Received: 09 September 2022 | Revised: 02 October 2022 | Accepted: 02 October 2022
DOI: 10.37917/ijeee.19.1.9 Vol. 19| Issue 1| June 2023

Open Access

Iraqi Journal for Electrical and Electronic Engineering
Original Article

Deep learning and IoT for Monitoring Tomato Plant

Marwa Abdulla*[1], Ali Marhoon[2]
[1] Computer Science, College of Education, University of Basrah, Basrah, Iraq
[2] Electrical Engineering Department, College of Engineering, University of Basrah, Basrah, Iraq

Correspondence
* Marwa Abdulla
Computer Science, College of Education,
University of Basrah, Basrah, Iraq
Email: cepsm510006@avicenna.uobasrah.edu.iq

Abstract
Agriculture is the primary food source for humans and livestock in the world and the primary source for the economy of many countries. The majority of the country's population and the world depend on agriculture. Still, at present, farmers are facing difficulty in dealing with the requirements of agriculture. Due to many reasons, including different and extreme weather conditions, the abundance of water quality, etc. This paper applied the Internet of Things and deep learning system to establish a smart farming system to monitor the environmental conditions that affect tomato plants using a mobile phone. Through deep learning networks, trained the dataset taken from PlantVillage and collected from google images to classify tomato diseases, and obtained a test accuracy of 97%, which led to the publication of the model to the mobile application for classification for its high accuracy. Using the IoT, a monitoring system and automatic irrigation were built that were controlled through the mobile remote to monitor the environmental conditions surrounding the plant, such as air temperature and humidity, soil moisture, water quality, and carbon dioxide gas percentage. The designed system has proven its efficiency when tested in terms of disease classification, remote irrigation, and monitoring of the environmental conditions surrounding the plant. And giving alerts when the values of the sensors exceed the minimum or higher values causing damage to the plant. The farmer can take the appropriate action at the right time to prevent any damage to the plant and thus obtain a high-quality product.

KEYWORDS: Deep learning, IoT, Smart farming system, mobile application, Remote Irrigation, plant monitoring, tomato disease.

I. Introduction

Agriculture considered the most critical development in human civilization, as people began farming thousands of years ago. In the ancient culture of Mesopotamia, agriculture was the country's main activity, and the reason for its prosperity was the abundance of fresh water. And agriculture continued to flourish like this. In 1979, Iraq was self-sufficient in many crops, including wheat [1]. But at the moment our country, Iraq, suffers from a decline in agricultural production of all kinds in general due to several reasons, including the little knowledge of the farmer to the plant diseases, accommodation on agricultural land, lack of seeds, fertilizers, water, modern agricultural machinery, etc. [2]. We can provide farmers with a system based on the Internet of Things (IoT) and Deep Learning (DL) to help them protect their farms. Introducing DL in the agricultural field is an essential modern research field. And this field includes sub-fields that still need research and development, such as classifying viral and fungal plant diseases, counting fruits of all kinds, predicting the date of harvesting fruits, knowing the type of plant, and others [3]. The applications of IoT in the agricultural field revolve around collecting environmental data that affect plants, including temperature, humidity, precipitation, wind speed, soil content, and others, by using sensors [4]. Thus, this data is used to automate farming techniques to make sound decisions, reduce waste, increase yields, and reduce the effort to manage crops. Therefore, the IoT in agriculture is a research topic that needs to be developed, especially in establishing systems to monitor climatic conditions that affect plant health, automating farming techniques, and reducing the challenges facing this process [5]. Several methods have been proposed for using sensors to measure and monitor environmental factors, in addition to using DL algorithms in predicting, classifying, and counting crops in agricultural fields [6], [7]. So IoT is involved in the agricultural field in many areas, including Monitoring the Climatic [8], [9], automatic irrigation of the soil to reduce water waste [10], Soil chemical properties [11]. Drones with Sensors and Cameras It is possible to draw maps, photograph, and survey agricultural lands [12], As for deep learning, it has entered into many fields, including classification of plant diseases, pest identification, fruit counting, etc. [13], [14]. So in this paper we focus on the tomato crop is one of the most important crops used in abundance locally and globally, and we have recently noticed the lack and poor quality of its local

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.
© 2023 The Authors. Published by Iraqi Journal for Electrical and Electronic Engineering by College of Engineering, University of Basrah.

production. So our objective in this paper was Create a mobile application that classifies tomato diseases based on a deep learning Convolution Neural Network (CNN). And designing a monitoring system on mobile based on IoT to remotely monitor the climatic conditions surrounding the plant and irrigate plants. Therefore, The paper proposes a monitoring system via IoT to remotely observe the state of plants regarding the environmental conditions and health status and accordingly the farmer can take action so the farmer can discover diseases, reducing the field visit to the farm because monitoring and watering are done remotely, thus improving the quality of the agricultural product. The organization of this paper arranged according to the following:
I the introduction of this paper. **II** related works. **III** the proposed system for this paper. **IV** discussion for the result obtained from this work with previous work. **V** conclusion.

