Article

Automatic Tomato and Peduncle Location System Based on Computer Vision for Use in Robotized Harvesting

M. Benavides, M. Cantón-Garbín *, J. A. Sánchez-Molina and F. Rodríguez

Departamento de Informática, Escuela Superior de Ingeniería, Universidad de Almería, ceiA3, CIESOL, Ctra. Sacramento s/n, 04120 Almería, Spain; MARTABENAVIDES@terra.es (M.B.); jorgesanchez@ual.es (J.A.S.-M.); frrodrig@ual.es (F.R.)
* Correspondence: mcanton@ual.es

check for updates

Abstract: Protected agriculture is a field in which the use of automatic systems is a key factor. In fact, the automatic harvesting of delicate fruit has not yet been perfected. This issue has received a great deal of attention over the last forty years, although no commercial harvesting robots are available at present, mainly due to the complexity and variability of the working environments. In this work we developed a computer vision system (CVS) to automate the detection and localization of fruit in a tomato crop in a typical Mediterranean greenhouse. The tasks to be performed by the system are: (1) the detection of the ripe tomatoes, (2) the location of the ripe tomatoes in the XY coordinates of the image, and (3) the location of the ripe tomatoes' peduncles in the XY coordinates of the image. Tasks 1 and 2 were performed using a large set of digital image processing tools (enhancement, edge detection, segmentation, and the feature's description of the tomatoes). Task 3 was carried out using basic trigonometry and numerical and geometrical descriptors. The results are very promising for beef and cluster tomatoes, with the system being able to classify 80.8% and 87.5%, respectively, of fruit with visible peduncles as "collectible". The average processing time per image for visible ripe and harvested tomatoes was less than 30 ms.

Keywords: image processing; greenhouse; automatic tomato harvesting; precision agriculture

1. Introduction

Few crops in the world are in such a high demand as the tomato. It is the most widespread vegetable in the world and the one with the highest economic value. During the 2003–2017 period, world tomato production increased annually from 124 million tons to more than 177 million tons. In the last 15 years, consumption has experienced sustained growth of around 2.5% [1]. These data make the tomato one of the most important vegetables in terms of job creation and wealth, and its future looks every bit as positive. According to data from the FAO [1], even though tomatoes are grown in 169 countries (for both fresh consumption and industrial use), the 10 main producers in 2017 (of which Spain is in eighth place) accounted for 80.45% of the world total. These countries are: China, India, The United States, Turkey, Egypt, Italy, Iran, Spain, Brazil and Mexico. The European Union is the world's second largest tomato producer after China. In Almería (south-east Spain), where the largest concentration of greenhouses in the world is located (more than 30,000 hectares), the main crop is tomato, representing 37.7% of total production [2]. Based on the data on the overall labor distribution in tomato cultivation, between 25% and 40% of all labor is employed in the highly repetitive task of harvesting [3]. Traditionally, harvesting is done manually with low-cost mechanical aids (harvesting trolleys, cutting tools, etc.), so most of the expense corresponds to human labor.

Automation is essential in any production system that tries to be competitive. It reduces production costs and improves product quality [2,4–6]. Protected agriculture is a sector where the application of such techniques is required, particularly for the problem of the automatic harvesting of fruit (from trees) and vegetables. This is typical of the type of process that needs to be robotized because it is a repetitive pick-and-place task.

1.1. Literature Review

Over the last 40 years, a lot of research effort has been expended on developing harvesting robots for fruits and tomatoes [5–13]. Mavridou et al. [14] presented a review of machine vision techniques in agriculture-related tasks focusing on crop farming. In [15], Schillaci et al. attempted to solve the problem of recognizing mature greenhouse tomatoes using an SVM (support vector machine) classifier; however, the results of this work were not quantified. Ji et al. [16] achieved a success rate of 88.6% by using a segmentation feature of the color difference 2R–G–B and a threshold to detect the tomatoes, although an artificial tomato-clip was used to detect the peduncle. Feng et al. [17] used a CCD camera and an HSI color model for image segmentation. The 3D distance to the center of each segmented tomato was obtained using a laser. The success rate for harvesting the tomatoes and the execution time of a single harvest cycle (tomato location, movement of arm and picking) were 83.9% and 24 s, respectively. In [18], the images captured by a color camera were processed, extracting Haar-like features from sub-windows in each original image. After that, an Adaboost classifier followed by a color classifier managed to recognize 96% of the ripe tomatoes, although 10.8% were false negatives and 3.5% of the tomatoes were not detected. The same authors [19] used an adaptive threshold algorithm to obtain the optimal threshold. Subsequently, two images (a* and I) from the L*a*b space were obtained and fused by means of wavelets. Ninety per cent of the test target tomatoes were recognized in a set of 200 samples. Li et al. [20] used a region segmentation method followed by erosion and dilation to enhance the contour, and fuzzy control to determine the locus of the tomatoes. According to the authors, the recognition time was significantly reduced compared with other methods, but they did not give details regarding the error rates. In [21], a human operator marked the location of the tomatoes on the screen. After that, the position was obtained by a stereo camera. With this human–robot cooperation, a detection success rate of about 94% was achieved. Taqi et al. [22] mimicked a greenhouse and a very controlled environment where it was easy to detect the ripe tomatoes by means of the red color. In [23], Wang et al. used a binocular stereo vision system with the Otsu method to segment the ripe tomatoes. The success rate for ripe tomato recognition was 99.3% while the recognition and pitching time for each tomato was about 15 s with a success rate of 86%. Zhang et al. [24] used a convolutional neural network (CNN) as a classifier with a classification success rate of 92%. Kamilaris et al. [25] did a survey of deep learning in agriculture. In [26], the R–G plane was used to segment the tomato branch. Eighty-three percent of the mature test branches were harvested, but 1.4 attempts and 8 s were needed per branch. In Malik et al. [27], an HSV transform was used to detect only red tomatoes. To separate the connected tomatoes, a watershed algorithm was used. The rate of red tomatoes detected was about 81.6%. In [28], a dual arm with binocular vision and an Adaboost and color analysis classifier achieved a classification rate of 96%. In Lin et al. [29], a novel approach for recognizing different types of fruits (lemon, tomato, mango and pumpkins) was developed using the Hough transform to detect curved sub-fragments in images of real tomato environments. To remove false positive centers, an SVM was applied to the mixed contours. Depending on the type of fruit, the precision of this method varied between 0.75 and 0.92. Yuan et al. [30] proposed a method for cherry tomato detection based on a CNN for reducing the influence of illumination, growth difference and occlusion. Yoshida et al. [31] obtained 3D images of the bunch tomato crops and detected the position of the bunch peduncle. Only six sample images were used in this work and the computation time for each image was not specified. The authors achieved a precision rate of 98.85%.

1.2. Objectives

In this work, the detection and automatic location of the ripe fruit and their peduncles in the (x,y) plane was performed with one camera. This is because, later on, it will be necessary to indicate to the mechanical system in charge of the collection the exact place where the fruit should be separated from the plant. We considered the main novelty of the work to be the tomato peduncle detection.

