learning (ML) methods. Supervised learning methods need a decision tree for their training. So, we define the decision tree, depicted in Fig. 2, with considering the environmental parameters necessary for the irrigation conditions.

For Step M, we split the data up to last manual irrigation into train-test data with 80% and 20% for the training. Then we apply four different ML algorithms and assess their performances in terms of true irrigation labels in the data set. The details of the comparison of these algorithms are given in Section V.

For Step A, firstly, the system performs the steps needed for Step M. Then, it filters the last 10 minutes of data and labels them as test data. It labels all remaining data as training data. After that, it applies ML algorithms for the decision of an irrigation instance. As more data are collected from the environment and more automatic irrigation decisions are made, the system begins to create more accurate decision models and enhances its learning process.

IV. SMART PLANT IRRIGATION SYSTEM

The proposed irrigation learning model is implemented in an prototype IoT system. This system has three main parts. The first one is the sensors and actuator part, which acquires sensor data from the environment. Besides, it actuates the water pump motor when the system decides irrigation using the proposed model. In this part of the system, we use sensors, motor actuators, and a NodeMCU module. The second part performs the proposed irrigation decision model on a cloud server and outputs a decision result. This server collects the sensor data into a database and periodically makes an irrigation decision using the proposed model. Besides, it interacts with the client users of the system. The third part is the end user part in which a mobile device can send manual irrigation requests and visualize the monitored irrigation data. The overall interactions in the system are illustrated in Fig. 3.

A. Sensors and Actuator Part

We connect all sensors and motor actuators to an edge device, named as NodeMCU. This device acquires data from the environment, timestamps them, and sends them to the central server over its WI-FI module. Also, as depicted in the process scheme of the firmware in it in Fig. 4, it periodically checks the irrigation decisions made by the proposed model performed in the server. If any irrigation decision is made from the server, it actuates the related motor and irrigation process starts.

B. Server Part

As mentioned earlier, the necessary environmental data are collected into a centralized server for the decision making process and data visualization. This server is the core of this decision-making process, which follows the proposed smart irrigation model. It collects the timestamped data coming from NodeMCU into its database, and with using these data, in every 10 minutes period, it makes an irrigation decision. Besides, this server interacts with system users to get manual irrigation requests and provide the necessary data to them.

C. System User Part

Users can interact with the server part of the system with a web panel or mobile phones. With the web panel, the user can visualize the data, add/change any plant information such as irrigation type, irrigation time needed for the plant. With mobile phones, users can send a manual irrigation signal to