II. Related Works

A few researchers have developed applications for remote plant monitoring and disease detection. **Uzhinskiy et al.[15]** Using the KNN (K-Nearest Neighbor) classifier to classify 15 types of diseases, they obtained a test accuracy of 86%. To improve the accuracy, they used single-layer perceptron estimators with a single input and output layer ending with softmax and Adam (Adaptive Moment Estimation) optimizer, obtaining an accuracy of 95.71% for 100 epochs then used Apache Cordova to build Classified mobile application but training of the network was entirely based on the 935 images collected from the Internet. **Valdoria et al.[16]** using deep learning and a set of images comprising eleven types of plant diseases with a total of 1650 images spread in the Philippines to build a system for classifying plant diseases using Android Studio. The deep learning model has been trained on images and published via Docker for use in a mobile application by android studio, used only few data are available for common plant diseases in the Philippines. **Smetanin et al.[17]** they built a mobile application to classify plant diseases in two methods. The first method is inserting the image into the platform and classifying it using deep learning. The second method is by entering a text with the type of problem and using the BERT (Bidirectional Encoder Representations from Transformers**)** model to classify similar texts. **Adedoja et al.[18]** the researchers created a mobile application using React Native to detect plant diseases by taking a picture of the affected plant. The training process used the NASNet model transfer learning method and a data set of 54,309 for 14 crops containing 32 classes of diseases. **Jasim & Al-Tuwaijari, [19]** used a proposed model for CNN to train the dataset on classifying plant diseases and building a classification interface in Matlab. In training, the network used a database from PlantVillage for three varieties (tomato, pepper, and potato) of plants with a size of 20636 images and obtained an accuracy of 98%. **Muangprathub et al.[20]** the researchers developed an IoT system that works inside the farm to monitor soil moisture and control water sprinklers automatically through a mobile application. Analyzing the data measured by temperature, humidity, and ultrasound sensors using the web application and printing all sensor values on the mobile application. **Kwok & Sun [21]** created an irrigation system based on plant type recognition from deep learning, using an algorithm previously trained with ImageNet. As for building the application that determines the type of plant, it was through Docker and Android Studio. Each plant has a certain percentage of moisture based on which the valve is opened and closed to reach the humidity limit. **Jacob et al.[22]** created a system for monitoring and classifying plant diseases using sensors and deep learning. Where temperature, soil moisture, co2, and smoke sensors were used, these values are sent to the application, analyzed, and alerts are given to the farmer. In detecting diseases, the Inceptionv3 model was used to train the data to discover plant diseases, where the model's accuracy in the test was 74.4.

III. The Proposed System

This work established a monitoring system for the tomato plant agriculture environment in two phases. The first is to monitor the plant's health status and detect the types of diseases that affect the tomato plant through deep learning models and building an application that classifies tomato leaf diseases. The second is building an IoT system for monitoring and control by using the sensors to measure the environmental factors surrounding the plant and irrigation remotely and using the application to monitor all these sensors' data. Fig.1 shows the general structure of the proposed system.

A. Deep Learning Phase

Deep learning is used to classify tomato plant diseases at this stage the classification process using deep learning went through several steps, including collecting and pre-processing the dataset and training the models, after which the best model is converted to a Tensorflow Lite Model (TFLM). Then build the application that classifies plant diseases. The following sections explain the steps in detail.

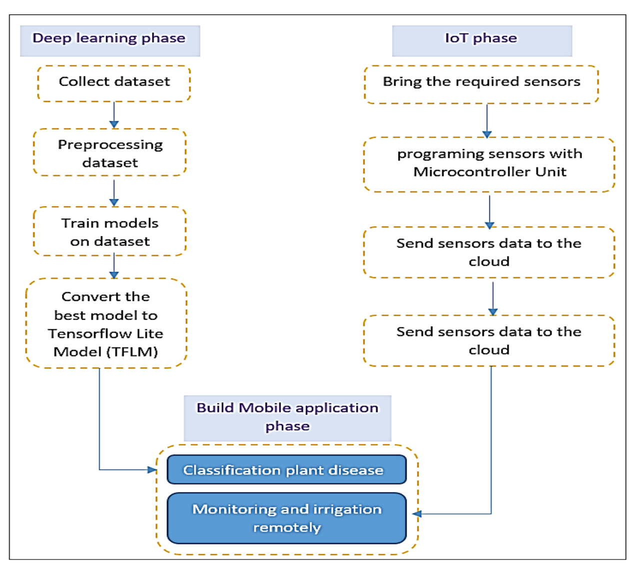


Fig. 1: General structure of the proposed system.

- *Data Collection and Preprocessing Phase*

Images of seven varieties of common tomato diseases appearing in Iraqi farms were taken from the PlantVillage tomato leaf dataset, a total of 11,192 images. Fig.3 shows a set of these images and performs pre-processing, splitting the dataset into training, validation, and testing, as shown in Fig. 4. Due to the training dataset being unbalanced do augmentation process to remove the unbalance to avoid overfitting during the training process through flipping, zooming, and brightness scales, this process generated 3,846 images. Fig.5 shows the training dataset before and after augmentation.

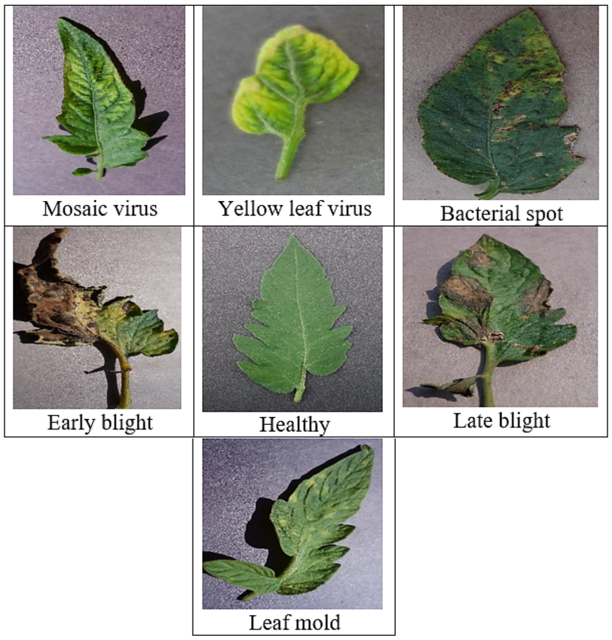


Fig. 3: Sample of the image used in training models.

