https://doi.org/10.20965/jrm.2020.p0437

Paper:

Cutting Point Detection Using a Robot with Point Clouds for Tomato Harvesting

Takeshi Yoshida*, Takanori Fukao*, and Takaomi Hasegawa**

*Ritsumeikan University
1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, Japan
E-mail: {yoshi-da, tfukao}@fc.ritsumei.ac.jp
**Denso Corporation
1-1 Syowa-cho, Kariya, Aichi 448-8661, Japan
E-mail: takaomi_hasegawa@denso.co.jp
[Received May 28, 2019; accepted January 7, 2020]

This paper proposes a method to detect cutting points on tomato peduncles using a harvesting robot. The main objective of this study was to develop automated harvesting robots. The harvesting robot was equipped with an RGB-D (Red, Blue, Green, and Depth) camera to detect peduncles and an end effector to harvest tomatoes. Robots must be able to detect where to cut crops during harvesting. The proposed method was used to detect the cutting points on peduncles using a point cloud captured by the RGB-D camera. Our robot was used to identify the cutting points on target tomato peduncles at an actual farm to demonstrate the effectiveness of our approach experimentally. Using the proposed method, the harvesting robot could detect the cutting points on tomatoes.

Keywords: harvesting robot, cutting point detection, point cloud processing, voxel processing

1. Introduction

Greenhouse cultivation has spread globally because it has the advantage of enabling a stable supply throughout the year without the influence of weather. However, there is considerable room for automation and efficiency improvement in greenhouse cultivation. Further technical innovation is needed to equalize labor costs and the required labor.

There are busy periods that require more workers than usual in agricultural work. However, this may lead to excess employment during other periods if farmers hire workers based on their appropriate number for busy periods. On the other hand, it may be difficult for farmers to harvest their crops at the proper time if they do not hire enough workers. Therefore, our study focused on the development of a harvesting robot that could be used as a temporary worker.

Various studies have focused on the use of robots in agriculture. These studies can be classified into three categories.

The first category mainly deals with end effectors and grippers. Bachche et al. proposed the design and modeling of a gripper and a cutting system for a five degrees-of-freedom (5-DOF) robotic arm to harvest sweet peppers in horticultural greenhouses [1]. In the proposed design, both the gripping and cutting operations were performed using only one servo motor to avoid complications in the system. Yaguchi et al. proposed the design and development of an autonomous tomato harvesting robot equipped with a rotational plucking gripper, which dealt with estimation errors robustly [2]. Van Henten et al. proposed the concept of a modular cucumber harvesting robot equipped with a thermal cutting device and tested it in a greenhouse [3–5].

The second category mainly deals with robot vision. Chen et al. proposed a harvesting humanoid robot system to pick tomatoes and a vision cognition approach that enables the robot to harvest tomatoes [6]. Tokunaga et al. proposed an algorithm and designed an integrated circuit for the recognition of circular patterns in a binary image based on template matching with a modified matching degree. Their proposed system is a part of a watermelon harvesting robot vision system [7]. Si et al. proposed the use of a mechanical vision system to design a robot that can automatically recognize and locate apples for harvesting [8]. Monta et al. proposed a robotic vision method using cameras, a 3D vision system, and a laser sensor for a tomato harvesting robot and a cucumber harvesting robot [9]. Fujinaga et al. proposed a method to generate a tomato growth state map with image mosaicking for automatic harvesting [10]. Fukui et al. proposed a robot that estimates tomato fruit volume to acquire automatically the growth data of not only red mature tomatoes but also green immature tomatoes [11]. Sa et al. proposed a method to detect the peduncles from 3D models reconstructed from the detection of sweet peppers using a robot equipped with a robotic arm and an RGB-D camera [12]. Luo et al. proposed a method to detect the peduncles of grapes using one side of 2D images captured by a stereo camera [13].

The third category mainly deals with an integrated robotic system. Irie et al. proposed an asparagus harvest-

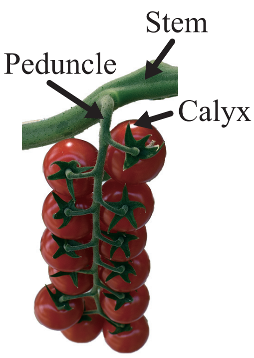


Fig. 1. Main parts of a target tomato used in this study.

