Detection of Tomato Growth State and Surveillance System using Computer Vision and Internet of Things

[1]Abhilash, [2]Fariya Bano, [3]Swetha R, [4]Syeda Simran Baseer

[1]Assistant Professor, [2,3,4]Student
Department of Information Science & Engineering,
Atria Institute of Technology, Bangalore, India.

Abstract : The main objective of this project is to develop a system for plant monitoring and watering using the Internet of Things and Computer Vision. With Raspberry Pi as Processor, and sensors for sensing environmental conditions, the system monitors different parameters like Temperature, Humidity and Soil Moisture. We will also conduct the image analysis by combining the deep learning for tomato fruit detection and image processing for color feature extraction to classify the fruit's maturity into a binary result of its growth stage (Eg., ripe or not ripe). In this work, we will collect images and detect the fruit section on images where the deep learning model will be trained to detect these fruit regions, then segment them from its background that may consist of soil and leaves. Finally, the system will notify farmers using an IoT notification platform. The observational phase of crop growth is essential to determine the quality aspect of agricultural sciences. This system enables the farmer to accomplish well-timed interventions that warrant for optimal yields at the end of the season. With the necessary information available regarding the problems that farmers face, we can accurately target those specific areas, while also accomplishing the efficient use of resources.

***IndexTerms* - Internet of Things (IoT), Bot Notification, Tomato Maturity, Deep Learning, Image Processing, Computer Vision (CV)**

I. INTRODUCTION

These days farming is considered a risky profession. According to the Consensus carried out by The Central Govt of India, 2015-16, the number of farmer families amounted to 14.5 crore. It is estimated that the numbers have dwindled since then. India is an agricultural country. Agriculture is only ~16% of the GDP, but it is the largest sector for employment. Tomato is considered one of the most important commercial plants that is widely cultivated in India while being the second most important crop in the World after potatoes. The risk in cultivating tomatoes comes from the unreliable identification of its maturity.

Presently, human experts grade agricultural goods based on their visual features. Nevertheless, manual inspection causes inaccuracy, inconsistency and inefficiency in defining the quality of agricultural goods. It is human nature to get weary or become distracted after a certain amount of time. The success in the automatic grading system of fruit on the basis of quality will result in accurate, efficient and consistent outputs. As a result, it optimizes on time and human labour and will provide immense help to economic development.

Generally, the quality of any fruit is majorly determined by its colour, shape and size. These factors are responsible for identifying the maturity level of tomatoes as well. Ripening of fruit is a compound occurrence that is conspicuous through chemical and physiological changes. Hence, consumers associate the internal qualities of a fruit through its external appearance. For a fruit like tomato that is delicate, it is essential to handle its grading carefully. Not every individual has the ability of sorting the fruit perfectly based on its shape, size and colour. This leads to multiple drawbacks in the manual sorting procedure, such as poor efficiency, slow speed, higher chances of error, high cost and complexity.

As a solution to the above-mentioned problems, an automated monitoring system was developed based on the concepts of computer vision and IoT. Image analysis is done by combining the deep learning for tomato fruit detection and image processing for colour feature extraction. Concepts like thresholding, erosion, dilation and distance transform are used to detect and classify the tomato as ripe or unripe. Thus, the colour identification system can classify tomatoes on the basis of their ripeness. Geometric features determination system approximates the size and shape of the tomato. The IoT section includes a DHT sensor and an IoT notification platform (Blynk). Based on sensor value, components like fan and water pump are operated through relays to provide the optimal plant growth conditions. The users can choose to do so via the app wherein they are also notified about the maturity of the tomato.

II. GENERAL STRUCTURE

The Internet of Things (IoT) is defined as a model in which objects equipped with sensors, actuators, and processors communicate with each other to serve a purpose. The below figure shows the structure of an IoT model. Data from the sensors is passed to a gateway i.e, a Raspberry pi board, arduino etc., where the data is processed and sent to a mobile application via a cloud service. It is possible to control certain devices such as fans, water pump, lights etc., via the mobile app.