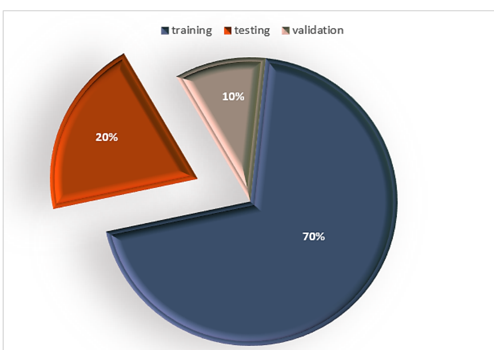


Fig. 4: The splitting of the dataset.

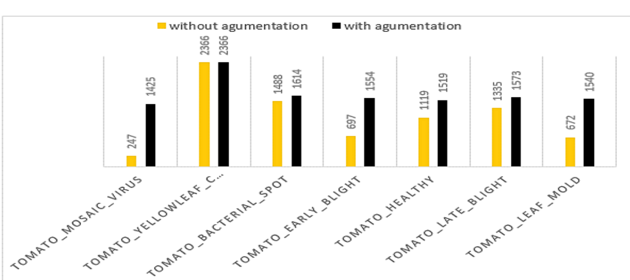


Fig. 5: number of dataset after augmentation.

We noticed that the PlantVillage data are laboratory data and lack diversity in the field environment. The model must be trained on data with various features to increase its strength in the correct classification. A data set on tomato diseases were collected from Google image, totaling 114 images. Fig. 6 shows a sample of these images. Since the dummy data obtained was few, an augmentation was also made using the same measures mentioned in the augmented in PlantVillage dataset. Table I shows the data collected from google and their number after the augmentation using the generated images in the training and accurate images for the test. And finally, resize images from (256 x 256) to (224 x 224) to fit the pre-trained network entries used.

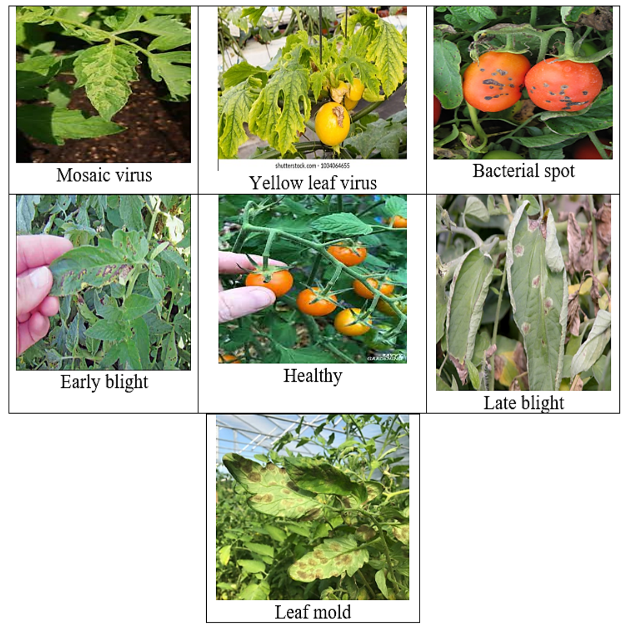


Fig. 6: Sample of tomato disease from google image.

TABLE I:
NUMBER OF IMAGES COLLECTED FROM GOOGLE AND AUGMENTED.

Tomato disease	Actual image	Augmentation image
Mosaic virus	13	130
Yellow leaf curl vires	15	120
Bacterial spot	18	126
Healthy	7	56
Early blight	21	168
Late blight	21	168
Leaf mold	19	112
Total	114	880

- *Training and Testing Phase*

After dividing the dataset, used 12,771 images in the training and validation process, where two models were used in extracting features from the images. The first model proposed for CNN consists of three main layers: the first is the input layer, the second is the hidden layers (feature extraction), and the third is the classification layer (output). Fig.7 shows the structure of the proposed model used in the training dataset.

The second model is pre-trained, MobileNet_v2, which used the transfer learning method of the model to train the dataset by taking the network weights. And fine-tune the input layer with a dataset for tomato plants and the output layer with an output layer for seven classes of plant diseases.

As for the parameters used in training the two networks, we used an optimizer of type Adam, categorical_crossentropy loss function, batch size=64, epochs=50, and accuracy metric. Table II shows the training accuracy and validation values obtained from training the two models. Fig.8 shows the accuracy and loss function curve during the training process.

Table II
ACCURACY FOR TRAINING AND VALIDATION MODELS.

Proposed method	Training accuracy	Validation accuracy
MobileNet_v2	99.89%	96.7%
Proposed method	98.15%	97.41%

Tested the two trained models on the test dataset of 2267 images and used metrics confusion matrix, precision, recall, f1-score, and accuracy in measuring the validity of the results obtained. Table III shows the values obtained during the test of the two models.