To date, we have not seen this issue addressed in the reviewed literature. To achieve this, an exhaustive study was conducted into the different digital image processing techniques [32], applying those that provided the best results, then analyzing the problems arising and providing possible solutions. In our study, other techniques from the fields of pattern recognition or computer vision, such as deep learning, were not used because our goal was not to recognize or classify different tomato types.

As commented on in a previous work [33], when designing a harvesting robot, the morphology must be considered in order to work with irregular volumes. Two key factors should also be taken into account: (i) given that plants and trees can be located on large areas of land, the robots need to be mobile (they are usually the harvester-hybrid type, i.e., manipulator arms loaded onto platforms or mobile robots); (ii) for the fruit-picking operation, the robot must pick the fruit and separate it from the plant or tree, thus the end-effector design is fundamental. Once the harvesting robot has been designed, it must carry out the following phases to pick up the fruit or vegetables: (1) system guidance, (2) environment positioning, (3) fruit detection (4) fruit location, (5) approaching the robot end-effector to the fruit, (6) grasping the fruit, (7) separating the fruit, and (8) stacking or storing the harvested fruits. This paper focused on the automatic detection and location of ripe tomato fruit. As Figure 1 shows, the subsystem for locating the fruit must provide the position and orientation of the end-effector ($Tool) so that it coincides with the position and orientation of the different elements of each fruit to be harvested ($Penduncle and $Fruit) in the manipulator workspace. The peduncle and fruit are considered separately because there are different end-effectors—either for separating the fruit from the plant based on cutting the peduncle [34], or embracing/absorbing the fruit [35,36].
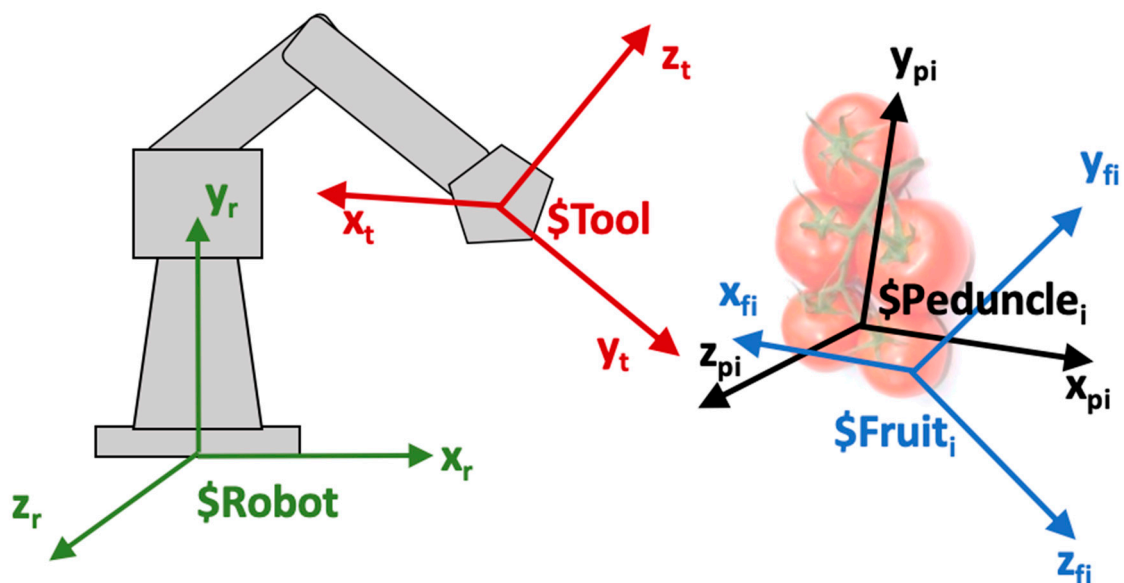

Figure 1. Fundamentals of robot harvesting.

An optimal solution to solve the position and orientation problems involves six degrees of freedom [37]: three for positioning in space (x, y, z) and three for orientation (pitch, roll and yaw), although certain hypotheses can be considered to simplify this. The first is to not consider the orientation problem because the end-effector can be designed by only knowing the position of the fruit elements; thus, one needs to know the coordinates (X, Y, Z) of the $Peduncle and $Fruit. The idea is

to combine a computer vision subsystem that provides the (x, y) coordinates with a laser mounted on a servo-based Pan-Tilt subsystem that points to the position calculated by the vision system to determine the z-coordinate of the tomato elements.

This work presents the beginning of the total automation—the automatic 2D detection and location of the ripe tomato fruits and their peduncles—as shown in Figure 2. For this, an exhaustive study was carried out employing the different computer vision and digital image processing [29] techniques, applying those that provided the best results.
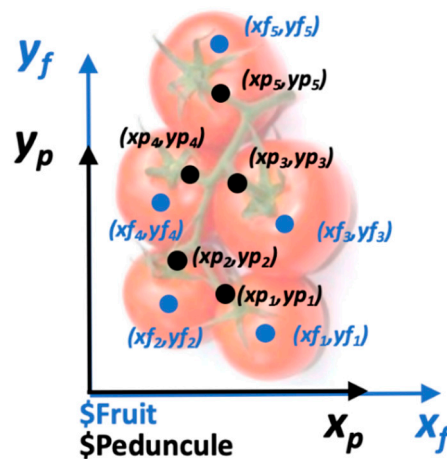

Figure 2. Automatic 2D detection and location of the ripe tomato elements.

As mentioned above, some tomato harvesting systems work by first pressing and then pulling on the tomatoes. In this work, a first step towards a tomato harvesting system is presented in detail, in which multiple digital image processing tools are used to obtain not only the position of the tomato but also that of its peduncle. This was applied to two types of crops, beef and cluster tomatoes, collecting the fruit by cutting the peduncle, not by pressing on the tomato, thus avoiding possible damage. These objectives were divided into a series of sub-objectives:

- Detection of the ripe tomatoes. From the image provided, the system must detect tomatoes that are ripe and segment them from the rest of the image.
- Location of the ripe tomatoes in XY. After recognizing the ripe tomatoes, the system should position them in the XY plane of the image.
- Location of the peduncle in XY. The system should provide the location of the peduncle of the ripe tomatoes in the XY plane of the image.

The main contribution of this work is: (1) the identification and location of the ripe tomatoes and their peduncles. (2) Every image is processed in less than 30 ms. (3) The system can be used for any end-effector based on cutting or suctioning the tomatoes. It is a very important contribution because this system can be used for any tomato harvesting robot, without having to develop a new vision system for each end-effector prototype.

This paper is organized as follows: in Section 2, the different materials and techniques used for the automatic detection and location of ripe tomato fruit and peduncles are described. In Section 3, the results of these processes are shown and discussed with regard to two tomato varieties: beef and cluster. Lastly, in Section 4, the main conclusions and future works are summarized.

