*International Journal of Engineering &Technology*, 7 (4) (2018) 6917-6920

**International Journal of Engineering & Technology**

*Website: www.sciencepubco.com/index.php/IJET*
*doi: 10.14419/ijet.v7i4.20246*
***Research paper***

# Development of tomato harvester robotic arm prototype using IoT application

**N. Yusof [1, 4] *, A. J. Ja'afar [4], N. MD Yunos [2, 4], S. Razali [3, 4]**

[1] *Advanced Manufacturing Centre (AMC)*
[2] *Optimization Modelling Analytic and Simulation (OPTIMAS)*
[3] *Center for Robotics and Industrial Automation (CeRIA)*
[4] *Fakulti Teknologi Maklumat & Komunikasi, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia*
**Corresponding author E-mail: norzihani@utem.edu.my*

## Abstract

In order to replace harvesting tomato manually, a robotic arm prototype for harvesting tomatoes using Internet of things (IoT) is presented in this paper. The objective to develop a robotic arm prototype which works autonomously is achieved. Due to the lack and high cost of labour, acrylic plastic materials prototype robot that controlled by Rasberry Pi3 is used for tomato recognition and localization. A coordinate or location used to pick tomatoes is identified. The usage of hue-saturation-value (HSV) colour could improve the harvesting efficiency. The movement of the actuator or the robotic arm from the initial position to the target's position is adopted to increase the accuracy of the targeted tomato. Concerning the control software, a graphic user interface is designed to submit the operator's commands and display the output. The performances of the robotic arm are shown.

***Keywords***: *IoT Application; Prototype; Raspberry Pi3; Robotic arm; Tomato harvester.*

## 1. Introduction

The idea of robotic harvesting was firstly proposed in 1960s [1] for citrus harvesting. Nowadays the harvesting robots were used in many farms to pick cucumber [2], citrus [3], apple [4], peach [5] and strawberry [6]. The tomato harvester robot is a robot that developed to harvest tomato fruit autonomously. This robot used arm mechanism to harvest tomato fruits. This arm mechanism provided the robot's ability to harvest the fruit like a human hand.
The tomato harvesting process needed many labour intensive and required multiple pickings. For the large-scale farming, the labour shortage was the main problem. This problem would cause the loss due to the post-harvest. In order to fulfil the local labour shortage, the robotic system for fruit picking could be implemented on the farm.
A variety approaches have long been investigated for application in fruit recognition. An olive recognition method using neural networks was presented by Gatica et al. [7]. The process of fruit recognition comprised of two stages: the first stage focused on deciding whether or not the candidate identified in the image corresponds to an olive fruit, the second stage focused on olives overlapping within the tree canopy. A statistical classifier, an ANN and a SVM classifier were built and used for detecting peach fruit by Kurtulmus et al. [5]. Authors reported that 84.6%, 77.9% and 71.2% of the actual fruits were successfully detected, using the three classifiers for the same validation set.
Fruit or vegetable detection for harvesting robot is conducted by various visual sensors. Xiang et al. [8] also studied a clustered tomato recognition method based on depth map acquired by a binocular stereo camera. The recognition accuracy of clustered tomatoes was 87.9% at an acquisition distance of 300–500 mm. Si et al. [9] used a stereo camera to detect and locate mature apples in tree canopies. The author reported that over 89.5% of apples were successfully recognized and the errors were less than 20 mm when the measuring distance was between 400 mm and 1500 mm.
By providing the learning ability, the machine learning algorithm was implemented. This algorithm enabled the robot to learn from the input data. So the farmer could teach the robot. Image processing method also implemented in this robot. This method provided the robot's vision. It was also used to differentiate between the raw and ripe tomatoes based on the fruits' colour. The engineering design loop was used to develop this project.
Most of the large farms in Malaysia used manual or hand harvesting method. The problems with the current method were very short local labour and required multiple pickings. Sometimes some tomatoes were lost during the post-harvest. It occurred when the local labour shortage took place during the harvesting season.
To overcome this problem, farmers should hire the foreign people as workers during harvesting season. They also must be trained before performing the harvesting process. The farmer had usually hired the professional trainers that could teach the workers about the ripeness stages and colour of the tomatoes that were ready to be harvested. If the farmers did not provide the trainers, the workers tended to harvest not fully ripe or unripe tomatoes. This type of tomatoes could not be sold to the fresh market or exported because they did not meet the quality standard. This problem could increase the payment that should be paid to the tomato farm labour. The current harvesting method also could cause defection to the tomato plant due to the wrong method of harvesting by hand.
This research developed a tomato harvester robot arm prototype using acrylic plastic and RasberryPi3 as the controller. The acrylic plastics were used as main material for the prototype's body part

because these materials were cheap, light weight and easy to cut down based on the robot's design. The Raspberry Pi3 Model B was used as the main controller or processor for the robot because it could compile the python languages and image processing. The image processing method was used as the main function for the robot to detect ripe tomatoes based on HSV red colour. The main criteria that was used to evaluate this prototype was the accuracy of the end effector of gripper part from the starting position to the targeted position.