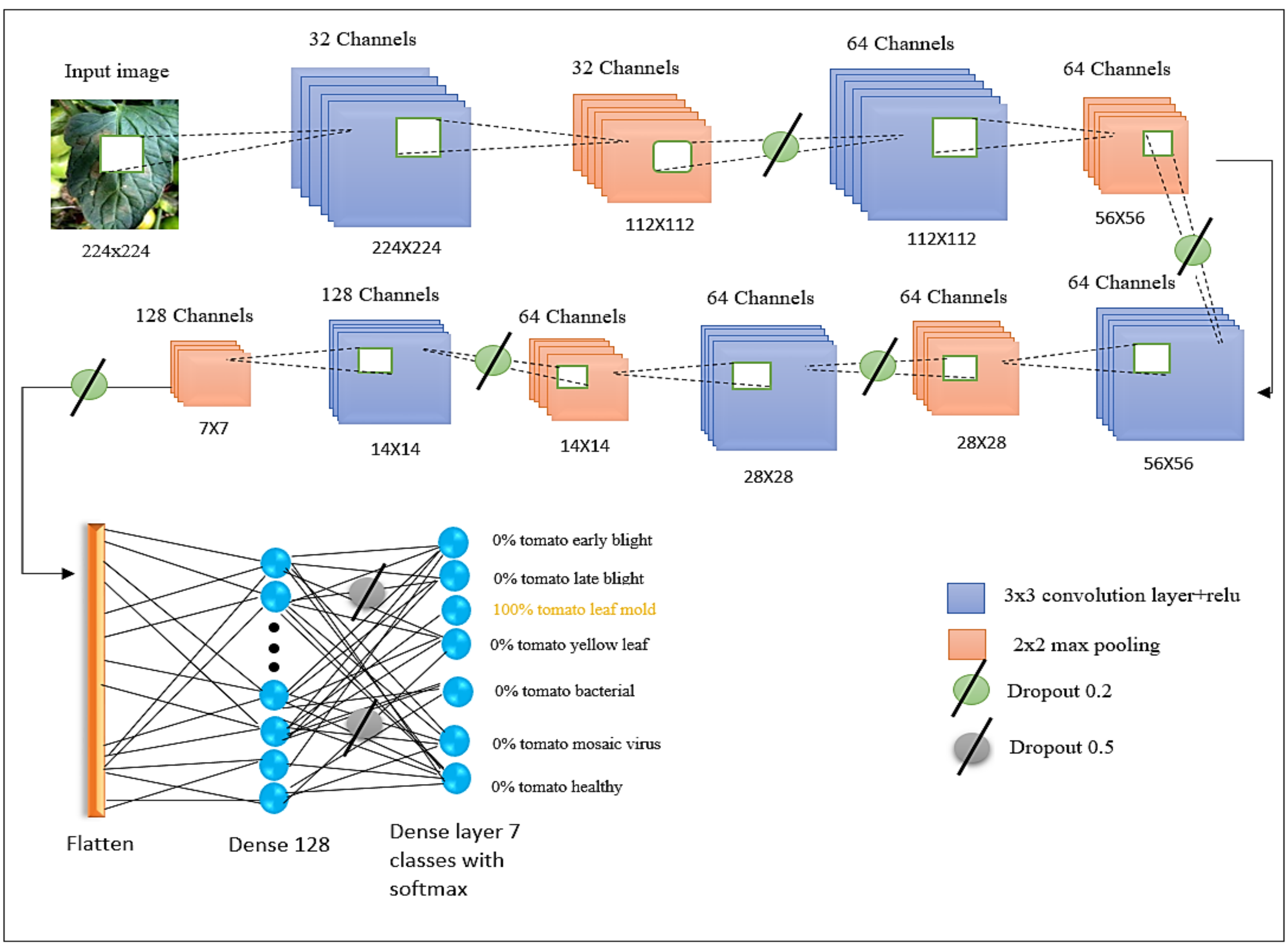


Fig. 7: The structure of the proposed model.

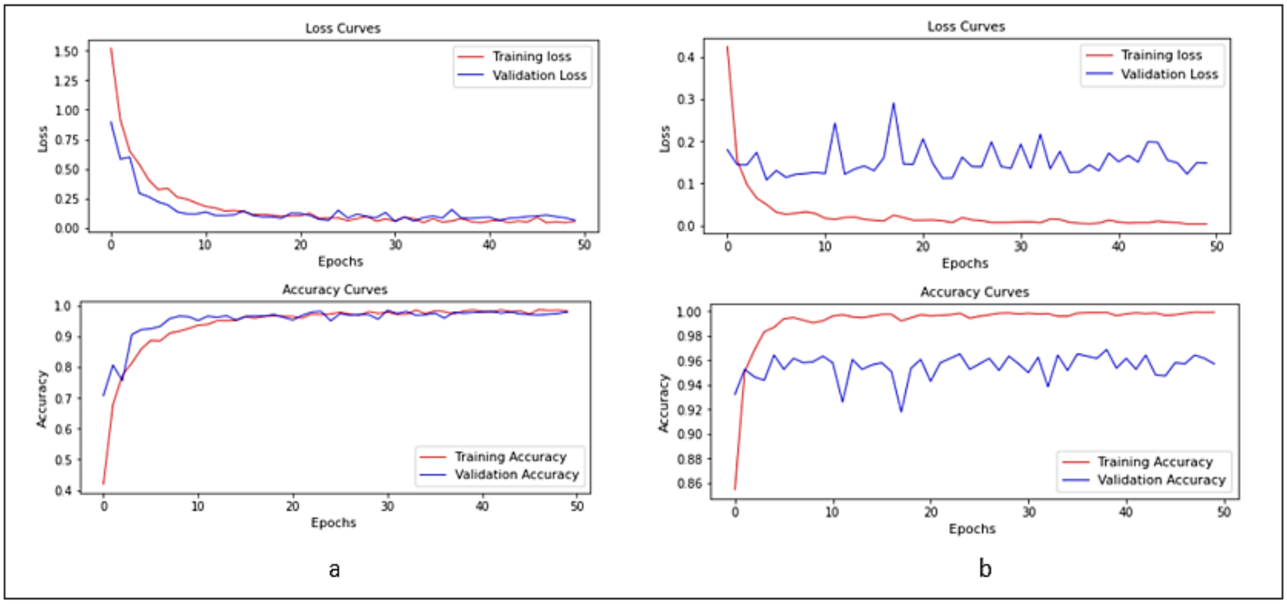


Fig. 8**:** Curve for training accuracy and loss for (a) proposed method, (b) MobileNet_v2.