2. Materials and Methods

2.1. Greenhouse Environment

The data used to develop the first version of the algorithm were acquired in the greenhouses of the Cajamar Foundation's Experimental Station in El Ejido, Almería Province, Spain (2°43′00″ W,

36°48′00″ N, and 151 m above sea level). The tomato crops were grown in a multi-span "Parral-type" greenhouse, with a surface area of 877 m² (37.8 × 23.2 m). The greenhouse orientation is east to west, whilst the crop rows are aligned north to south, with a double plant row separated by 1.5 m. The tomato crop was transplanted in August and the season finished in June (long season). This variety has indeterminate growth, the fruit ripens by height and position on the branch, so cultivation tasks are continuous throughout the season.

In this situation, tomato harvesting is carried out at least once a week from November to June. The growing conditions and crop management are very similar to those in commercial tomato greenhouses. The climate parameters inside the greenhouse are monitored continuously every 30 s. Outside the greenhouse, a weather station measures the air temperature and relative humidity (with a ventilated sensor), solar radiation and photosynthetic active radiation (with a silicon sensor) and precipitation (with a rain detector). It also records the CO_2 concentration and wind speed/direction.

During the experiments, the indoor climate variables were also recorded, especially the air temperature, relative humidity, global solar radiation, photosynthetic active radiation, soil and cover temperature, water and electricity consumption, an irrigation demand tray, water content, electrical conductivity and soil temperature.

2.2. Image Acquisition and Processing

A Genie Nano (C1280 + IRF 1 1280 × 1024) with an OPT-M26014MCN optics (6 mm. f1.4) was used to acquire the images in the real working environment (Figure 3). With these, a data set of images was built to test the system operation. The image acquisition for our application was simply achieved by reading the files using an image format, in our case, the ".jpg" format. The camera was located perpendicular to the greenhouse soil surface, at a 20–30 cm distance from the plant, its height depending on the harvesting phase. An external flash was used to enhance the tomatoes and to make their central area brighter than the rest.

Figure 3. Greenhouse work environment.

The computer used was a MacBook Pro (Intel i9, 2.33 GHz, 16 GB DDR4) running a Windows 10 operating system with Bootcamp. To build our system, the NI Vision Development Module from NI Labview 2015 was used.

2.3. Tomato Detection Algorithm

As can be observed in Figures 3 and 4, the mature tomatoes are usually located in the lower part of the plant, where there are practically no leaves.

Figure 4. Lower part of the crop with mature tomatoes.

The system performs a series of operations to detect those ripe tomatoes that are in the foreground (not occluded) and segment them from the rest of the image elements. At the end of this stage, each ripe tomato is represented by a single region. The flowchart of the operations performed to detect the ripe tomato is shown in Figure 5.
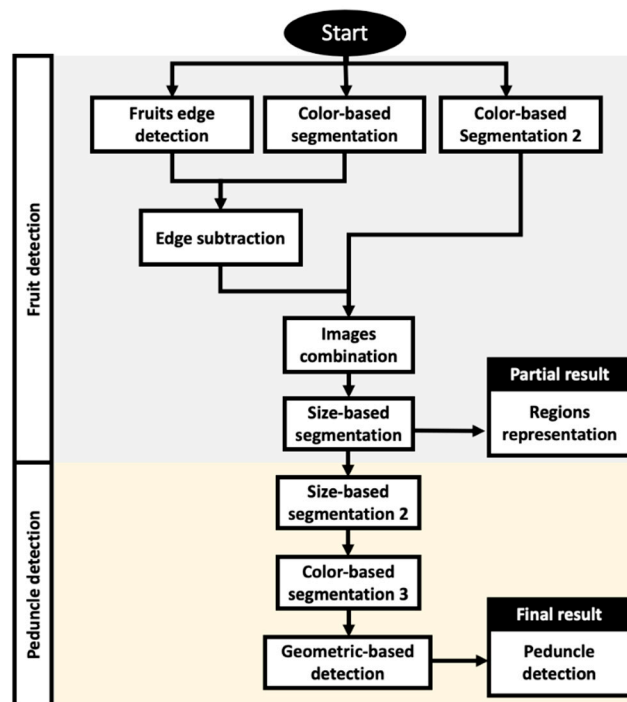

Figure 5. Flowchart of the ripe tomato detection stage.

During this stage, several operations were carried out simultaneously on different copies of the original image, chosen for its characteristics to show the results of each sequence:

- Tomato-Edge Detection

Figure 6a illustrates a typical situation in tomato greenhouses. First of all, the green container in the image measures the amount of drip irrigation water for the plants; this is usually present in many of the greenhouse corridors. As shown in Figures 3 and 4, the tomatoes begin to ripen at the bottom of the plant, where there are few leaves. Nevertheless, there are smaller leaves and tomatoes that have been removed by segmentation and other processes on the right and left sides. In addition, in greenhouse horticulture, the leaves are usually removed from the bottom (the standard cultivation

technique), so only conditions that are normal for greenhouses in this area are being reproduced in the article.
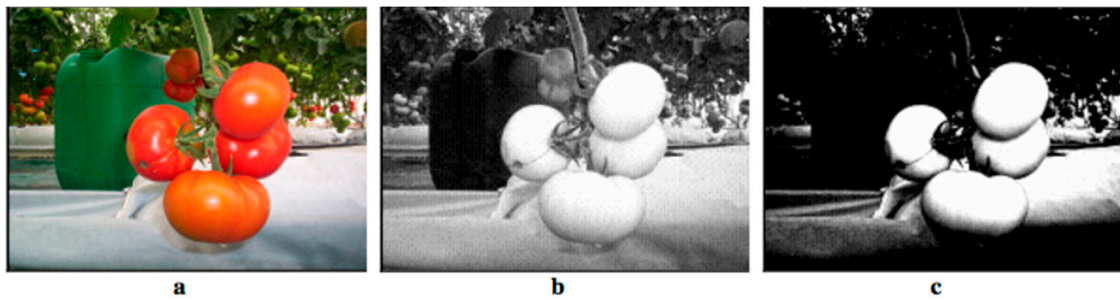

Figure 6. (**a**) Original image; (**b**) R plane; and (**c**) power transform and contrast enhancement.