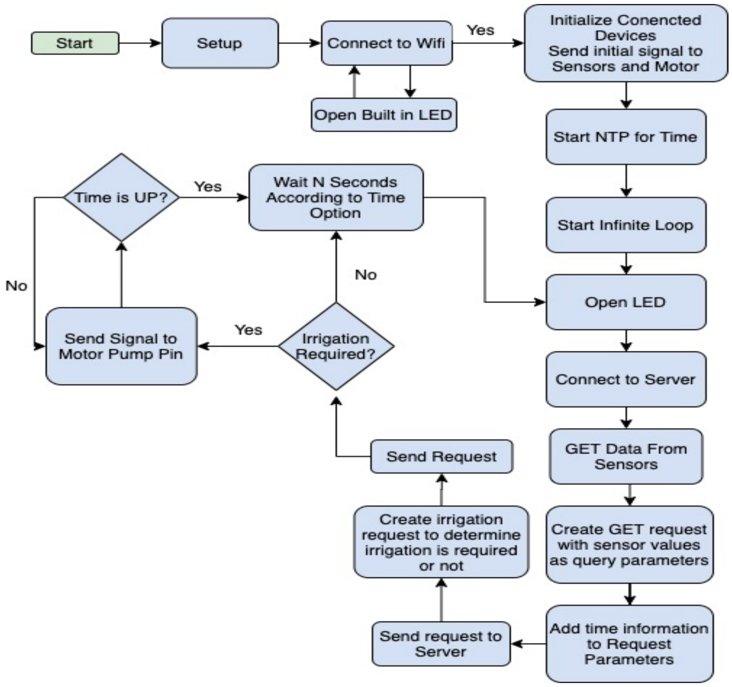


Fig. 4: NodeMCU Program Flow

TABLE I: MANUAL IRRIGATION TESTS RESULTS

Test Case	Label	Algorithm	Precision
Peace-Lilly 1 Manual Irrigation	I	GBRT	100%
Peace-Lilly 1 Manual Irrigation	N	GBRT	100%
Peace-Lilly 1 Manual Irrigation	I	GNB	100%
Peace-Lilly 1 Manual Irrigation	N	GNB	100%
Peace-Lilly 1 Manual Irrigation	I	LR	100%
Peace-Lilly 1 Manual Irrigation	N	LR	100%
Peace-Lilly 1 Manual Irrigation	I	KNN	100%
Peace-Lilly 1 Manual Irrigation	N	KNN	100%
Peace-Lilly 2 Manual Irrigation	I	GBRT	100%
Peace-Lilly 2 Manual Irrigation	N	GBRT	100%
Peace-Lilly 2 Manual Irrigation	I	GNB	100%
Peace-Lilly 2 Manual Irrigation	N	GNB	100%
Peace-Lilly 2 Manual Irrigation	I	LR	89%
Peace-Lilly 2 Manual Irrigation	N	LR	100%
Peace-Lilly 2 Manual Irrigation	I	KNN	95%
Peace-Lilly 2 Manual Irrigation	N	KNN	100%
Sardinia 3 Manual Irrigation	I	GBRT	100%
Sardinia 3 Manual Irrigation	N	GBRT	100%
Sardinia 3 Manual Irrigation	I	GNB	100%
Sardinia 3 Manual Irrigation	N	GNB	100%
Sardinia 3 Manual Irrigation	I	LR	100%
Sardinia 3 Manual Irrigation	N	LR	100%
Sardinia 3 Manual Irrigation	I	KNN	100%
Sardinia 3 Manual Irrigation	N	KNN	100%

the server and also visualize the environmental data in the database with selecting plants.

V. Tests and Results

The experimental setup that we built as a prototype is depicted with the system scheme in Fig. 3. We implement the Sensors and Actuator part of the system described in Section IV in an Arduino ESP8266 NodeMCU device. The sensors and the motor of the water pump are embedded in this device. This device communicates with the server part of the system in a home-wide WLAN from the IoT edge. The overall procedure, which is explained by the Server Part in Section IV is implemented in a MAC notebook. Sensor data acquired by the server part is stored in a Postgre SQL database. The system user part which interacts with the server is implemented in a Mobile Phone.

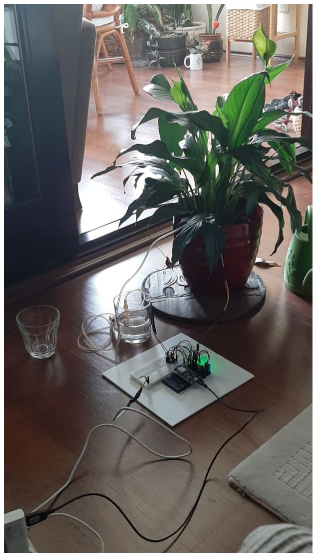
The motor which is embedded in the Arduino Device is submersible and, as depicted in Fig. 5b, a serum pipe is connected to this motor for drip irrigation, which allows water to drip slowly to the root of the plant. Moreover, Figure 5a illustrates the overall sensing/actuation edge of the system.

We devised tests with two different indoor plants that have different irrigation characteristics; peace-lilly and sardinia. Peace-Lilly needs irrigation within a week, whereas Sardinia needs 3-days period irrigation.

In the experiments, we collected soil moisture, indoor air temperature, and humidity data for a couple of weeks and record them into several data sets. We used these data sets to evaluate the performance of the proposed irrigation model when we apply both manual and automatic irrigation. As mentioned above, we examined the performance of four different ML algorithms; Gradient Boosting Regression Trees (GBRT), Logistic Regression (LR), Gaussian Naive Bayes (GNB) and K-Nearest Neighbours (KNN) algorithms [4], and adopted GBRT in our system.

A. Manual Irrigation Tests

We devised manual irrigation tests in three different cases and collected three data sets for each of them. Figures 6-7 illustrate the change of sensor values and show the manual ir-



(a) Sensing/Actuation Edge of Irrigation System.

(b) Water pump and Serum pipe.

Fig. 5: Sensing/Actuation Edge of the testbed and Irrigation Equipment for the experimental setup.