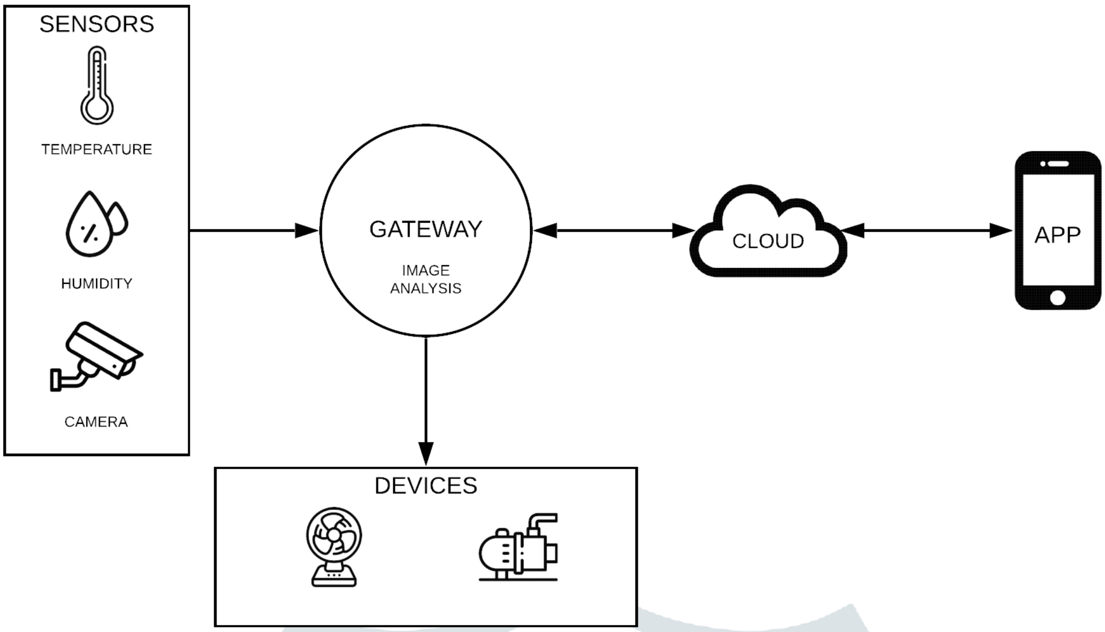

Figure 1: General Structure of an IoT model

III. Related Survey

3.1 Thresholding

It is a technique in OpenCV used to convert an image into binary format. In this technique pixels are assigned values in relation(comparing) to the given threshold value.

Thresholding is usually done on grayscale images. A grayscale image has pixels whose values range from 0-255. Thresholding the grayscale image helps to classify the pixels into groups, setting an upper and lower bound to each group. The two groups could be:

- Pixels in range 0-127
- Pixels in range 128-255

Now all the pixels corresponding to (i) are converted to 0 (black) and the pixels corresponding to (ii) are converted to 1(white). The resulting image is a binary image in which all pixels values are either 0 or 1.

Types of thresholding:

- Global Thresholding:
  This type of thresholding can be applied only when there is a distinct difference in the intensities of background and foreground pixels.
  If g(x, y) is a threshold version of f(x, y) at some global threshold T,

$$g(x,y) = \begin{cases} = 1 & \text{if } f(x,y) \geq T \\ = 0 & \text{otherwise} \end{cases}$$

- Local Thresholding:
  This is used when there is uneven illumination due to shadows or direction of illumination. If T is the threshold which depends on the grayscale level of f(x,y) then local thresholding can be given by:

$$g(x, y) = \begin{cases} 0 & \text{if } f(x, y) < T(x,y) \\ 1 & \text{if } f(x, y) \geq T(x, y) \end{cases}$$

$$\text{Where } T(x, y) = f(x, y) + T$$

- Adaptive Thresholding:
  Adaptive thresholding is a technique that is applied on a grayscale image and it outputs a binary image with segmentation. The threshold is calculated for each pixel of the image. If the value of the pixel is below the threshold, it is set to the background value, else it is set to the foreground value.

3.2 Past Research of Morphological Transformations

Morphology is an image processing technique that is based on the structure and proportion of objects. By using morphological methods, a structuring element is applied to an input image and an output image is created at the same size.

Morphological operations rely only on the relative ordering of pixel values rather than on their numerical values. Therefore they are especially suited to the processing of binary images. Morphological operations can also be applied to grayscale images such that their light transfer functions are undisclosed. Therefore their absolute pixel values are of no or insignificant interest. An image is probed in morphological techniques using a structuring element which is a small shape or template.

This structuring element is placed at all possible locations in the image and it is then compared with the corresponding neighbourhood of pixels. There exist certain morphological operations that test whether the element "fits" within the neighbourhood, while others test whether it "hits" or intersects the neighbourhood.