First, we choose the R component of the RGB image (Figure 6b). To enhance the contrast, we apply a power law transform $s = c\, r^{\Upsilon}$, where r are the initial gray levels of the R image, c is constant (usually 1), Υ 1 lightens the image and Υ 1 darkens the image, and s are the final gray levels of the image after the contrast enhancement (Figures 6c and 7a). The parameters Υ and c vary depending on the type of tomato in the image because the color and reflectance are not the same for all types of tomatoes. The image's histogram analysis allows one to obtain sufficiently good values for Υ and c.
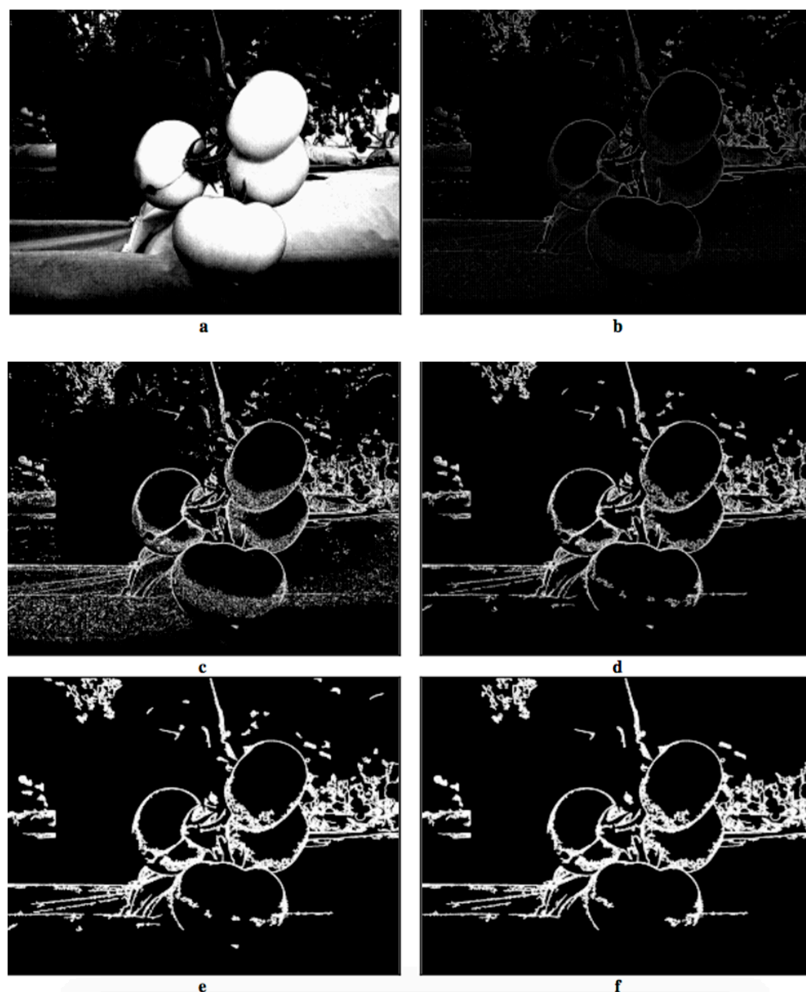

Figure 7. (**a**) Power transform and contrast enhancement; (**b**) Sobel operator; (**c**) gray-scale segmentation; (**d**) size segmentation; (**e**) dilation; and (**f**) size segmentation 2.

After increasing the contrast, the tomato-edge detection could be carried out with one of several operators (Sobel, Roberts, Prewitt, etc.); however, after conducting an exhaustive study applying the different types of operators, we decided to use Sobel because it provided a more precise positioning of the tomatoes and peduncles (Figure 7b).

The noise and the outline of the shadows that appear on the fruit surface make it difficult to capture their exact contour. To keep only what interests us, a series of operations were carried out on the image in Figure 7b. The first was a segmentation based on grayscale or intensity, which allows us to eliminate a large part of the image noise and the effects of the shadows (Figure 7c); this was followed by a segmentation based on size (regions of connected pixels that do not exceed a certain number are eliminated) (Figure 7d).

Following the previous functions, the morphological operation of dilation was applied (Figure 7e). The dilation objective is to be able to join all the dashed lines to form a contour without discontinuities, or at least, with many less than those presented at the beginning.

Finally, we again performed segmentation based on size (Figure 7f) to eliminate the elements that continue appearing on the fruit surface but that are not part of its contour.

- Image Binary Inversion

After detecting the fruit edge, the obtained image is still not ready to be used for the edge subtraction (Figure 5) since the binary image of the fruit contours is inverted (Figure 8a). In addition, segmentation based on size was carried out (Figure 8b) to eliminate the small regions that remained inside the contours, making them more defined.
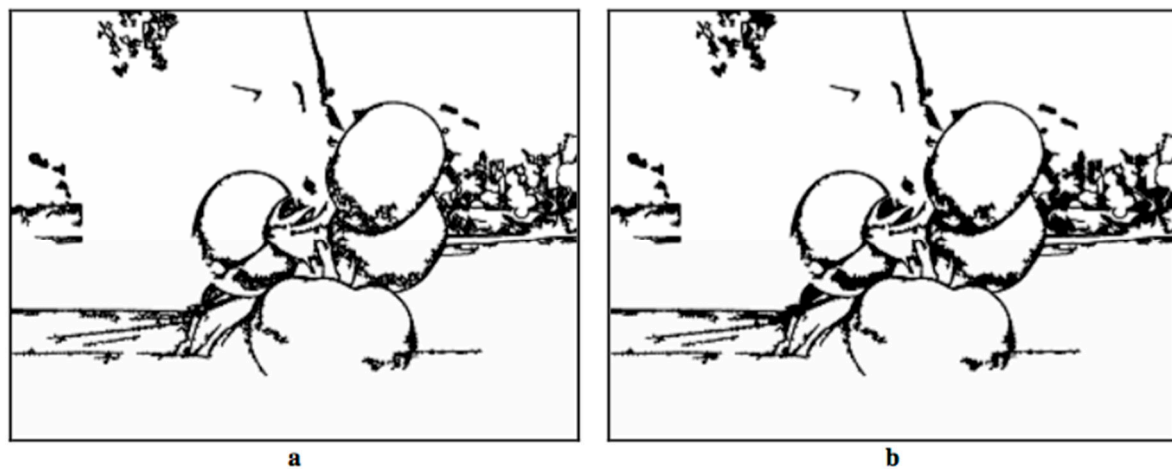

Figure 8. (**a**) Binary inversion; and (**b**) size segmentation.

- Segmentation based on color 1 (Figure 9) to obtain a separated region for each mature tomato that appears in the image.

Figure 9. (**a**) Original image; and (**b**) segmentation based on color 1 (the whole ripe surface is obtained).

- Edge subtraction (Figure 10): next, we applied the logical AND function on Figures 8b and 9b. The result was a new binary image where the region, or regions, representing the total ripe surface, appears divided into regions that already represent individual ripe tomatoes (Figure 10b).
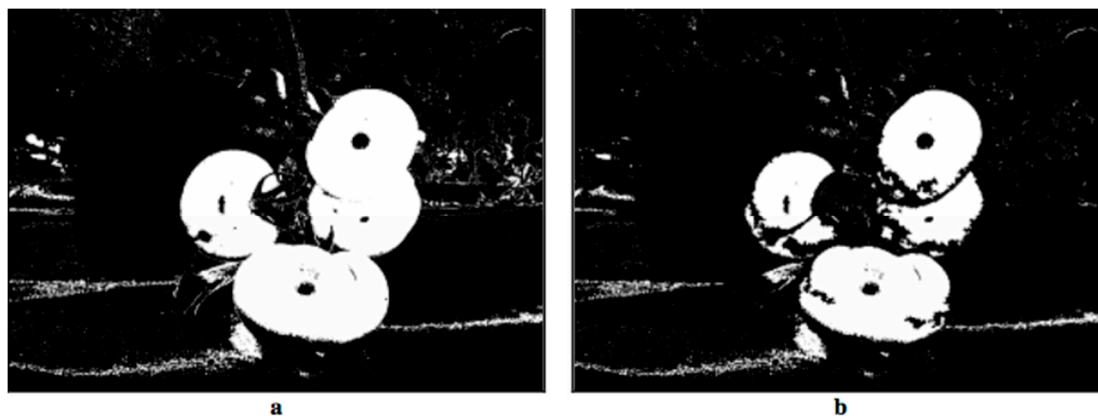

Figure 10. (**a**) Segmentation based on color 1; and (**b**) edge subtraction (Figures 8b and 9b).