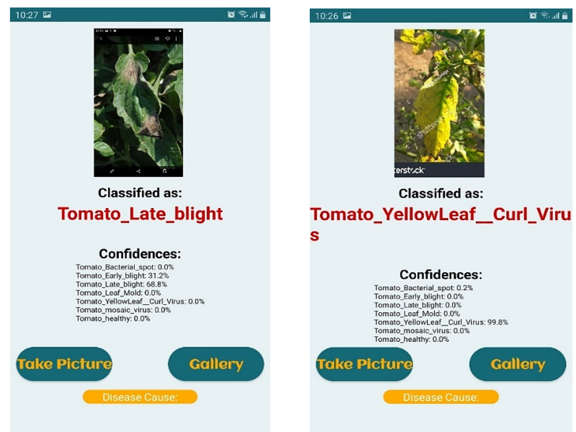
TABLE III

TESTING METRICS FOR PROPOSED METHOD AND MOBILENET_V2.

Models	Testing accuracy	Precision	recall	F1-score
MobileNet_v2	96.6%	96%	96%	96%
Proposed method	97.62%	97%	98%	97%

- *Build Classified Application Part*

At this stage, it is building a classification section for tomato plant diseases in the mobile application using Android Studio. The proposed model converted to TFLM provides many advantages, including small size and fast conclusion, enabling it to work on a mobile device with limited memory and computing. Android Studio builds the application's front end through the activity_main.xml file and the backend through the MainActivity.java file. Therefore, built the application's front end to contain two buttons (Gallery) for fetching photos from the gallery to classify them and (Take Picture) for taking photos using the mobile camera. And the third button (Disease Cause) displays the causes of the disease. After completing the step of building the classification interface, programmed the application's backend to classify tomato leaf diseases using TFLM with print classification confidence. Mobile application was run on the Android operating system and has been tested on 114 images for tomato diseases collected from the internet due to the difficulty of obtaining actual test samples. As the classification of this image is shown in Fig. 9, where (a) represents the classification image for late blight disease, and (b) for yellow leaf curl virus. Therefore, when the farmer discovers that the plant has a disease, he can know the type of disease through this phase of the application.



a b

Fig. 9**:** Test mobile app to classification google image for tomato diseases with actual classification results.

B. *IoT System Phase*

At this stage, we use the IoT system to monitor the environmental conditions of the plant and control irrigation remotely. A set of sensors is used to measure environmental conditions and programmed using a microcontroller to send data via WiFi to the cloud. Then the data is called to the mobile application for processing and decision-making. Sensors monitor the environmental conditions surrounding the tomato plant, which affect it, including temperature and humidity, soil moisture, water quality, and carbon dioxide concentration (Co_2). And use the soil moisture sensor for remote plant irrigation. Fig.10 shows the flowchart for monitoring and irrigation system remotely.

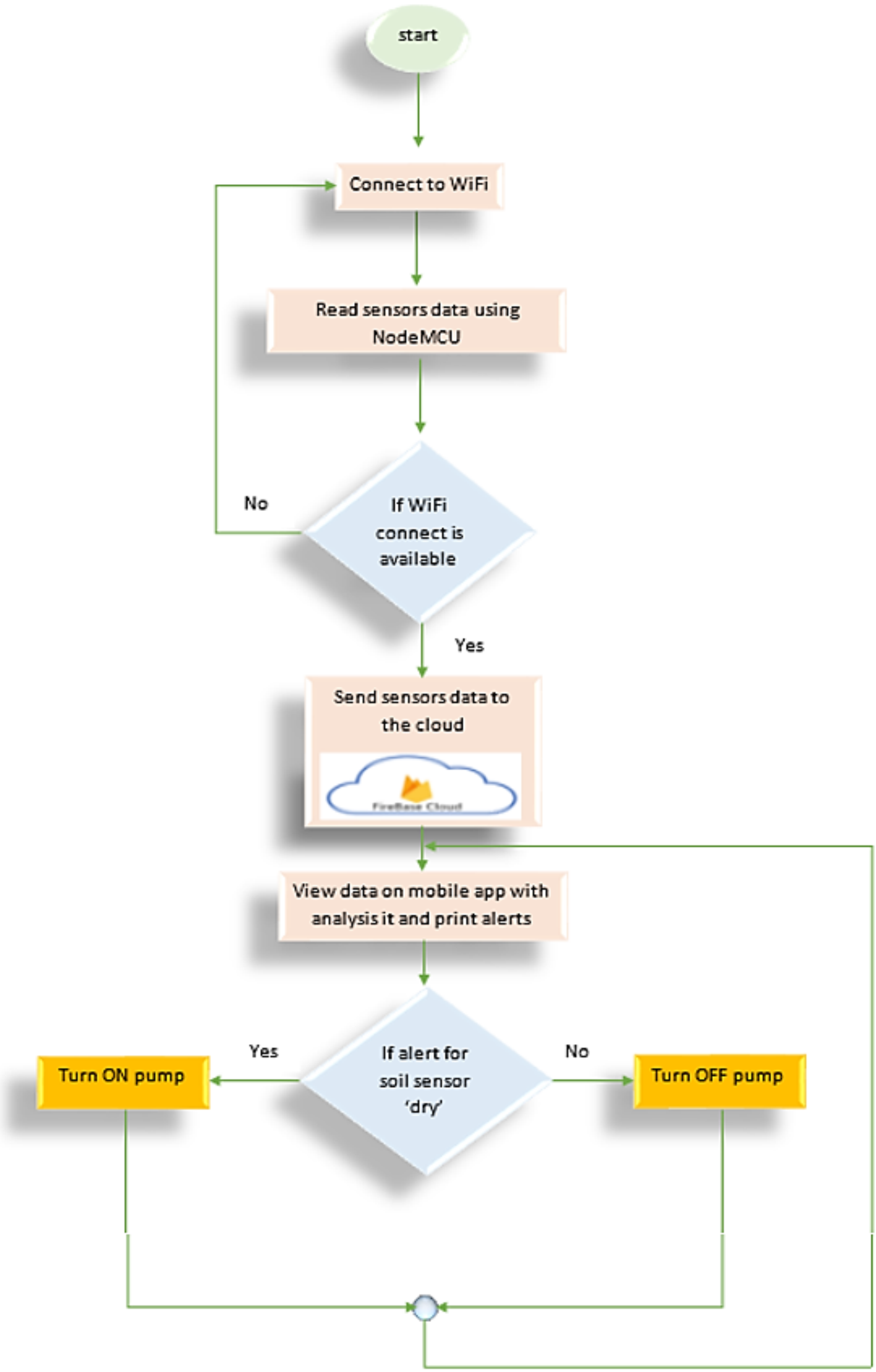


Fig. 10**:** Flowchart of IoT system for monitoring and irrigation remotely.