Raspberry Pi (RPi) is a credit-sized mini-computer with great capabilities similar to a PC. The allure of the Raspberry Pi comes from a combination of the computer's small size and affordable price. Raspberry Pi can be used to develop a biometric architecture as it has provision of connecting with cameras, fingerprint scanners etc. via USB ports. It has an Ethernet port for Internet connectivity or can be connected to a Wi-Fi hotspot via USB Wi-Fi adapters [10].

Internet of Things (IoT) is one of the most encouraging technologies in the current era. A wide range of industrial IoT applications [11–13] have been developed and deployed in different domains such as transportation, agriculture, energy, healthcare, food processing industry, military, environmental monitoring, or security surveillance.

Through the application of IoT, the farmer could activate the robot at anytime and anywhere. The current captured image would be matched with the data stored in the database through IoT.

## 2. Methodology

The overall architecture of the developed system is presented in Figure 1, which consisted of three phases, namely input phase, process phase and output phase. The input phase, consisted of image processing and camera as a sensor. Image processing contained of an information of how tomato could be harvested, while the sensor captured an image of tomato from tomato plant.

In the process phase, a Raspberry Pi3 was used as an operating system, and it was connected to a computer for setting up all the programs. From here all the information of the image processing were stored, the process of image recognizing, as well as the decision of tomato to be harvested.

As for the output phase, once the system decided to harvest the tomato, the system would read the coordinate of the tomato to be harvested and hence the harvesting started.

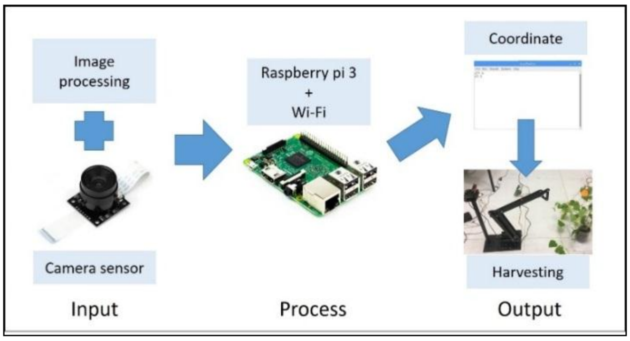


**Fig. 1:** The Overall Project Architecture.

Flowchart of designing a tomato harvester system was shown in Figure 2. Basically, the system would start by reading the information of tomato to be harvested, which was known as the image processing. When there was a tomato image captured by camera, it would recognize the image and compare the information with the information in image processing. If there was a similarity in both images, the system would decide to harvest and get the coordinate of the tomato to be harvested and hence the harvesting started, but if there was no similarity, the system would decide not to harvest it yet.

Next, the system would check if there was a tomato left in the plant. If the answer was yes, the system would continue through the same process which was comparing the captured image with the information in image processing, but if the answer was no, the system would end.

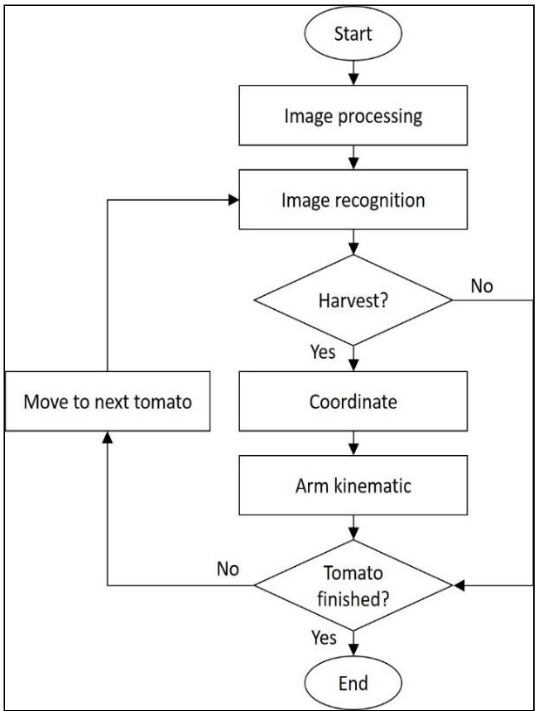


**Fig. 2:** A Flowchart of the Developed System.

## 3. Results and discussion

For testing this system, a tomato harvester prototype was developed, as shown in Figure 3. Using this prototype, the accuracy of the system would be tested. To test the accuracy of the system, each module in the system was checked in order to work out successfully. There were four modules, namely image processing module, image recognizing module, database module and arm kinematic module.