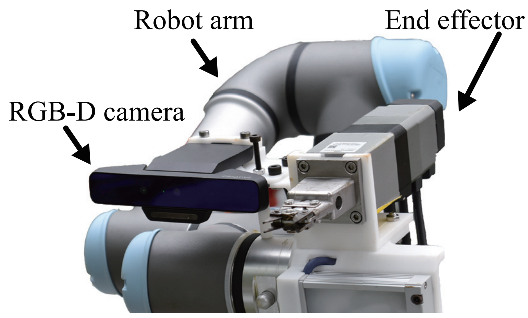


Fig. 2. Proposed harvesting robot.

Fig. 3. Representative field used for the experiment.

ing robot that measured whether the asparagus was tall enough to harvest using a 3D sensor [14, 15]. They also proposed a robotic arm mechanism and an end effector to grasp and cut asparagus. Kondo et al. proposed components, manipulators, end effectors, and visual sensors for fruit harvesting robots adapted for use with tomatoes, cherry tomatoes, strawberries, and grapes [16]. Hayashi et al. proposed a strawberry-harvesting robot consisting of a cylindrical manipulator, end effector, machine vision unit, storage unit, and traveling unit [17]. Arima et al. proposed a cucumber harvesting robot using a visual sensor, manipulator, end effector, and traveling device [18]. Wang et al. proposed a tomato harvesting robot consisting of a four-wheel independent steering system, 5-DOF harvesting system, laser navigation system, and binocular stereo vision system [19]. The harvesting robot was designed for harvesting tomatoes in a greenhouse.

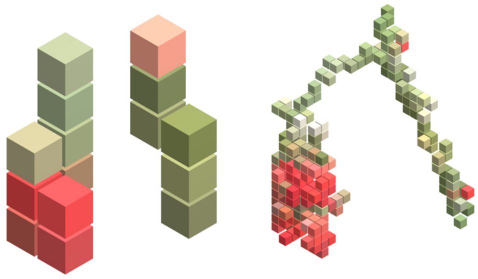
It is necessary for robots to detect where crops need to be cut during harvesting. Therefore, our study focused on detecting the cutting points on tomato peduncles to improve autonomous harvesting. Thus, this study focused on the above second category. We previously proposed a method to detect long peduncles with a 3D point cloud acquired with an RGB-D camera [20]. Voxels converted from point clouds were divided into layers, which were then treated as targets for evaluation. An energy function was defined based on the three conditions of a peduncle, and it was minimized to identify the cutting point on each of the peduncles. When a peduncle is short, there are cases where the calyx of the tomato obstructs the detection of the cutting point or the cutting point is at approximately the same height as the tomato fruit. Therefore, our previous method could not detect the correct cutting points in these cases. In addition, this method presupposes that the peduncles are vertically in front of the stems. Therefore, detecting peduncles irrespective of their length and angles is essential.

We propose a more advanced method to detect cutting points on peduncles with 3D point clouds containing depth information without depending on the state of the peduncles. **Fig. 1** shows the main parts of a target tomato used in this study. Each tomato is harvested in a bunch by cutting a peduncle. The proposed method uses only one frame of the point cloud data acquired with the RGB-D camera in contrast to the method proposed by Sa et al., which reconstructs dense point clouds of sweet peppers from multiple views. Voxelization was applied to reduce the amount of data. Consequently, the computational time for calculating the position between voxels was reduced. The proposed method constructs a directed acyclic graph after voxel clustering with several types of Region Growing methods. Finally, the Mahalanobis distance, which is defined based on statistic information, was used to detect appropriate cutting points on the peduncles.

2. Harvesting Robot

Figure 2 shows the proposed tomato harvesting robot. This harvesting robot has a 6-DOF robot arm, end effector, and RGB-D camera. The end effector has the functions of cutting tomato peduncles and holding tomatoes. Our proposed method does not consider the volume of the end effector because our method inserts the center of a pair of scissors of the end effector into the cutting point straightly at harvest time. The robot recognizes targets for harvesting using colored point clouds acquired with the RGB-D camera.

Figure 3 shows an example of the representative fields used for the experiment. A rail is placed between lines of tomatoes. The robot moves on the rail while facing one side of a row.



(a) Sparse resolution (b) Dense resolution

Fig. 4. Two different voxel resolutions.