- *Hardware Tools for System*

Building a monitoring system requires a set of hardware tools that measure the values of the environmental conditions surrounding the plant includes:

1- *DHT22* (Digital Humidity and Temperature) is a sensor for measuring the temperature and humidity of the atmosphere surrounding the plant. It was characterized by its accuracy, low cost, and ease of reading its digital signal by the controller. Its temperature reading ranges between (-40-80°C) and its humidity reading between (0-100%). Use this sensor because the temperature and humidity of the air are among the most critical factors that affect the photosynthesis process.

2- *Analog soil moisture* sensor, which measures the percentage of water available in the soil for automatic irrigation. To improve the water use of crops and reduce the appearance of root rot diseases due to the high humidity of the roots.

3- *MQ 135* sensor to measure the percentage of CO2 gas in the air surrounding the plant for its necessity in the process of photosynthesis and plant growth. Its ratio in the air ranges (0-3000 ppm) and the useful ratio of the plant is between (300-1800 ppm).

4- *TDS* (Total Dissolved Solid) analog sensor to measure water quality for irrigating crops, as it measures the percentage of dissolved solids in water in PPM unit, where its value ranges (0-500 ppm) suitable for irrigation, but more than 500 ppm is invalid.

5- *The water pump* is used to pump water to the farm. It operates at a voltage of 12 V, weighs 69 G, and pumps 240 L/H.

6- *Relay* to control of the water pump through the digital signal.

7- *NodeMCU* (Node Micro Controller Unit) ESP8266 Wireless has powerful on-board processing and storage that allows it to be combined with other sensors and devices through its GPIO modules. It features nine digital signals and one analog signal.

8- *16 Analog Multiplexer* is used to control the analog signals digitally. It is used here because our work needs three analog signals while the NodeMCU contains only one analog signal.

9- *Rechargeable lithium battery* with a voltage of 3.7 to run water pump.

- *Software Tools for System*

To program the sensors and build the monitoring and irrigation system, we need to open source **Arduino IDE**, version 1.8.16. use to program the NodeMCU with the sensors. **Firebase cloud** is a platform developed by Google for realtime mobile and web applications to store sensor values in the realtime database. Using **Android Studio** for building part of the monitoring and control system application. The steps below illustrate the process of building a monitoring and irrigation system.

- *Design Monitoring & Irrigation System*

At this stage, the monitoring and control system was designed following several steps: sensors are programmed through the Arduino IDE program, and the code is uploaded to the NodeMCU board. The control of the water pump is done by connecting it to a relay that works as an electric switch by entering a control signal into it by the controller board. And through the Esp8266 piece on the board, NodeMCU sends the sensor values to the firebase cloud every half hour via WiFi. Fig.11 shows the final design of the IoT system.

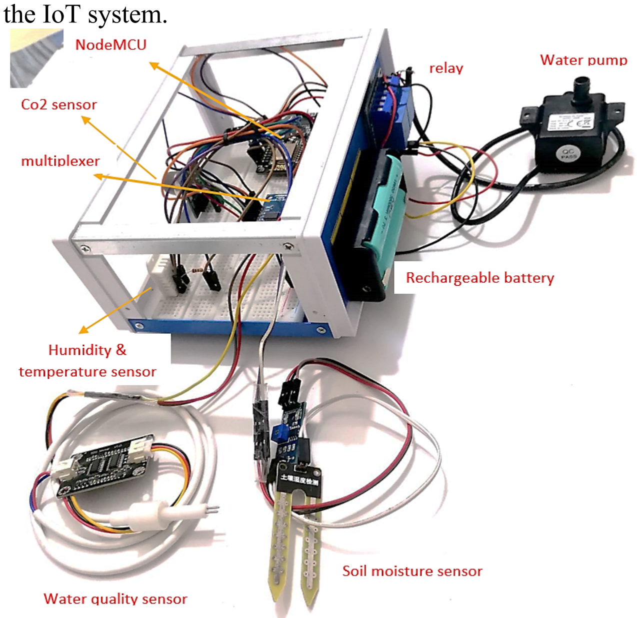


Fig. 11**:** IoT system design.

- *Build Monitoring and Irrigation Application Part*

At this stage, created the application section is responsible for monitoring plant and irrigation remotely using Android studio. First, Android Studio is connected to the Firebase to read and deal with the sensor values. In the monitoring section, the application's front end is built, consisting of four donut progress bar sfor humidity, temperature, Co2, and TDS for water, through which data from the Firebase is read and seen, with alerts printed for each read value. The remote irrigation interface contains one donut progress bar to read the value of the soil moisture sensor and print the alert message. Through the read, value farmers can use the ON and OFF buttons on the irrigation interface to opening and closing of the water pump. The sensor values are read inside the mobile application, processing and printing the values in the interface for climate monitoring. Fig.12 monitoring interface contains the values of temperature, humidity, co2, and water quality .With alerts set for each sensor if the read values exceed the minimum or maximum allowed for tomato plants.

In the plant irrigation section, the farmer is controlled by turning ON and OFF the motor, as shown in Fig 13, where the soil moisture sensor reads the soil moisture value. If the value of the sensor is 500-600, the soil is dry, he starts the motor, and then the value changes between 100-300, meaning the soil is wet. But if the sensor reads another value, it is moving from the soil, and its reading is wrong.