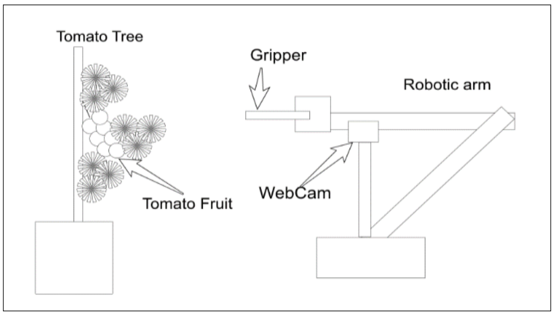


**Fig. 3:** Tomato Harvester Robotic Arm Prototype Layout.

The function of the image processing module was to receive the information on which type of tomato could be harvested. Next, once the information had been received, the image recognizing module would categorize the tomato based on the previous module as part of the training process in machine learning algorithm. If the information retrieved in both modules were similar, database module would read a minimum and maximum point of coordinate X and Y, and gave coordinate of the targeted tomato to be harvested. Lastly, arm kinematic module would move to the given coordinate and hence harvesting started.

As a result, the output of the tested module is shown in Table 1.

**Table 1:** Result of Tested Module

| Test Module | Result (Function/Not function) |
|---|---|
| Image processing module | Function |
| Image recognizing module | Function |

| | |
|---|---|
| Database module | Function |
| Arm kinematic module | Function |

As for details, the image detected from image processing module is shown in Figure 4, while one of the processes of image recognizing module is shown in Figure 5. The image was detected based on the hue-saturation-value (HSV) colour. The tomato fruit found would be displayed with crosshair on it.


**Fig. 4:** Image Processing Module.

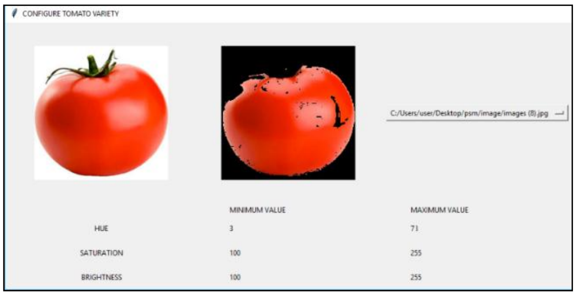


**Fig. 5:** One of the Process of Training in Image Recognizing Module.

Figure 6 shows a temporary file that stores minimum and maximum point coordinate while Figure 7 shows a targeted coordinate of tomato to be harvested. All these values would be used to find the correct distance of the targeted image.

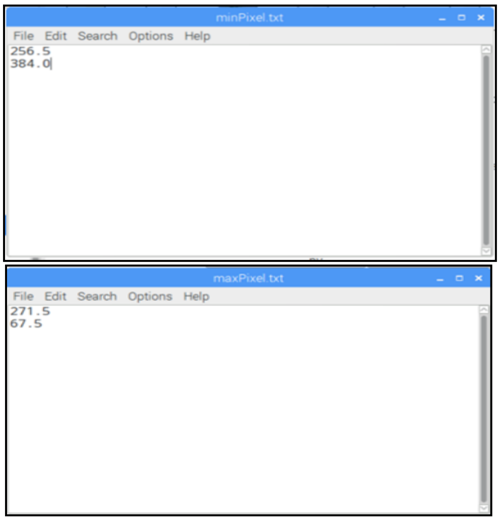


**Fig. 6:** Temporary File That Stores Minimum and Maximum Point Coordinate.

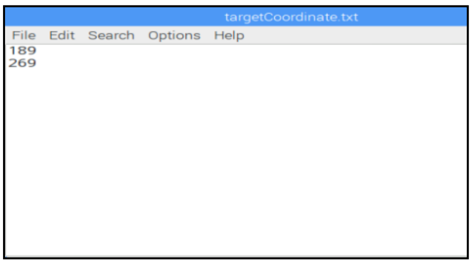


**Fig. 7:** Targeted Coordinate of Tomato to Be Harvested.

In Figure 8 displays the implementation of the pseudocode in the python script. In order to view overall process that happens during execution of the python script, Graphical User Interface (GUI) was used. Figure 9 shows arm kinematic of tomato harvester prototype. It is used as a hardware to harvest the tomato fruit. It would move to the targeted tomato based on the coordinate obtained.

```
################## CALCULATION PROCESS #################
pixelTargetY=minArmPixelY-targetCoordinateY
pixelDiff=minArmPixelY-maxArmPixelY
angleDiff=angleMin-angleMax
targetRealPixelDiff=minArmPixelY-targetCoordinateY
print("ANGLE DIFF: ",angleDiff)
print("PIXEL DIFF: ",pixelDiff)
maskTargetAngle=(angleDiff*targetRealPixelDiff)/pixelDiff
print("MASK TARGET: ",maskTargetAngle)
realAngle=angleMin-maskTargetAngle
print("TARGET ANGLE: ",realAngle)
servoConfig.setArmAngle(realAngle,2)
realY=recalibrate.recalibrateTarget()
print("REAL Y: ",realY)
print("Y Target coordinate: ",targetCoordinateY)
```