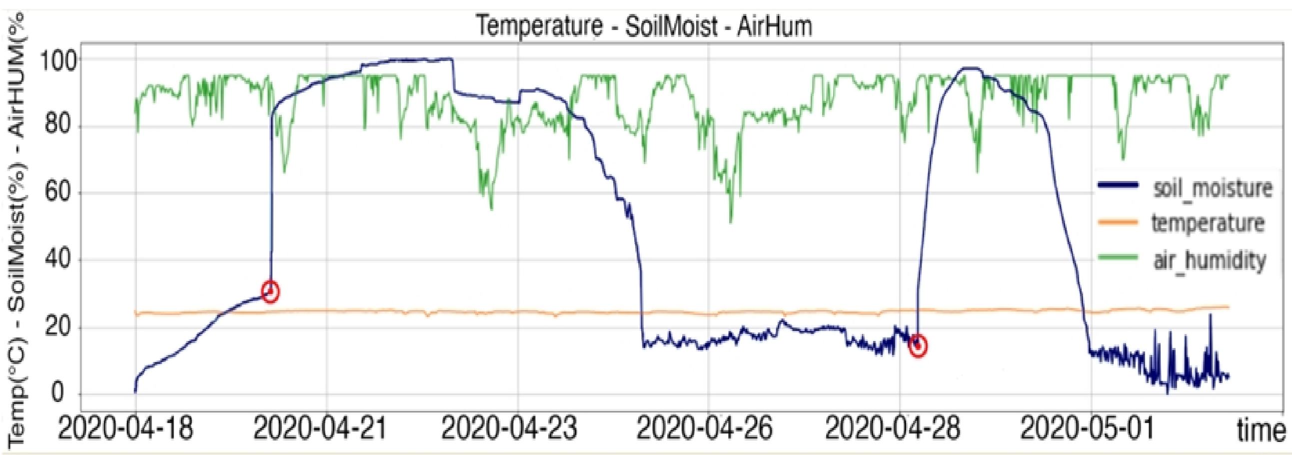


Fig. 6: Change of Sensor Values for Peace Lilly 3 Week Automatic Irrigation

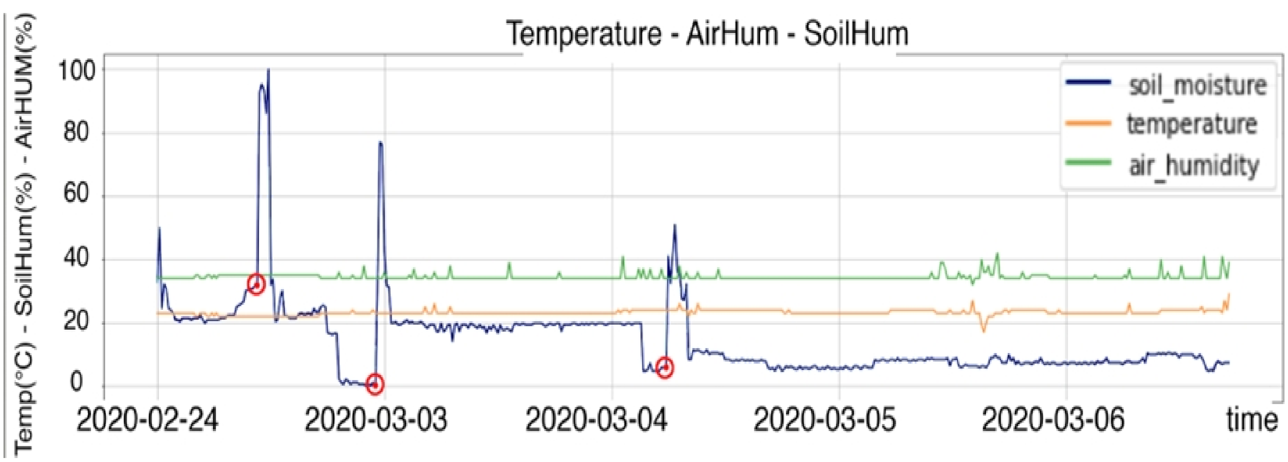


Fig. 7: Change of Sensor Values for Sardinia 10 Days 3 Manual Irrigation

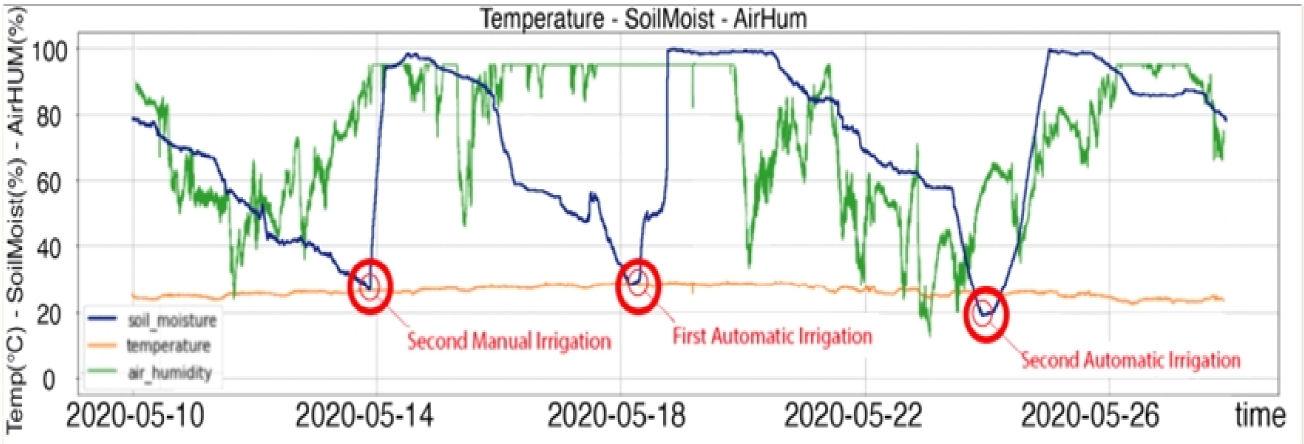


Fig. 8: Change of Sensor Values for Peace Lilly 3 Week Automatic Irrigation

rigation points with red dots and circles, for the corresponding data set. For each test case, the server of the system gets the data in these data sets from the database and provide them as input to the learning procedure of the proposed model. After that, it performs the manual irrigation point determination steps of the proposed model given in Fig. 1a. By performing these steps, the system determines the true irrigation points when it uses the aforementioned ML algorithms in its learning process. So, we compare the success of these algorithms, considering the precision in the determination of irrigation (I) and non-irrigation (N) labels. Table I gives these precision percentage results. With a few exceptions, algorithms perform with 100 % accuracy for finding the true manual irrigation labels when the proposed model assists their process.

B. Automatic Irrigation Tests

To observe the performance of the model for automatic irrigations, we performed a test case that spans two manual irrigations and two automatic irrigations. Fig. 8 depicts the variation of each sensor data and shows the corresponding manual and automatic irrigation points. Overall, this graph reveals that, after the second manual irrigation, the proposed model precisely finds the two automatic irrigation points as expected.

VI. Conclusions and Future Work

In this study, we propose a smart irrigation model based IoT system that gradually learns the watering nature of a plant without any pre-prepared data initially given to it. As a proof of concept, we implemented a prototype application. This application adapts itself to the conditions necessary for the irrigation after a couple of manual irrigations. To evaluate its performance, we devised tests both for manual and automatic irrigations when different ML algorithms are used. The results show that the model performs with high accuracy in making irrigation decisions.