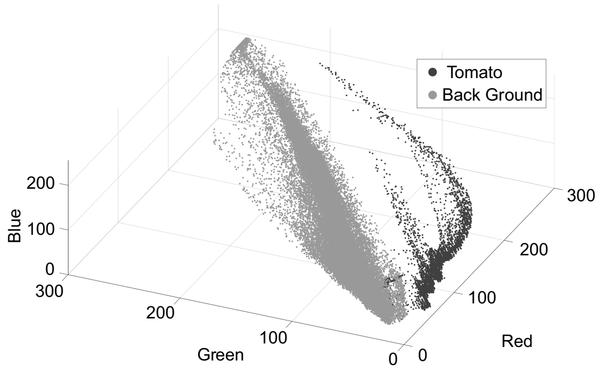


Fig. 5. RGB data used for learning with the SVM.

3. Algorithm for Cutting Points Detection

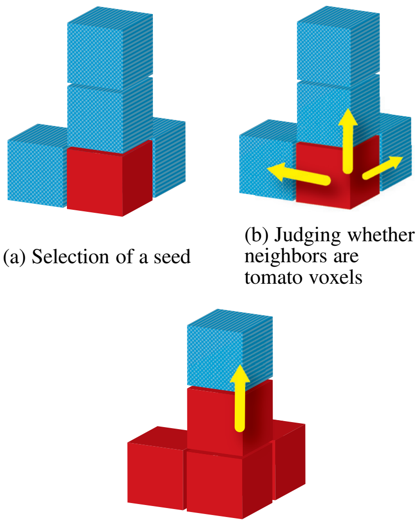
This section describes the cutting points detection. The proposed method contains five steps: voxelization of point clouds, clustering of the tomato regions, expanding the tomato voxels, constructing a directed acyclic graph, and identifying the cutting points.

3.1. Voxelization of Point Clouds

This section describes the voxelization of point clouds acquired with the RGB-D camera. Voxels have three pieces of information. The first piece of information is the coordinates of the voxels. The second is which point clouds are included in the voxels. The third is the adjacency information among voxels. Based on this information, voxelization not only has the effect of reduction of the entire data but also contributes to reducing the calculation time required for searching neighbor voxels. Our method uses voxelization with two resolutions to accelerate the process (**Fig. 4**). Sparse resolution voxels were used to cluster the tomato regions. Dense resolution voxels were used to identify cutting points. The sparse resolution is determined as the resolution that is close to the size of the tomato fruit and is sufficiently large to keep the tomato fruit from disappearing. The dense resolution is determined as the resolution that is close to the size of the peduncle of the tomato and is sufficiently large to keep the peduncle of the tomato from disappearing.

3.2. Clustering of Tomato Regions

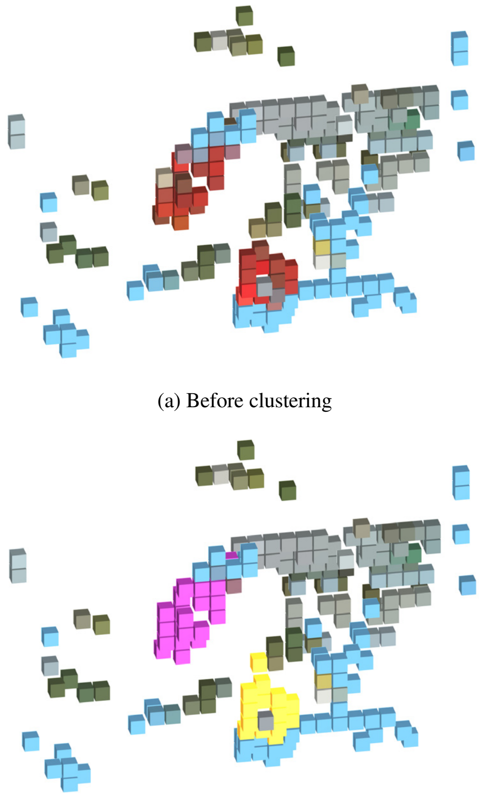
Clustering of tomato regions extracts each of the tomato voxels from the entire voxels and integrates the adjacent tomato voxels. First, the entire voxels are divided into voxels having the color of tomatoes and the other voxels. It is easy to divide voxels with a support vector machine (SVM) because there are not many objects whose color is close to that of tomatoes in farms. The SVM learns the hyperplane using the RGB data extracted from tomato images and background images. **Fig. 5** shows a graph in which the three coordinates represent the RGB data of the images. The dark gray points in this graph indicate the RGB data of the tomatoes, and the light gray points in this graph indicate the RGB data of the background. The RGB data of the tomatoes can be separated from the RGB data of the background easily because this graph shows that the RGB data of the tomatoes almost do not mix with the RGB data of the background. If not all the tomato voxels can be extracted in this step, the following step attempts to integrate the rest. Then, each of the tomato regions is extracted from the set of voxels having the color of tomatoes using Region Growing based on the adjacency information of voxels. First, a voxel is randomly selected as a seed, which indicates a starting point for Region Growing. Next, voxels adjacent to the seed are judged on whether the voxels are tomato voxels or not. If the voxels are determined to be tomato voxels, they are integrated into a tomato region. Finally, the voxels integrated into the tomato region are treated as new seeds. This processing is continued until no voxels can be selected as a seed. **Fig. 6** shows the transfer of seeds. **Fig. 7(b)** shows the result obtained when the voxels of **Fig. 7(a)** are extracted by clustering. **Fig. 7(b)** shows the selection of two tomato regions.