The most fundamental morphological operations are erosion, dilation, opening and closing.

- Erosion:
  For the computation of erosion of a binary input image by the structuring element, we consider each of the foreground pixels in the input image in succession. We require the origin of the structuring element to coincide with the input pixel coordinates, hence for each input pixel we superimpose the structuring element on top of the input image. Erosion eliminates small-scale details from a binary image but simultaneously minimizes the size of regions of interest. The erosion of the binary image A by the structuring element B is defined by:

$$A\,!\;\; B=\{z\in E|B_{z}\subseteq A\}$$

- Dilation:
  Dilation is a transformation that produces an image that is the same shape as the original, but is a different size. Enlarging the areas of foreground pixels is the primary effect of dilation on binary images. The objective of morphological dilation is to make objects more conspicuous and fill in minor holes in objects. The dilation of A by the structuring element B is defined by:

$$A\oplus B=\bigcup_{b\in B} A_b$$

- Opening:
  The opening operation consists of eroding an image and then dilating the eroded image, using the same structuring element for both operations. Morphological opening is practical for eliminating small objects from an image while also preserving the structure and dimensions of larger objects in the image.
- Closing:
  The closing operation consists of dilating an image and then eroding the dilated image, using the same structuring element for both operations. Morphological closing is practical for packing small holes from an image while preserving the structure and dimensions of the objects in the image.

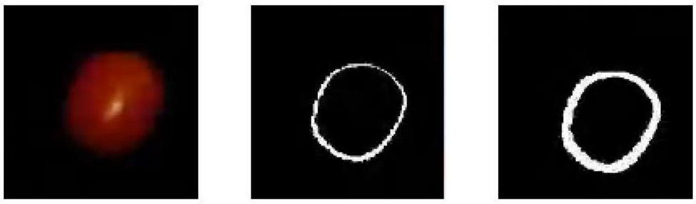

IMAGE EROSION DILATION

Figure 2: Morphological Transformation applied on Tomato

3.3 Distance Transform

Usually distance transform is applied to binary images. The transformed output is a gray level image in which the points inside the foreground have intensities in proportion to their distance from the closest boundary.

Every point in the foreground is given a value in accordance with the minimum distance from the nearest background point. For every point in the background, a value is assigned in accordance with the minimum distance from the nearest foreground point. If the value is 1, it indicates a point in the foreground. In contrast, 0 indicates that the point is in the background.

The chosen metric qualifies the transform. Metrics can be one of the following:

- Euclidean Distance: It is the normal distance between the two points. The distance between two points p and q is defined as the square root of the sum of the squares of the differences between the corresponding coordinates of the points. The two-dimensional euclidean geometry, the euclidean distance between two points *a = (ax, ay)* and *b = (bx, by)* is defined as,

$$d(a,b)=\sqrt{(bx-ax)^2+\ (by-ay)^2}$$

- Manhattan Distance: It is the distance between the two points measured at right angles, along the axes. It is simply the sum of the horizontal and vertical components. If $u=(x_1,y_1)$and $v=(x_2,y_2)$ are two points, then the manhattan distance between *u* and *v* is given by,

$$MH(u,v)=|x_1-\ x_2|+|y_1-\ y_2|$$

- Chebyshev Distance: It is also known as maximum value distance. It is the absolute magnitude of the difference between the coordinates of points.The Chebyshev distance between two variables X and Y is defined as,

$$d=\max_i|X_i-Y_i|$$

- Chamfer Distance: The chamfer distance relatively well approximates the euclidean distance and is widely used because of its relatively small computational requirements as it imposes only 2 scans of the n-dimensional image independently of the dimension of the image. The chamfer distances are widely used in image analysis of the euclidean distance with integers. The chamfer distance d_μ between 2 points A and B is the minimum of the associated costs to all the paths P_{AB} from A to B is computed as,

$$d_\mu(A, B) = \min_{P_{AB}} W(P_{AB})$$

- Chessboard Distance: The chessboard distance is a metric defined on a vector space where the distance between two vectors is the greatest of their differences along any coordinate dimension. In two dimensions, i.e. plane geometry, if the points P and Q have Cartesian coordinates *(x1, y1)* and *(x2, y2)*, their chessboard distance is,

$$D_{chess} = \max(|x_2 - x_1|, |y_2 - y_1|)$$

Advantages:
- It is significantly faster.
- It generates more data points.
- It achieves a true radial dilation.