- Color-Based Segmentation 2: Obtaining Separate Regions.

The subsequent processing stage (Figure 11a) was to perform a new segmentation based on color (segmentation in color—2), in order to achieve a binary image in which each ripe tomato appears represented by a single region, separated from the rest of the regions (Figure 11b). This task is quite complicated, since ripe tomatoes often appear in the image superimposed on one another, or so close to each other that their regions come together. The difficulty lies in the fact that ripe tomatoes are all practically the same color, which makes it very difficult to obtain a separate region for each of them. Nonetheless, the tomatoes appear much brighter in their central area, and darker as we get closer to the edges. This makes it easier for us to carry out a color-based segmentation in which only the central part of the ripe tomato is detected, meaning that the tomatoes appear represented by separate regions even if they overlap (Figure 11b).
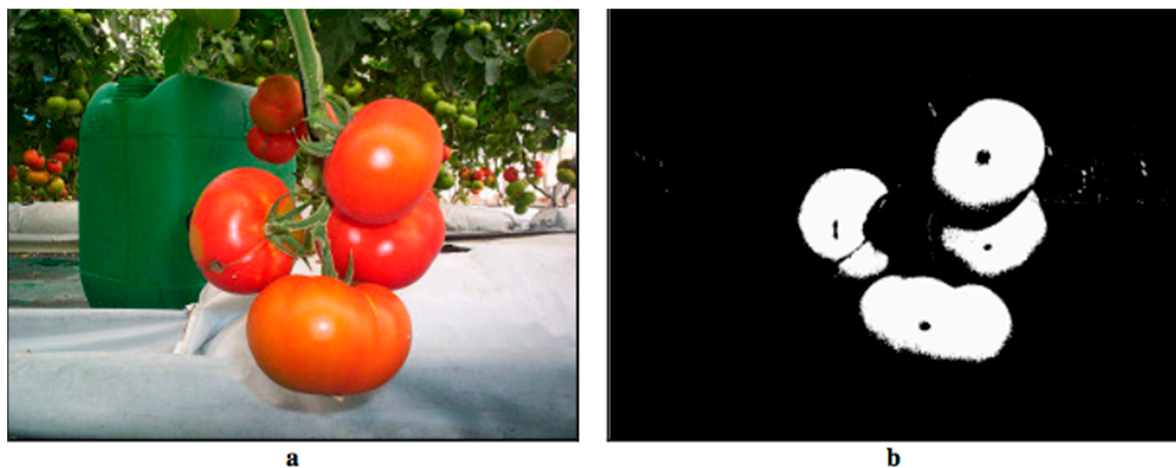
Figure 11. (**a**) Original image; and (**b**) color-based segmentation 2 (obtaining separated regions).

- Image combination (Figure 12): the binary images resulting from edge subtraction (Figure 10b) and color-based Segmentation 2 (Figure 11b) were combined into a single image using the OR (logical addition) operation. Sometimes, after subtracting the edges, a region belonging to the same tomato is divided into two or more smaller regions. The objective of this step is to link them to form a single region that represents the tomato. An added value is that the area of the regions corresponding to ripe tomatoes increases, maintaining the separation between them.
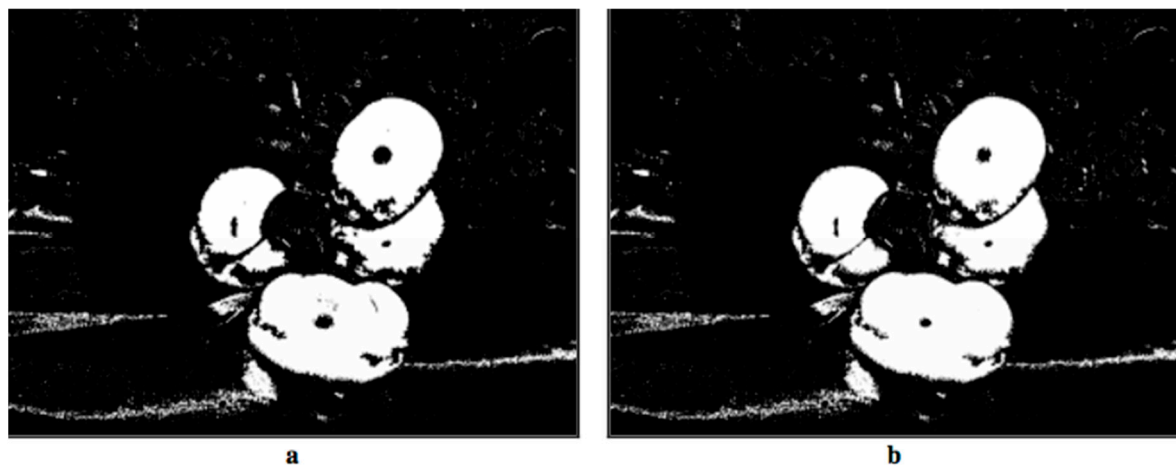

Figure 12. (**a**) Edge subtraction; (**b**) addition (OR) of Figures 10b and 11b.