(a) Selection of a seed (b) Judging whether neighbors are tomato voxels

(c) New seed actions following the same rules

Fig. 6. Transition of seeds for Region Growing.



(a) Before clustering

(b) After clustering

Fig. 7. Clustering of tomato regions.

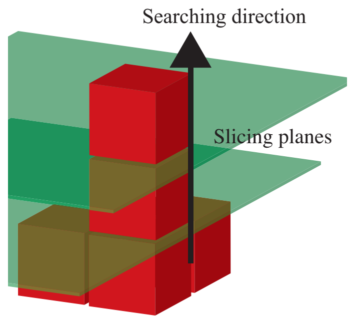
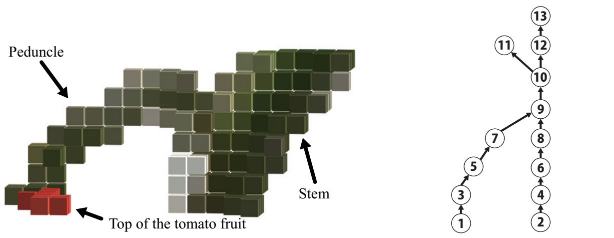


Fig. 8. Relation between the searching direction and slicing planes.



(a) Voxels around a peduncle (b) Directed acyclic graph

Fig. 9. Construction of a directed acyclic graph from voxels around a peduncle.

3.3. Expanding Tomato Voxels

Sparse resolution voxels used for clustering reduced the calculation time because searching for all the voxels requires time. The following steps use dense resolution voxels that correspond to sparse voxels clustered as tomato regions. Voxels extracted with clustering, as described in the previous section, did not include the voxels corresponding to the peduncles but only included voxels corresponding to the tomato fruits. The voxels are expanded further to add voxels corresponding to peduncles with an alteration of Region Growing, as described in the previous section. This method does not expand tomato voxels themselves but expands the region selected as tomato voxels. In this part of the study, voxels grew toward only the vertical upper direction rather than growing toward the surrounding directions, as described in the previous section. Additionally, all the voxels adjacent to the seed are candidates for integration in Region Growing, as described in this section, which is in contrast to the previous section in which only tomato-colored voxels were considered candidates. Voxels of the peduncles and stems were added by beginning to search from each tomato region.

3.4. Constructing a Directed Acyclic Graph

The expanded voxels described in the previous section included voxels that corresponded to a tomato, peduncle, and stem. These voxels were divided into pieces to judge whether each piece was suitable for a cutting point. This study treats the relation between these pieces as a graph and these pieces as nodes for constructing the graph. The first decides a direction to search the cutting point. Then, voxels were sliced at each discrete plane that crossed the searching direction vertically. These slicing plane angles correspond to the angle of a pair of scissors of the end effector. The angles of the slicing planes and the searching direction change according to the change in the angle of the end effector. **Fig. 8** shows the relation between the searching direction and slicing planes. Then, each sliced voxel is divided into pieces that are composed only of neighboring voxels. Region Growing is applied to each sliced voxel to divide into the nodes of the graph.

Next, the node relationships were defined to construct the graph expressing the candidates for the cutting points. Adjacent node relationships were clarified by examining the adjacent voxel relationships included in the nodes. The clarified adjacent relationships were treated as paths on the graph. These paths have directions from the lowest nodes to the highest nodes along the searching direction. A directed acyclic graph was constructed from voxels around a peduncle using these processes. **Fig. 9** shows an example of voxels and a directed acyclic graph constructed using this approach.