IV. METHODOLOGY

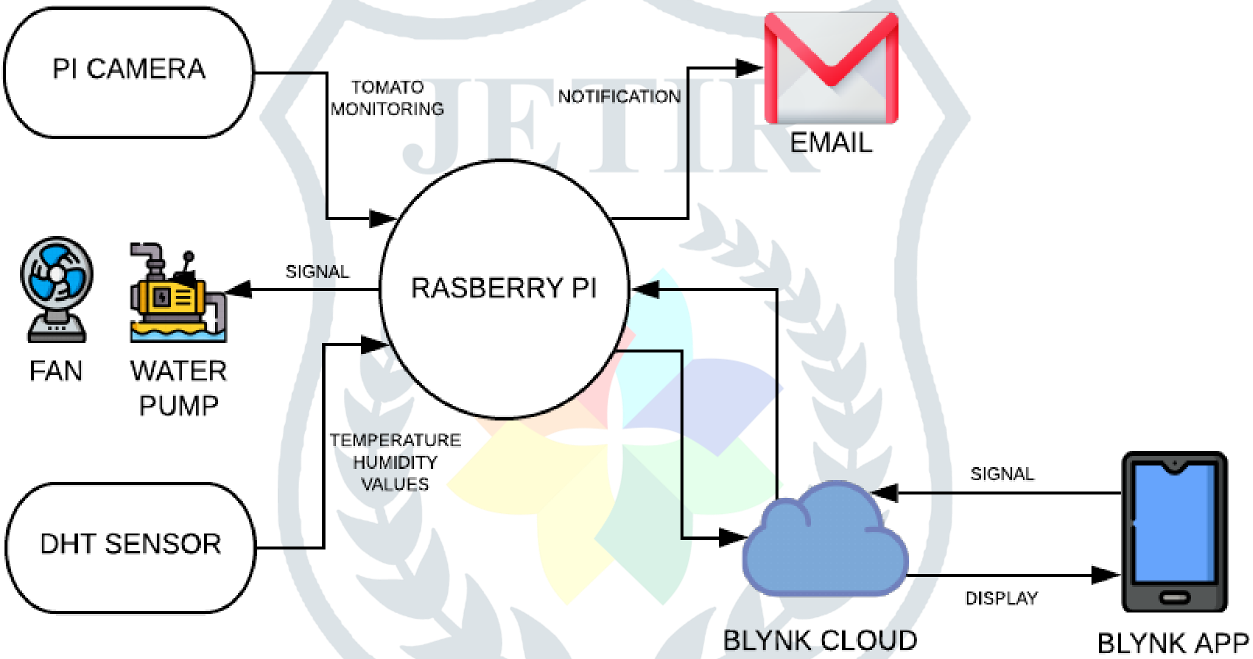

Figure 3: Architecture Diagram

This model involves two domains: Image Processing and IoT. Image processing is concerned with the monitoring of the tomato plant maturity. IoT deals with regulating and providing optimal tomato growth conditions.

4.1 Image Processing:

The PI camera constantly monitors the tomato plant . Image analysis is performed on each frame of the video captured at a certain frame rate. Each frame is converted to a grayscale image, then a threshold is applied to it followed by morphological and distance transform to reduce the noise in the image. Finally the image is analysed for the percentage of red colour present. If the percentage is above a certain threshold then the tomato is classified as ripe. An email containing the image of the ripe tomato is sent to the user.

4.2 IoT:

The central component of the system is the Raspberry PI which is responsible for all of the computation. The Raspberry PI is connected to the DHT sensor, pi camera, the relays and the cloud. The relays are in turn connected to a fan and water pump and the cloud is connected to the Blynk App. The PI Camera connected to the Raspberry PI, continuously captures the image frames of the tomato fruit. These image frames are transmitted to the Raspberry PI, where it determines its maturity state using the aforementioned algorithms. The DHT sensor provides information about the humidity and temperature in the vicinity of the plant. This information is transmitted to the owner of the plant. The relays connected to the fan and water pump can be controlled through the app, hence, providing complete control of the external factors that affect the plant to the owner. The fan and water pump are powered by an external power supply.

Requirements:

1. Raspberry Pi: The Raspberry Pi is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins that allow you to control electronic components for physical computing and explore the Internet of Things (IoT).

Figure 4: Raspberry PI

2. Relay: A relay is an electromagnetic switch operated by a relatively small electric current that can turn on or off a much larger electric current. The heart of a relay is an electromagnet.