**Fig. 8:** Implementation the Pseudocode in the Python Script.

**Fig. 9:** Arm Kinematic of Tomato Harvester Prototype.

## 4. Conclusion

As a conclusion, the tomato harvester robotic arm prototype gave a lot of benefits which could help the users to determine the appropriate tomato that could be harvested. This could ensure the good harvest base on quality standard. It could also reduce the cost for paying human labour during the harvesting process. Besides, this robotic arm prototype could work well and it could be extended into a real robotic design.

## Acknowledgement

The authors would like to gratefully acknowledge the assistance, support and funding made available by the Universiti Teknikal Malaysia Melaka (UTeM), Faculty of Information and Communication Technology (FTMK), Centre for Research and Innovation Management (CRIM), Advanced Manufacturing Centre (AMC), Optimization Modelling Analytic and Simulation (OPTIMAS) and Center for Robotics and Industrial Automation (CeRIA).

## References

[1] C. E. Schertz and G.K. Brown, "Basic considerations in mechanizing citrus harvest", *Trans. ASAE*, Vol. 11, No. 2, (1968), pp.343–346. https://doi.org/10.13031/2013.39405.

[2] C. Ji, J. Zhang, T. Yuan and W. Li, "Research on key technology of truss tomato harvesting robot in greenhouse", *Applied Mechanics and Materials*, Vol. 442, (2014), pp.480–486. https://doi.org/10.4028/www.scientific.net/AMM.442.480.

[3] S. S. Mehta, W. MacKunis and T. F. Burks, "Nonlinear robust visual servo control for robotic citrus harvesting", *The 19th world congress of the International Federation of Automatic. Control,*

*Cape Town South Africa*, (2014), pp.8110–8115. https://doi.org/10.3182/20140824-6-ZA-1003.02729.

[4] Y. S. Si, G. Liu and J. Feng, "Location of apples in trees using stereoscopic vision", *Comput. Electr. Agric.*, Vol. 112, No. 3, (2015), pp.68–74. https://doi.org/10.1016/j.compag.2015.01.010.

[5] F. Kurtulmus, W.S. Lee and A. Vardar, "Immature peach detection in colour images acquired in natural illumination conditions using statistic classifiers and neural network". *Precis. Agric.*, Vol. 15, (2014), pp.57–79. https://doi.org/10.1007/s11119-013-9323-8.

[6] X. Q. Wei, J. Kun and H.L. Jin, "Automatic method of fruit object extraction under complex agricultural background for vision system of fruit picking robot", *Optics*, Vol. 125, No. 12, (2014), pp.5684–5689. https://doi.org/10.1016/j.ijleo.2014.07.001.

[7] G. Gatica, S. Best, J. Ceroni and G. Lefranc, "Olive fruits recognition using neural networks", *Procedia Comput. Sci.*, Vol. 17, (2013), pp.412–419. https://doi.org/10.1016/j.procs.2013.05.053.

[8] R. Xiang, H.Y. Jiang and Y.B. Ying, "Recognition of clustered tomatoes based on binocular stereo vision", *Comput. Electr. Agric.*, Vol. 106, No. 8, (2014), pp.75–90. https://doi.org/10.1016/j.compag.2014.05.006.

[9] Y.S. Si, G. Liu and J. Feng, "Location of apples in trees using stereoscopic vision", *Comput. Electr. Agric.*, Vol. 112, No. 3, (2015), pp.68–74. https://doi.org/10.1016/j.compag.2015.01.010.

[10] S. Dhvani and B. Vinayak, "IoT based Biometrics Implementation on Raspberry Pi", *Procedia Computer Science*, Vol. 79, (2016), pp.328 – 336. https://doi.org/10.1016/j.procs.2016.03.043.

[11] D. Xu, W. H. Li and S. Li, "Internet of things in industries: A survey", *IEEE Trans. Ind. Inf.*, Vol. 10, No. 4, (2014), pp.2233–2243. https://doi.org/10.1109/TII.2014.2300753.

[12] S. Li, T. Tryfonas and H. Li, "The internet of things: a security point of view", *Internet Res.*, Vol. 26, No. 2, (2016), pp.337–359. https://doi.org/10.1108/IntR-07-2014-0173.

[13] Y.I. Yuehong, Y. Zeng, X. Chen and Y. Fan, "The internet of things in healthcare: An overview", *J. Ind. Inf. Integr.*, Vol. 31, no. 1, (2016), pp.3–13. https://doi.org/10.1016/j.jii.2016.03.004.