3.5. Identifying Cutting Points

One node on the directed acyclic graph was identified as a node that includes a cutting point in this part of the

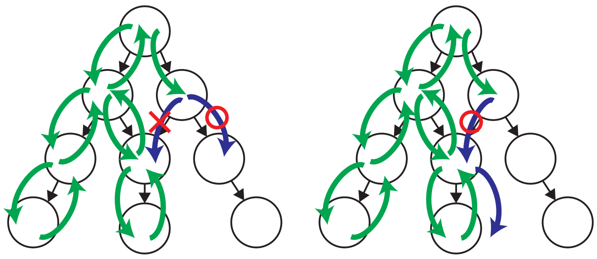


Fig. 10. Difference in search algorithms.

study. The first identifies all the reachability paths from the lowest nodes to the highest nodes. The longest paths were selected as paths that are possible candidates including the cutting point. This path is not adopted as a candidate even if this path may include the cutting point for this reason. If the end of the path did not reach the highest node, the end of that path probably reached the tip of a calyx of a tomato.

A backtracking algorithm was used to select the longest paths [21, 22]. The basic flow of the backtracking algorithm is close to the depth first search algorithm used in graph algorithms. The depth first search algorithm does not again search nodes that have already been searched to the end (**Fig. 10(a)**). However, the backtracking algorithm searches nodes again to the end from the other parent nodes even if the target nodes have already been searched to the end (**Fig. 10(b)**). The backtracking algorithm can acquire all the possible paths by continuing to record the passing nodes from the beginnings of the nodes to the end of the nodes. Two criteria are used to identify which node in the longest paths is suitable for the cutting point to harvest the tomatoes. One criterion is the similarity of the thickness of each of the nodes included in the selected path to the thickness of the peduncle. The thickness of each node is calculated from the width of the voxels included in the node. The direction of this width is parallel to the slicing plane. Another criterion is the similarity of the distance from the lowest node to each of the nodes to the half-length of the peduncle.

The Mahalanobis distance was used to handle these criteria without weightings. Eq. (1) expresses the Mahalanobis distance, where $w(n)$ is the thickness of each node, μ_w is the average thickness of the peduncles, σ_w is the standard deviation of the thicknesses of the peduncles, $d(n)$ is the distance from the center of a node to the top of a tomato fruit, μ_d is the average of this distance, and σ_d is the standard deviation of this distance.

$$D(n) = \frac{(w(n)-\mu_w)^2}{\sigma_w^2} + \frac{(d(n)-\mu_d)^2}{\sigma_d^2} \quad . \; . \; . \quad (1)$$

Averages and standard deviations used with the Mahalanobis distance were measured on an actual farm. The Mahalanobis distance was used to identify whether each node included in the selected path was the peduncle, the stem or others. If the pass did not continue from the peduncle node to the stem node, we concluded that for this target, it was difficult to clearly find the cutting point. This target was excluded from the harvesting candidates in this case.

The Mahalanobis distance has some local minimum values. The node of the first local minimum value is considered to be the node that includes the cutting point on the peduncle. This study determined the first local minimum value of the Mahalanobis distance by using it with hill climbing. First, some consecutive local values were selected randomly from all the values. The smallest value was determined from these local values. The first local minimum value was determined by repeating these processes several times.