- Segmentation based on size (Figure 13): in the binary image obtained after combining the images, not only do the regions appear that correspond to the ripe tomatoes in the foreground (which are the ones that really interest us), but many others also do, those belonging to tomatoes from more remote plants, and other objects that are in the environment whose color falls within the established segmentation thresholds, etc.
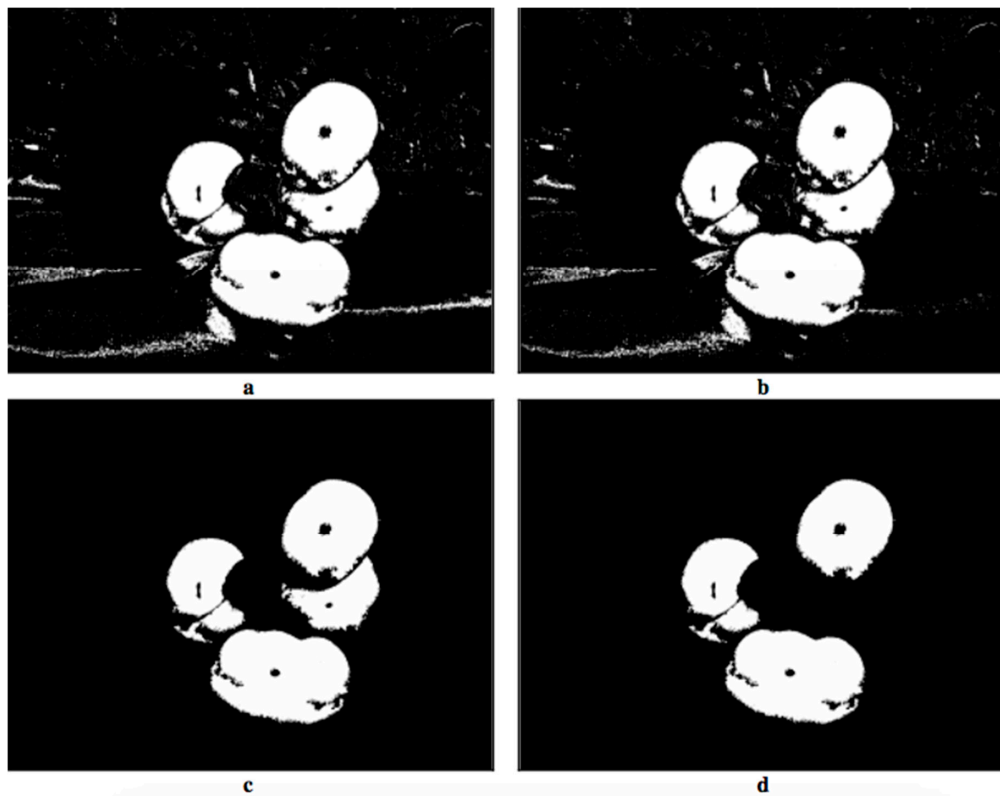
Figure 13. (**a**) Figure 12b; (**b**) edge removal of image objects; (**c**) segmentation based on size 1; and (**d**) segmentation based on size 2.

The objective of segmentation based on size is to eliminate all of these regions, keeping only those that represent the ripe tomatoes in the foreground. It also removes regions that belong to ripe tomatoes cut off by the edge of the image (Figure 13b). As we can see, two size-based segmentations were needed. The first segmentation (Figure 13c) to remove little regions, and the second to remove the regions that are less than half the size of the largest region (Figure 13d). In this way, the fact that no ripe tomato appears in the image is no longer a problem.

- Representation of the regions (Figure 14): this shows the user which regions obtained after the segmentation based on size represent the possible "collectible" tomatoes. To achieve this we computed the convex area of Figure 13d. Not all of these will be so, since it will depend on whether their peduncles are visible or not from the perspective from which the image was taken.
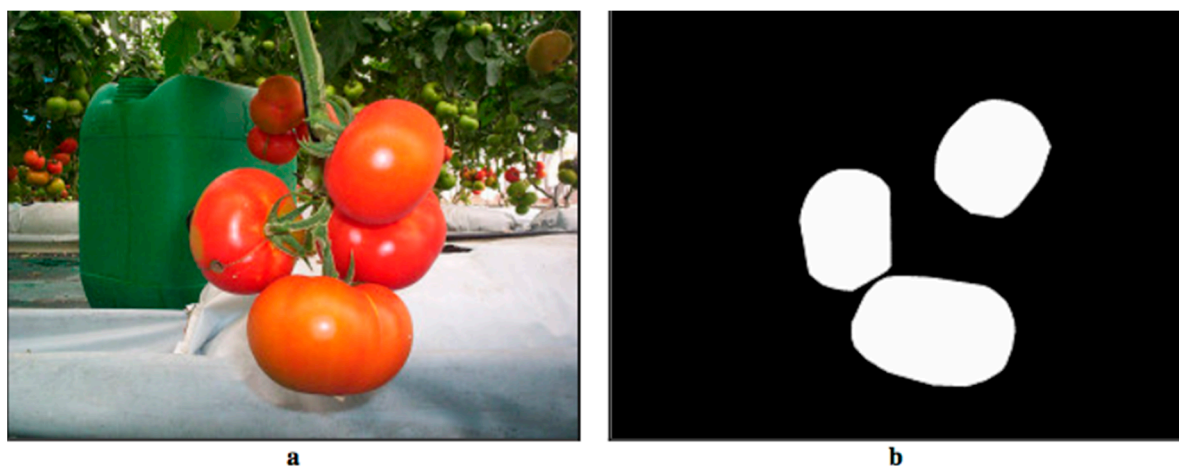

Figure 14. (**a**) Original image; and (**b**) convex area of Figure 13d.

2.4. Location of the Tomatoes and Their Peduncles

During this stage (Figure 15a), the system provides the location of each ripe tomato in the XY plane of the image by computing the gravity center (c.g.) of the convex area of each tomato. In the text, we call this the "center". In addition, it also calculates the position of the tomato's peduncle in the XY plane of the image; this is because, later on, it will be necessary to indicate to the robot the place where the ripe tomato must be separated from the rest of the plant. To begin this stage, the image from the previous stage was used (Figure 13b), in which only the regions representing ripe tomatoes (one region per tomato) appear. Before calculating the positions of the tomatoes and their peduncles, it is necessary to compute a series of descriptors for these regions.
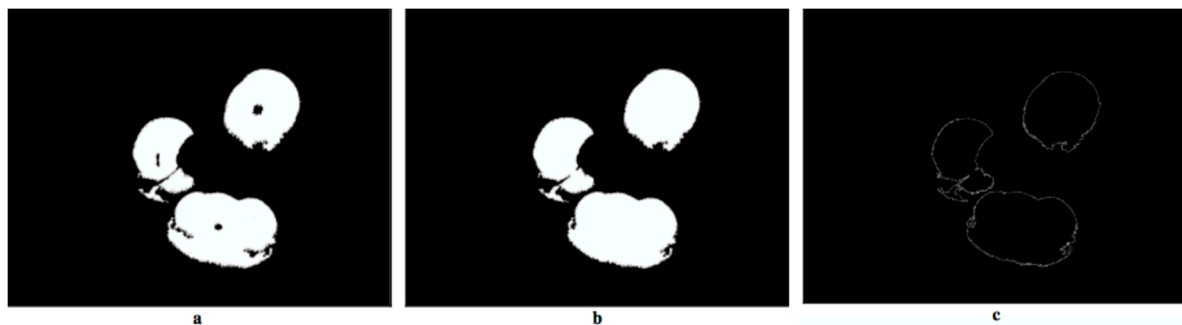

Figure 15. (**a**) Segmentation based on size 2; (**b**) gap filling; and (**c**) external gradient.

The regions obtained for each ripe tomato after the detection stage may have gaps or "holes" inside. The first operation is to fill in the gaps (Figure 15b) in order to make the measurements carried out below more precise. After that, we computed the external gradient of the previous image (Figure 15c).

For each of these regions, two sets of descriptors were obtained. The first set were:

- Center X: x coordinate (in pixels) of the region's c.g.;
- Center Y: y coordinate (in pixels) of the region's c.g.;
- Height and width in pixels of the circumscribed rectangle;
- Minor axis in pixels of the equivalent Feret ellipse;
- Orientation: ellipse orientation in degrees.