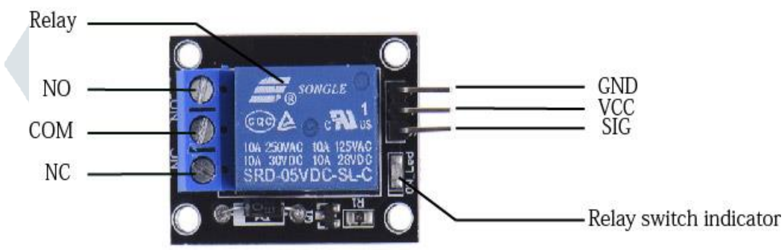

Figure 5: Relay

3. DHT Sensor: DHT is a low-cost digital sensor for sensing temperature and humidity.

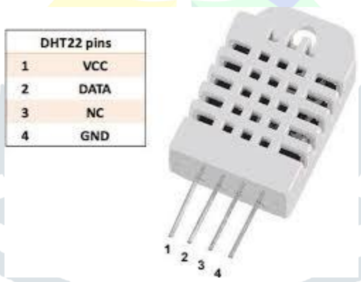

Figure 6: DHT Sensor

IV. RESULTS AND DISCUSSION

The main component of our model is the raspberry pi which performs both image analysis and IoT functions. The image below shows the prototypical setup of the system.

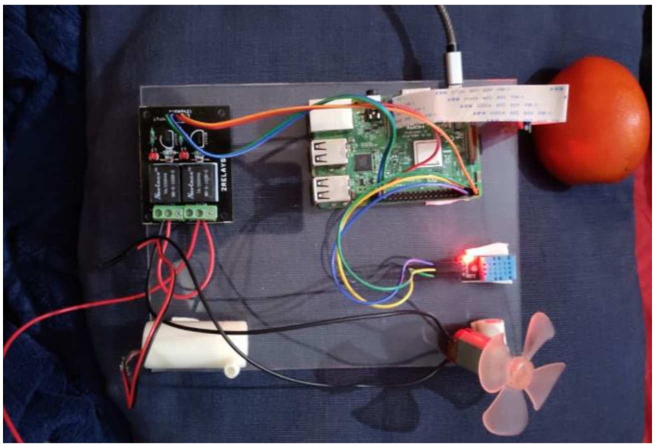

Figure7: Real time model

Our project has achieved successful results in its fundamental objective of detecting a tomato fruit's maturity state as ripe. This provides required information to the owner of the time when the fruit is ripe for picking. Figure 8 shows the red color detection of the ripe tomato after which a notification is sent via our notification systems.

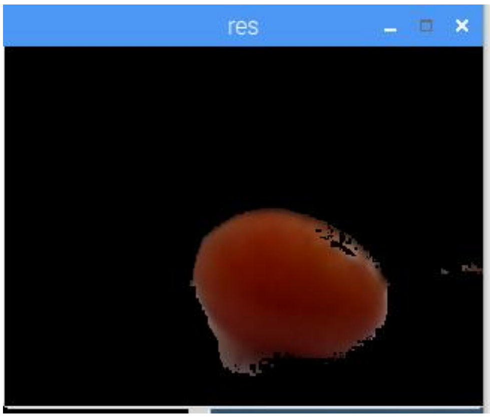

Figure 8: Tomato detection

This notification system informs of the fruit's ripe state and is sent to the owner through the Blynk App.