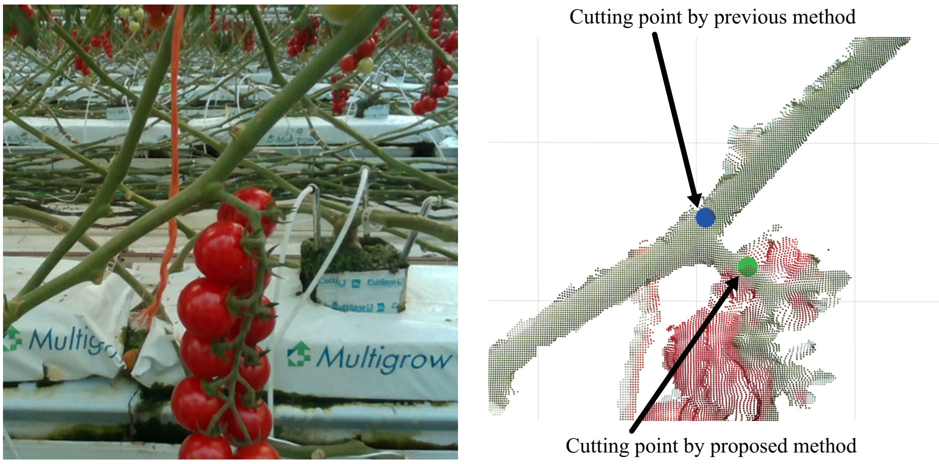
4. Experiments

Our harvesting robot detected the cutting points on tomato peduncles on an actual farm and confirmed our proposed method. Robot harvesting was performed using the following processes. First, the harvesting robot moved in front of a target tomato. Then, the robot observed the target tomato with the RGB-D camera. The RGB-D camera is set to be facing one side of a row of tomatoes. The image plane getting with the RGB-D camera is parallel to a row of tomatoes. The robot detected the cutting point of the peduncle of the target tomato. Finally, the robot inserted its end effector into the detected cutting point and harvested the target tomato without dropping the tomato. The calculation time with Intel Core i7-6820EQ (Quad core, 2.8 GHz) is approximately 1 s.

The searching directions in this experiment are vertical and horizontal to the robot to deal with various attitudes of the tomatoes. The horizontal direction is used when a peduncle is in landscape orientation. Directed acyclic graphs were constructed for the two searching directions. Two candidates for the cutting point were selected from the two directed acyclic graphs. The cutting point was decided by comparing which Mahalanobis distance from the candidates was lower.

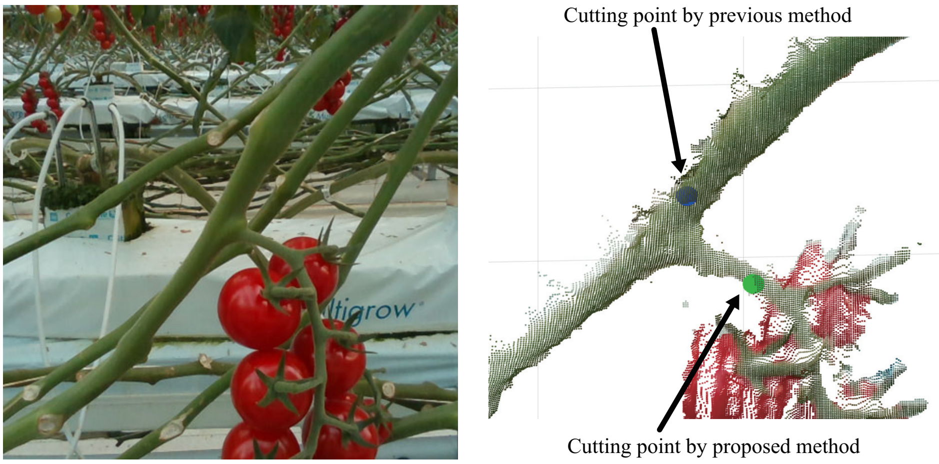
We attempted to detect the cutting points for 50 tomato samples. The stem and the peduncle were adjacent to each other in many samples. We categorized these samples into three groups based on the length of their peduncles.

Figures 11–13 indicate the detection of cutting points using our proposed method and our previous method when the peduncle was short, medium, and long. These figures show that our proposed method could detect cutting points even if the peduncles were short. However, all the cutting points detected by our previous method were on the stem. Our proposed method could detect cutting points on the peduncles as the candidate components were divided correctly using the directed acyclic graphs. **Fig. 13** also shows that our proposed method could detect the cutting point in the case when the angle of the peduncle was almost horizontal. This result showed that our proposed method of using two search directions to detect cutting points is effective.



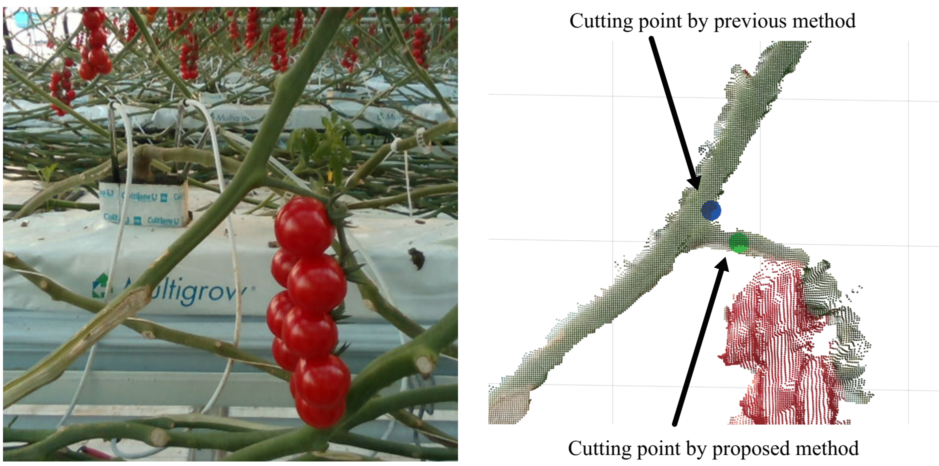
(a) Image of target (b) Point cloud and cutting points

Fig. 11. Result when the peduncle was short.