To obtain the second set of descriptors, we computed the external gradient of the regions without gaps. This operator returns a binary image with the external contour of the input image regions (Figure 15c). From this new image, we build the second set of descriptors, consisting of:

- X center: x coordinate (in pixels) of the center of gravity of the region's external contour. To distinguish it from Center X of the first set, we will call it Center XGdExt;
- Y center: y coordinate (in pixels) of the center of gravity of the region's external contour. To distinguish it from the Y Center of the first set, we will call it Y Center YGdEx*t*.

After carrying out a large number of tests using different combinations of these and other features, they proved to be the ones that gave us the most accurate results when locating the peduncles.

2.5. Plant Detection

The main objective of this process is to obtain the approximate position of the stem from which the tomatoes in the image "hang".

In addition to this data, we obtained a binary image showing the "green" parts (stem, branches, peduncles, calyces, etc.) of the plant that are in the foreground. Figure 16 show the steps to locate the plant stem´s centroid.
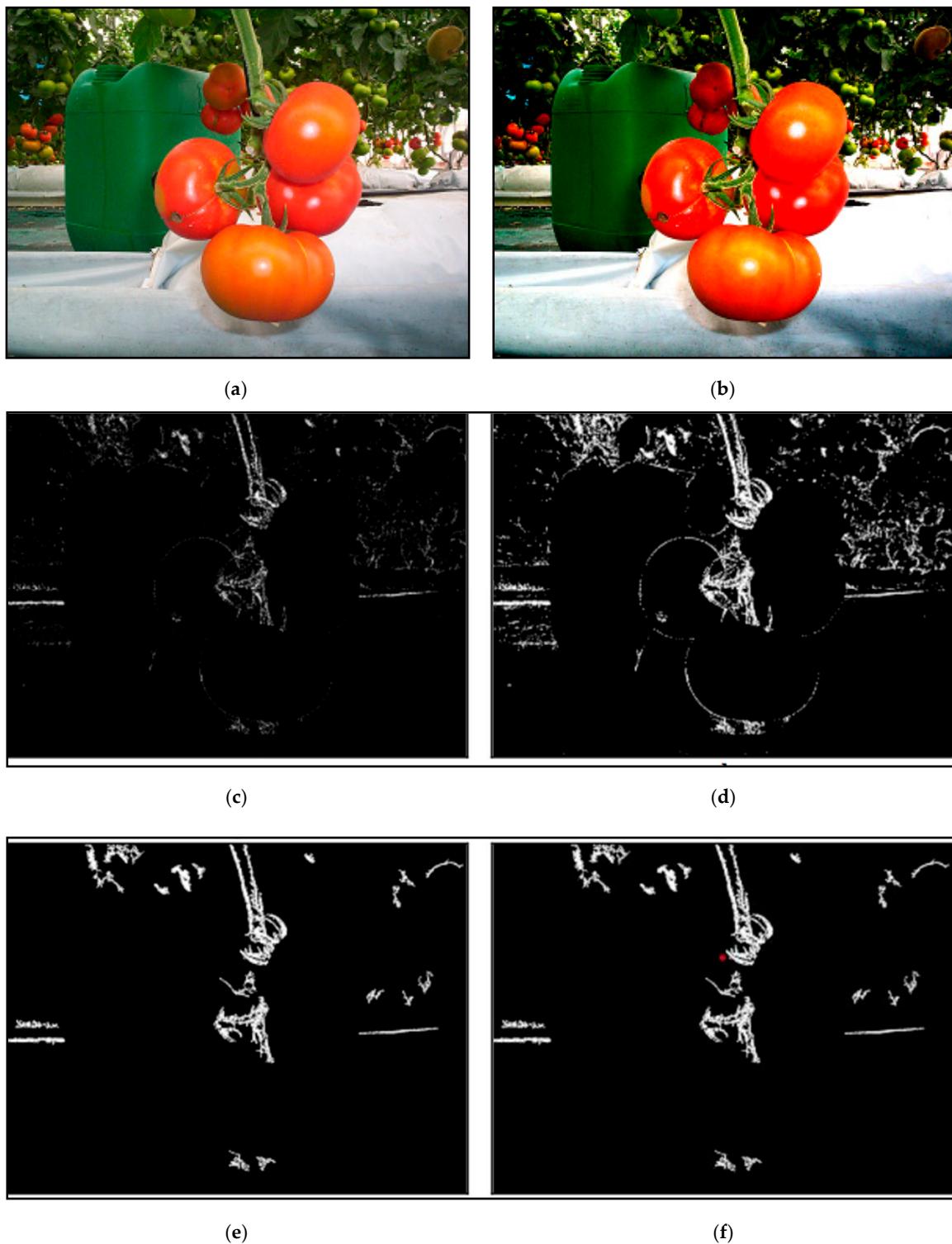
Figure 16. (**a**) Original image; (**b**) contrast enhancement; (**c**) color-based segmentation; (**d**) dilation; (**e**) size segmentation; and (**f**) centroid.

2.6. Peduncle Detection

The approximate position of the peduncle is achieved by applying a series of geometric rules based on the morphology of the plant (Figure 17), from which we obtained four possible peduncle positions for each mature tomato candidate to be collected. The final position of the peduncle will

be that meeting certain requirements. If none of the four possibilities fulfil these requirements, it is assumed that the peduncle is not visible, as is usually the case.
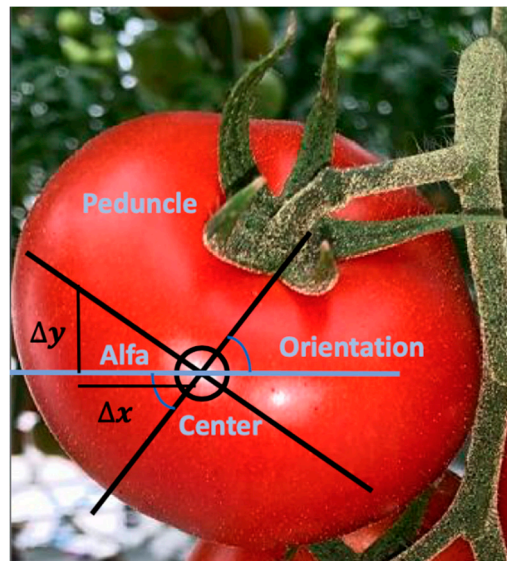

Figure 17. Geometric relationship based on tomato morphology.