For future work, we will investigate the performance of the proposed model with long-term irrigation tests with more plants. Our initial tests were conducted in an indoor environment where temperature changes are negligible. Additionally, we will extend our test bed to outdoor environments to observe the behaviour of the model in different environmental conditions.

VII. Acknowledgement

This work was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK), Grant No: 119N049.

References

[1] J.Kwok, S .Yu.“A smart iot-based irrigation system with automated plant recognition using deep learning”,*In Proc. 10th International Conference on Computer Modeling and Simulation*, 2018.

[2] S. Millan, C. Campillo, J. Casadesus, M. Monino, A. Vivas, M. Prieto, “Automated Irrigation Scheduling for drip-irrigated plump trees”, *Precision Agriculture ’19*, pp. 951-957, 2019.

[3] L. Atzori, A. Iera, G. Morabito, “The Internet of Things: A Survey”, *Computer Networks*, 2010, pp. 2787-2805

[4] A. Goldstein, L. Fink, A. Meitin, S. Bohadana, O. Lutenberg, G. Lavid, “Applying machine learning on sensor data for irrigation recommendations: revealing the agronomist’s tacit knowledge.”, Precision agriculture 19.3, (2018): 421-444.

[5] T. Adeyemi, I. Grove, S. Peets, T. J. Norton, “Advanced monitoring and management systems for improving sustainability in precision irrigation.”, Sustainability 9.3, (2017): 353.

[6] M.N. Umeh, N.M. Njideka, S.O. Okafor, F.C. Agba, “Intelligent Microcontroller-Based Irrigation System with Sensors”, *American Journal of Computer Science and Engineering*, vol. 2, pp. 1 - 4, 2015.

[7] K. Kansara, V. Zaveri, S. Shah, S. Delwadkar, K. Jani, “Sensor based Automated Irrigation System with IoT: A Technical Review”, *International Journal of Computer Science and Information Technologies*, vol. 6, no. 6, pp. 5331 - 5335, 2015.

[8] H.K. Pradeep, P. Jagadeesh, M. S. Sheshshayee, D. Sujeet, “Irrigation System Automation Using Finite State Machine Model and Machine Learning Techniques”, *In Proc. International Conference on Intelligent Computing and Communication, ICICC 2019*, 2019

[9] C. Meher, A. Sahoo, S. Sharma, “IoT based Irrigation and Water Logging monitoring system using Arduino and Cloud Computing”, *In Proc. 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking*, 2019

[10] A. Vij, S. Vijendra, A. Jain, S. Bajaj, A. Bassi, A. Sharma, “IoT and Machine Learning Approaches for Automation of Farm Irrigation System”, *Procedia Computer Science*, vol. 167, pp. 1250 - 1257, 2020.