(a) Image of target (b) Point cloud and cutting points

Fig. 12. Result when the peduncle was medium in length.



(a) Image of target (b) Point cloud and cutting points

Fig. 13. Result when the peduncle was long.

Table 1. Detection results for different peduncle lengths.

	Short	Medium	Long
Length of peduncles [mm]	11–23	23–26	27–42
Number of samples	14	18	18
Proposed method	10	14	17
Previous method [20]	3	6	0

Table 1 shows the detection results for all the samples with our proposed method and our previous method. This table shows that the peduncle lengths, the number of peduncles, and the number of cutting points were correctly detected. These results show that our proposed method performed better than our previous method in all the cases. Our previous method could not divide candidate components correctly even if the peduncles were long because the stem and the peduncle were adjacent to each other in many samples. This result showed that our proposed method of using the directed acyclic graphs is effective.

5. Conclusions

This paper proposed a method for detecting the cutting points on tomato peduncles using a harvesting robot. In this approach, a directed acyclic graph was constructed with several types of Region Growing. The Mahalanobis distance, which is defined based on statistic information, was used to detect appropriate cutting points on the peduncles. The experimental results confirmed that cutting point detection accurately directed the harvesting robot to harvest tomatoes without depending on the state of the peduncles.

References:

[1] S. Bachche and K. Oka, "Design, Modeling and Performance Testing of End-Effector for Sweet Pepper Harvesting Robot Hand," J. Robot. Mechatron., Vol.25, No.4, pp. 705-717, 2013.

[2] H. Yaguchi, K. Nagahama, T. Hasegawa, and M. Inaba, "Development of an autonomous tomato harvesting robot with rotational plucking gripper," Proc. of 2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 652-657, 2016.

[3] E. J. Van Henten, J. Hemming, B. A. J. Van Tuijl, J. G. Kornet, J. Meuleman, J. Bontsema, and E. A. van Os, "An Autonomous Robot for Harvesting Cucumbers in Greenhouses," Autonomous Robots, Vol.13, No.3, pp. 241-258, 2002.

[4] E. J. Van Henten, B. A. J. Van Tuijl, J. Hemming, J. G. Kornet, J. Bontsema, and E. A. Van Os, "Field Test of an Autonomous Cucumber Picking Robot," Biosystems Engineering, Vol.86, No.3, pp. 305-313, 2003.

[5] E. J. Van Henten, B. A. J. Van Tuijl, G.-J. Hoogakker, M. J. Van Der Weerd, J. Hemming, J. G. Kornet, and J. Bontsema, "An Autonomous Robot for De-leafing Cucumber Plants grown in a High-wire Cultivation System," Biosystems Engineering, Vol.94, No.3, pp. 317-323, 2006.

[6] X. Chen, K. Chaudhary, Y. Tanaka, K. Nagahama, H. Yaguchi, K. Okada, and M. Inaba, "Reasoning-based vision recognition for agricultural humanoid robot toward tomato harvesting," Proc. of 2015 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 6487-6494, 2015.

[7] Y. Tokunaga, T. Hakukawa, and T. Inoue, "Algorithm and Design of an Intelligent Digital Integrated Circuit for a Watermelon Harvesting Robot," J. Robot. Mechatron., Vol.11, No.3, pp. 173-182, 1999.

[8] Y. Si, G. Liu, and J. Feng, "Location of apples in trees using stereoscopic vision," Computers and Electronics in Agriculture, Vol.112, pp. 68-74, 2015.

[9] M. Monta, N. Kondo, S. Arima, and K. Namba, "Robotic Vision for Bioproduction Systems," J. Robot. Mechatron., Vol.15, No.3, pp. 341-348, 2003.