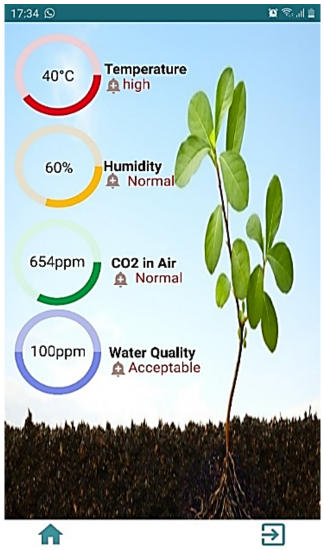
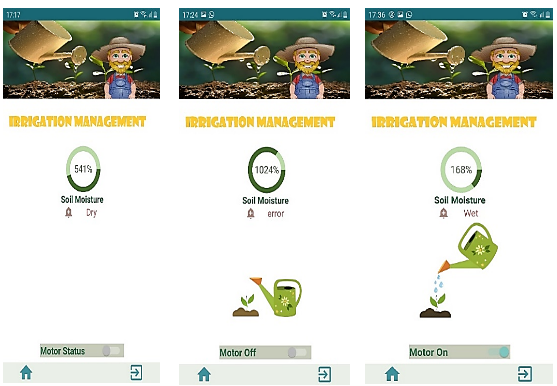


Fig. 12**:** Read sensors data.



a b c

Fig. 13: Irrigation system where (a): sensor read dry soil, (b): motor on and soil is wet, (c) represent error reading for the sensor.

IV. Discussion

The monitoring system designed using deep learning and IoT is proven effective at work. In terms of deep learning, we have discovered its effectiveness in classifying tomato diseases with high accuracy of 97%, even with field images that lack training. Thus, the farmer can rely on it to determine the plant's type of disease with high accuracy. In terms of monitoring using sensors, the system has proven its effectiveness in reading sensor values and giving alerts when tested. Thus, the monitoring system is like a farmer's eye on his farm. He can monitor it remotely by alerting him if the read values of temperature, humidity, soil moisture, water quality, and co2 exceed the permissible limit for the plant. The automatic irrigation system was the complementary part of this system. For the farmer to have the ability to control and monitor his farm fully, the system proved its ability to irrigate crops remotely through the farmer's control of opening and closing the water pump according to the reading alert message. Fig. 14 IV shows the comparison of our system with previous related work.

The system	Deep learning technique	Dataset used	Testing accuracy	IoT technique	Mobile or web application technique	The objective of the system
Alajrami & Abu-naser [23]	✓	5266 image of tomato type	93% for proposed CNN	-	-	Determining the type of tomato crop
Alhasnawi et al. [24]	-	-	-	✓	Web application	Smart irrigation system
Badran & kashmoola [25]	-	-	-	✓	-	Smart irrigation system
Faisal shenan et al. [26]	-	-	-	✓	Web application	Monitoring and control on temperature, light, and smart irrigation
Al-Akkam & Altaei [27]	✓	34,934 images for 15 classes of plant disease	96.18% for the proposed CNN	-	-	Classify plant disease
Jacob et al.[22]	✓	Public dataset	74%	✓	Web application	Monitoring and classifying image
Our system	✓	11,192 images for tomato plant	97.62% for the proposed CNN	✓	Mobile application	Classify tomato diseased, and monitor weather conditions with smart irrigation

Fig.14: Comparing the proposed system with a group of previous methods.

V. Conclusions

At the end of this work, one can conclude that there is an attempt to create a smart farm utilizing the power of Deep Learning and IoT techniques. Both are integrated via developing a mobile application using android studio tools. The proposed system enables the farmer to manage his farm without needing residency inside the farm. The resultant smart farm collects the environmental information that affects crop status utilizing the IoT assets. analyze this information to issue timely alerts to the farmers through the mobile application. This alert enables the farmer to make suitable decisions to protect his crop from disasters. Thus, increasing the quality and quantity of production while reducing the losses, improving the human labor used, and reducing the challenges facing the farms at present, including the increase in temperatures, the change in the level of rainfall, the lack of water, and Plant diseases and the difficulty of identifying them and others.

And by using deep learning, the farmer can easily identify the type of disease that the plant has. Our system designed to classify tomato diseases exceeded the accuracy test 97% by using 15,918 images from PlantVillage and google images after augmenting it to solve unbalanced and A few field images in google. And thus, it can be relied upon in classifying and taking the appropriate action to prevent the spread of the disease without the need to call agricultural experts, thus saving more time and money.

By using the sensors for measuring the environmental conditions surrounding the plant, the farmer can monitor his farm remotely without needing to visit it in the field, and he can irrigate his farm remotely using the automatic irrigation system. The work of the application is not limited to monitoring conditions only. It also can send alerts to the farmer by reading the values of the sensors and comparing them with a threshold limit for the appropriate climatic conditions for the tomato plant. If it exceeds it, the farmer is alerted to take the proper action to reduce its impact on the plant. Since the designed system has proven its efficiency in classifying tomato diseases and in monitoring the environmental conditions of the plant and automatic irrigation.

CONFLICT OF INTEREST

The authors have no conflict of relevant interest to this article.

REFERENCES

[1] Jaradat, “Agriculture in Iraq: Resources, potentials, constraints, research needs and priorities,” *Agriculture*, p.83, 2003. [Online]. Available: http://ddr.nal.usda.gov/dspace/handle/10113/47863.

[2] H. Abbas Drebee and N. Azam Abdul-Razak, “The Impact of Corruption on Agriculture Sector in Iraq: Econometrics Approach,” *IOP Conf. Ser. Earth Environ. Sci.*, vol. 553, no. 1, 2020 doi: 10.1088/1755-1315/553/1/012019.