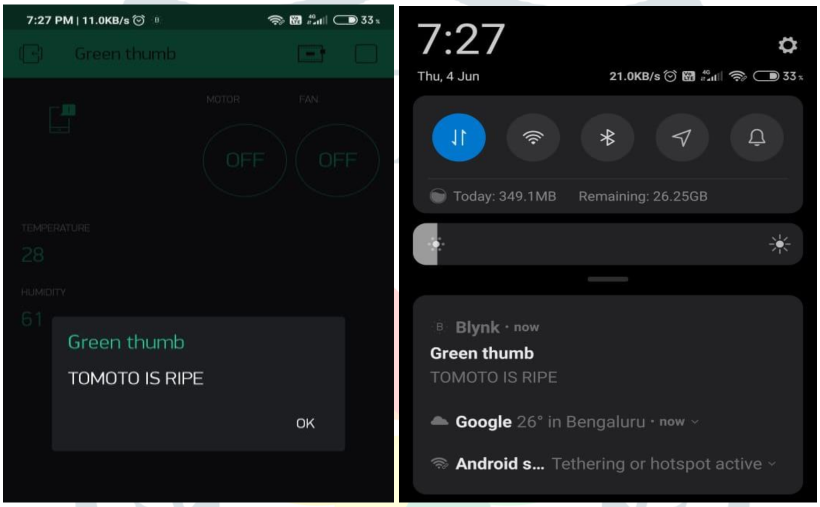

Figure 9: Blynk App notification

The system has also been made capable of sending notifications via Email. The figure below shows the Email sent to the owner consisting of images captured when the fruit is at its ripened stage.

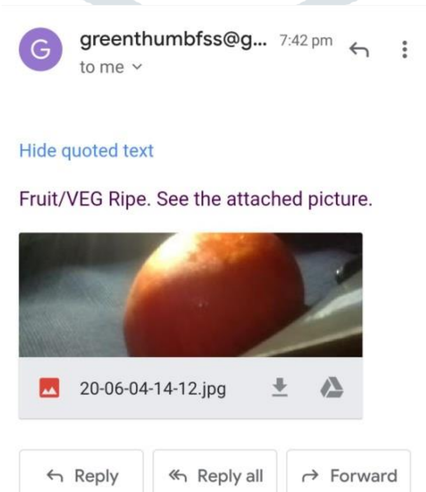

Figure 10 : Email notification with image

The values of humidity and temperature that have been measured by the sensors from the locality of the plant are displayed on the Blynk App. This feature assists the owner to better care for the plant while equipping them with the necessary information to operate the fan and water pump.

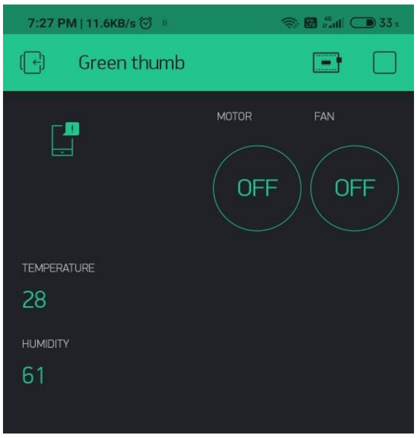

Figure 11 : Temperature and humidity values displayed along with device control buttons.

REFERENCES

[1] Arda Aydin, Anıl Çay and Burak Polat, “Image Analysis Applications in Plant Growth and Health Assessment”. Journal of Agricultural Faculty of Mustafa Kemal University, vol. 4, pp 229-230, 2017, pp. 112-126.

[2] Harvey Jake G. Opeña, John Paul T. Yusiong “Automated Tomato Maturity Grading Using Abc-trained Artificial Neural Networks”. Frontiers in Plant Science, 2017, pp 12-26.

[3] Vishal G. Thengane , Mohit B. Gawande, “Arduino Based Tomato Ripening Stages Monitoring System”. International Journal of Innovative Research in Science, Engineering and Technology, 2018,Vol. 7, Issue 1.

[4] Redmond R. Shamshiri, James W. Jones, Kelly R. Thorp , Desa Ahmad, Hasfalina Che Man, and Sima Taheri, “Review of optimum temperature, humidity, and vapour pressure deficit for microclimate evaluation and control in greenhouse cultivation of tomato: a review”. Int. Agrophys., vol. 32, no. 2, pp. 287-302, January 2018.

[5] Abhishek L and Rishi Barath B, “Automation in Agriculture Using IOT and Machine Learning”. International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 8, Issue 8, pp. 2278-3075, June 2019.

[6] K. Yosuke, I. Masahiro, O. Nobuhiro, and I. Hiroshi, “Development of ‘IoTOMATO’ that is an agriculture monitoring system by using chat bot”. FIT2017, proceeding of IPSJ and IEICE, vol. 4, pp 229-230.

[7] D. Halil, O. G. Ece, and K. Murvet, “Disease detection on the leaves of the tomato plants by using deep learning”. 6th International Conference on Agro-Geoinformatics, 2017.

[8] Nuttakarn Kitpo,Yosuke Kugai, Masahiro Inoue, Taketoshi Yokemura and Shinichi Satomura, “Internet of Things for Greenhouse Monitoring System Using Deep Learning and Bot Notification Services”. IEEE International Conference on Consumer Electronics (ICCE), March 2019.