[10] T. Fujinaga, S. Yasukawa, B. Li, and K. Ishii, "Image Mosaicing Using Multi-Modal Images for Generation of Tomato Growth State Map," J. Robot. Mechatron., Vol.30, No.2, pp. 187-197, 2018.

[11] R. Fukui, K. Kawae, and S. Warisawa, "Development of a Tomato Volume Estimating Robot that Autonomously Searches an Appropriate Measurement Position – Basic Feasibility Study Using a Tomato Bed Mock-Up –," J. Robot. Mechatron., Vol.30, No.2, pp. 173-179, 2018.

[12] I. Sa, C. Lehnert, A. English, C. McCool, F. Dayoub, B. Upcroft, and T. Perez, "Peduncle Detection of Sweet Pepper for Autonomous Crop Harvesting – Combined Color and 3-D Information," IEEE Robotics and Automation Letters, Vol.2, No.2, pp. 765-772, 2017.

[13] L. Luo, Y. Tang, X. Zou, M. Ye, W. Feng, and G. Li, "Vision-based extraction of spatial information in grape clusters for harvesting robots," Biosystems Engineering, Vol.151, pp. 90-104, 2016.

[14] N. Irie, N. Taguchi, T. Horie, and T. Ishimatsu, "Development of Asparagus Harvester Coordinated with 3-d Vision Sensor," J. Robot. Mechatron., Vol.21, No.5, pp. 583-589, 2009.

[15] N. Irie and N. Taguchi, "Asparagus Harvesting Robot," J. Robot. Mechatron., Vol.26, No.2, pp. 267-268, 2014.

[16] N. Kondo and M. Monta. "Fruit Harvesting Robotics," J. Robot. Mechatron., Vol.11, No.4, pp. 321-268, 1999.

[17] S. Hayashi, K. Shigematsu, S. Yamamoto, K. Kobayashi, Y. Kohno, J. Kamata, and M. Kurita, "Evaluation of a strawberry-harvesting robot in a field test," Biosystems Engineering, Vol.105, No.2, pp. 160-171, 2010.

[18] S. Arima and N. Kondo. "Cucumber Harvesting Robot and Plant Training System," J. Robot. Mechatron., Vol.11, No.3, pp. 208-212, 1999.

[19] L. L. Wang, B. Zhao, J. W. Fan, X. A. Hu, S. Wei, Y. S. Li, Q. B. Zhou, and C. F. Wei, "Development of a tomato harvesting robot used in greenhouse," Int. J. of Agricultural and Biological Engineering, Vol.10, No.4, pp. 140-149, 2017.

[20] T. Yoshida, T. Fukao, and T. Hasegawa, "Fast Detection of Tomato Peduncle Using Point Cloud with a Harvesting Robot," J. Robot. Mechatron., Vol.30, No.2, pp. 180-186, 2018.

[21] R. Stephens, "Essential Algorithms: A Practical Approach to Computer Algorithm," Wiley, 2013.

[22] R. Sedgewick, "Algorithms in C++ Part 5: Graph Algorithms (3rd edition)," Addison-Wesley Professional, 2001.

Name:
Takeshi Yoshida

Affiliation:
Associate Professor, Research Organization of Science and Technology, Ritsumeikan University

Address:
1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, Japan

Brief Biographical History:
2015- Assistant Professor, Aoyama Gakuin University
2017- Assistant Professor, Ritsumeikan University
2019- Associate Professor, Ritsumeikan University

Main Works:
- Robotics, computer vision

Membership in Academic Societies:
- The Robotics Society of Japan (RSJ)
- The Society of Instrument and Control Engineers (SICE)
- Information Processing Society of Japan (IPSJ)

Name:
Takanori Fukao

Affiliation:
Professor, Department of Electrical and Electronic Engineering, Ritsumeikan University

Address:
1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, Japan

Brief Biographical History:
1996- Assistant Professor, Kyoto University
2001- Visiting Scientist, Carnegie Mellon University
2004- Associate Professor, Kobe University
2015- Professor, Ritsumeikan University

Main Works:
• Robotics, autonomous vehicles

Membership in Academic Societies:
• The Japan Society of Mechanical Engineers (JSME)
• The Institute of Electrical and Electronics Engineers (IEEE)

Name:
Takaomi Hasegawa

Affiliation:
Project Assistant Manager, Robotics System Development Department, Denso Corporation

Address:
1-1 Showa-cho, Kariya, Aichi 448-8661, Japan

Brief Biographical History:
2006- Denso Corporation

Main Works:
• Robotics