[3] M. Abdulla and A. Marhoon, “Agriculture based on Internet of Things and Deep Learning,” vol. 18, Issue 2, pp. 1–8, 2022. doi: 10.37917/ijeee.18.2.1.

[4] M. S. Farooq, S. Riaz, A. Abid, K. Abid, and M. A. Naeem, “A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming,” *IEEE Access*, vol. 7, pp. 156237–156271, 2019. doi: 10.1109/ACCESS.2019.2949703.

[5] O. Elijah, S. Member, T. A. Rahman, and I. Orikumhi, “An Overview of Internet of Things (IoT) and Data Analytics in Agriculture : Benefits and Challenges,” vol. 4662, no. c, pp. 1–17, 2018. doi: 10.1109/JIOT.2018.2844296.

[6] V. Meshram, K. Patil, V. Meshram, D. Hanchate, and S. D. Ramkteke, “Machine learning in agriculture domain: A state-of-art survey,” *Artif. Intell. Life Sci.*, vol. 1, no. October, p. 100010, 2021. doi: 10.1016/j.ailsci.2021.100010.

[7] K. Lova Raju and V. Vijayaraghavan, "IoT Technologies in Agricultural Environment: A Survey", Springer US, vol. 113, no. 4. 2020.

[8] J. Ye, B. Chen, Q. Liu, and Y. Fang, “A Precision Agriculture Management System Based on Internet of Things and WebGIS,” 2013 21st International Conference on Geoinformatics, 2013.

[9] R. Mathur, V. Pathak, and D. Bandil, "Emerging Trends in Expert Applications and Security", Proceedings of ICETEAS 2018, 2019.

[10] M. Mannan J, S. Kanimozhi Suguna, M. Dhivya, and T. Parameswaran, “Smart scheduling on cloud for IoT-based sprinkler irrigation,” *Int. J. Pervasive Comput. Commun.*, vol. 17, no. 1, pp. 3–19, 2021, doi: 10.1108/IJPCC-03-2020-0013.

[11] D. Sehrawat and N. S. Gill, “Smart sensors: Analysis of different types of IoT sensors,” *Proc. Int. Conf. Trends Electron. Informatics, ICOEI 2019*, no. Icoei, pp. 523–528, 2019, doi: 10.1109/ICOEI.2019.8862778.

[12] P. Daponte *et al.*, “A review on the use of drones for precision agriculture,” *IOP Conf. Ser. Earth Environ. Sci.*, vol. 275, no. 1, 2019, doi: 10.1088/1755-1315/275/1/012022.

[13] B. Faithpraise and C. Chatwin, "Automatic Plant Pest Detection And Recognition Using K-Means Clustering Algorithm And Correspondence Filters," International Journal of Advanced Biotechnology and Research, vol. 4, no. 2, pp. 189–199, 2013.

[14] N. Soontranon, P. Tangpattanakul, P. Srestasathiern, and P. Rakwatin, "An Agricultural Monitoring System : Field Server Data Collection and Analysis on Paddy Field," 14th International Symposium on Communications and Information Technologies (ISCIT), 2014.

[15] A. Uzhinskiy, G. Ososkov, P. Goncharov, and A. Nechaevskiy, “Multifunctional platform and mobile application for plant disease detection,” *CEUR Workshop Proc.*, vol. 2507, pp.110–114, 2019.

[16] J. C. Valdoria, A. R. Caballeo, B. I. D. Fernandez, and J. M. M. Condino, “IDahon: An Android Based Terrestrial Plant Disease Detection Mobile Application Through Digital Image Processing Using Deep Learning Neural Network Algorithm,” *Proc. 2019 4th Int. Conf. Inf. Technol. Encompassing Intell. Technol. Innov.Towar. New Era Hum. Life, InCIT 2019*, pp. 94–98, 2019 doi: 10.1109/INCIT.2019.8912053.

[17] A. Smetanin, A. Uzhinskiy, G. Ososkov, P. Goncharov, and A. Nechaevskiy, “Deep learning methods for the plant disease detection platform,” *AIP Conf. Proc.*, vol. 2377, no. September, 2021, doi: 10.1063/5.0068797.

[18] A. O. Adedoja, P. A. Owolawi, T. Mapayi, and C. Tu, “Intelligent Mobile Plant Disease Diagnostic System UsingNASNet-Mobile Deep Learning,” *IAENG Int. J. Comput. Sci.*, vol. 49, no. 1, pp. 216–231, 2022.

[19] M. A. Jasim and J. M. Al-Tuwaijari, “Plant Leaf Diseases Detection and Classification Using Image Processing and Deep Learning Techniques,” *Proc. 2020 Int. Conf. Comput. Sci. Softw. Eng. CSASE 2020*, pp. 259–265, 2020, doi: 10.1109/CSASE48920.2020.9142097.

[20] J. Muangprathub, N. Boonnam, S. Kajornkasirat, N. Lekbangpong, A. Wanichsombat, and P. Nillaor, “IoT and agriculture data analysis for smart farm,” *Comput. Electron. Agric.*, vol. 156, no. June 2018, pp.467474,2019. doi: 10.1016/j.compag.2018.12.011.

[21] J. Kwok and Y. Sun, “A smart IoT-based irrigation system with automated plant recognition using deep learning,” *ACM Int. Conf. Proceeding Ser.*, pp. 87–91, 2018, doi: 10.1145/3177457.3177506.

[22] P. M. Jacob, S. Suresh, J. M. John, P. Nath, P. Nandakumar, and S. Simon, “An Intelligent Agricultural Field Monitoring and Management System using Internet of Things and Machine Learning,”2020 Int. Conf. Data Anal. Bus. Ind. Towar. a Sustain. Econ. ICDABI 2020, 2020, doi:10.1109/ICDABI51230.2020.9325612.