Usually, the peduncle is on the upper straight line perpendicular to the tomato's main axis. Computing the centroid, the equivalent ellipse, and the major and minor axis of the ellipse or the circumscribed rectangle, and using elemental trigonometry, it is possible to compute Δx and Δy, and thus the peduncle position. Finally, it is necessary to check that the peduncle is not on the tomato and that it is over the plant (Figure 18).
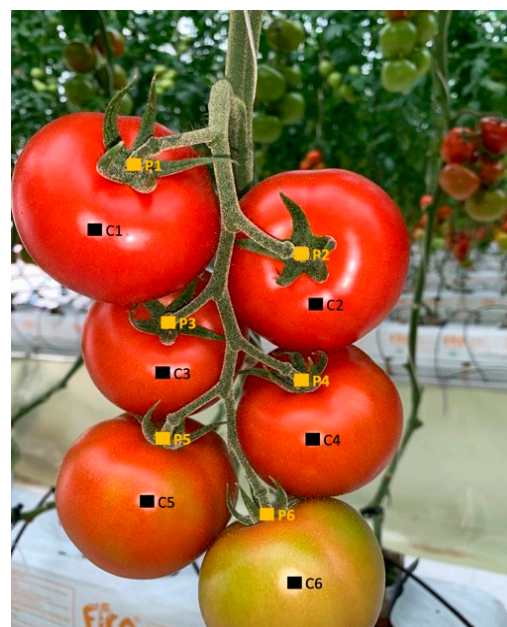

Figure 18. Convex area center (C) and peduncle (P) tomato detection.

3. Results

The system under study was tested using 175 images captured in a real environment for two different crop types: beef and cluster tomatoes. For each type, the success and failure rates (Tables 1 and 2) were calculated in relation to three different "objects":

1. That corresponding to the location of the tomatoes;
2. That corresponding to the location of the peduncles;
3. That corresponding to the tomato peduncle set.

Table 1. Results provided by the system for all beef tomato images.

Success and Failure Rates		Total Elements That Should Have Been Correctly Located by the System *
(a) Tomatoes	Success	90%
	Failure 1	10%
	Failure 2	0%
	Failure 3	0%
(b) Peduncles	Success	91.3%
	Failure 1	8.7%
	Failure 2	0%
	Failure 3	0%
(c) Tomatoes peduncles	Success	80.8%
	Failure 1	19.2%
	Failure 2	0%
	Failure 3	0%

* Tomato peduncle success = ((number of successful tests)/(total number of tests)) × 100.

Table 2. Success and error rates for the cluster tomatoes.

Success and Error Rates		Of the Total Elements That Should Have Been Correctly Located by the System *
(a) Tomatoes	Success	79.7%
	Failure 1	6.8%
	Failure 2	0%
	Failure 3	11.9%
(b) Peduncles	Success	69.5%
	Failure 1	27.1%
	Failure 2	0%
	Failure 3	3.4%
(c) Tomatoes peduncles	Success	63.2%
	Failure 1	29.4%
	Failure 2	0%
	Failure 3	7.4%

* Tomato peduncle success = ((number of successful tests)/(total number of tests)) × 100.

There are three different types of failures, which we will call:

- Failure 1: An object that should have been detected/located is NOT detected or located;
- Failure 2: An object is detected or located that should NOT have been detected/located;
- Failure 3: An object that should be detected/located by the system, is detected/located but not correctly.

The reason why the system must only detect fruit located in the foreground, and which are not occluded (or that their occlusion is not relevant), is because only tomatoes meeting these characteristics (in addition to those related to the degree of maturity) can be collected first. In order to detect (and collect) the occluded ripe tomatoes, it is first necessary to harvest the tomatoes that lie in front. Moreover, one must take into account that each time a tomato is harvested, the position of the other ripe fruit is usually affected. For these reasons, if the system were implemented on a real robot picker, a new image of the plant would need to be taken after each tomato is collected, and then recalculate the position of the next tomato for harvesting, since the picking of its neighbor could have altered its previous position.

3.1. Beef Tomatoes

The success and failure rate for the "tomato peduncle" set is what predicts the final success of the system, since it indicates how many of the ripe tomatoes with visible peduncles can finally be harvested. According to the results, 80.8% of these tomatoes were classified as "collectible" by the system. The system fails to detect the remaining 19.2% which, in theory, could also be collected. A very positive outcome is that there were no errors of location nor errors classifying "not harvested" tomatoes as "harvested". Figures 19 and 20 show two examples of the results for a set of beef tomatoes.

	Ripe Tomatoes		
Num.	**c.g. (X,Y)**	**Peduncle (X,Y)**	**Harvest**
1	(325.73, 469.90)	(326.21, 475.92)	Yes
2	(329.69, 442.94)	(328.53, 455.31)	Yes
3	(326.27, 423.13)	(325.89, 436.03)	Yes
4	(328.41, 401.54)	(328.02, 409.49)	Yes
5	(326.02, 386.27)	(328.18, 415.87)	Yes
6	None	None	No
Total harvesting fruits:			5
Processing time:			27 ms
Overview			
	Correct	**Fails**	**Total**
Fruits	5	1	6
Peduncles	6	0	6

Figure 19. Beef tomatoes: centers, peduncle detection and results.

	Ripe Tomatoes			
	Num.	**c.g. (X,Y)**	**Peduncle (X,Y)**	**Harvest**
	1	(825.94, 461.21)	(844.73, 482.91)	Yes
	2	(766.92, 463.40)	(785.00, 486.13)	Yes
	3	(887.47, 365.60)	(892.00, 393.82)	Yes
	Total harvesting fruits:			3
	Processing time:			25 ms
	Overview			
		Correct	**Fails**	**Total**
	Fruits	3	0	3
	Peduncles	3	0	3

Figure 20. Beef tomatoes: centers, peduncle detection and results.

For this type of crop, 100 images were taken; these images included configurations of all kinds: using only natural lighting, using the camera flash, images taken at very different distances, and even images in which the camera was not positioned perpendicular to the ground. Of those images, only 79 met the conditions established for correct system operation. We will analyze the results provided by the system for these beef-type tomato images (Table 1).

In a research experiment, errors are never desirable, but of the different types of errors that may occur, not all are equally important. For example, making a mistake when calculating the location of a tomato or its peduncle (Failure 3) is much more serious than the system not detecting a fruit or peduncle that it should have detected (Failure 1). In a research experiment, errors are never desirable, but of the different types of errors that may occur, not all are equally important. For example, making a mistake when calculating the location of a tomato or its peduncle (Failure 3) is much more serious than the system not detecting a fruit or peduncle that it should have detected (Failure 1).

This is because, if the system were implemented in a real collecting robot, a calculation error regarding the position of the fruit or peduncle could cause irremediable damage to the plant or surrounding fruit when trying to collect it. In contrast, not detecting a fruit or peduncle does not translate into any kind of harmful effect to the environment. As can be seen in Table 1, error types 2 and 3 are 0% in all cases. There is one failure 2, but it is partially covered and the peduncle is visible, which is an excellent outcome. The processing time per image was 27 ms.

3.2. Cluster Tomatoes

In this case, about 75 images were taken (42 met the conditions established for correct system operation). The system managed to classify as "collectible" 87.5% of tomatoes with visible peduncles. This percentage is at least as good as that obtained for the beef-type tomatoes, taking into account that the system was designed based solely on the results obtained for the beef-type crop. Figures 21 and 22 show two examples of the results for a set of cluster tomatoes.
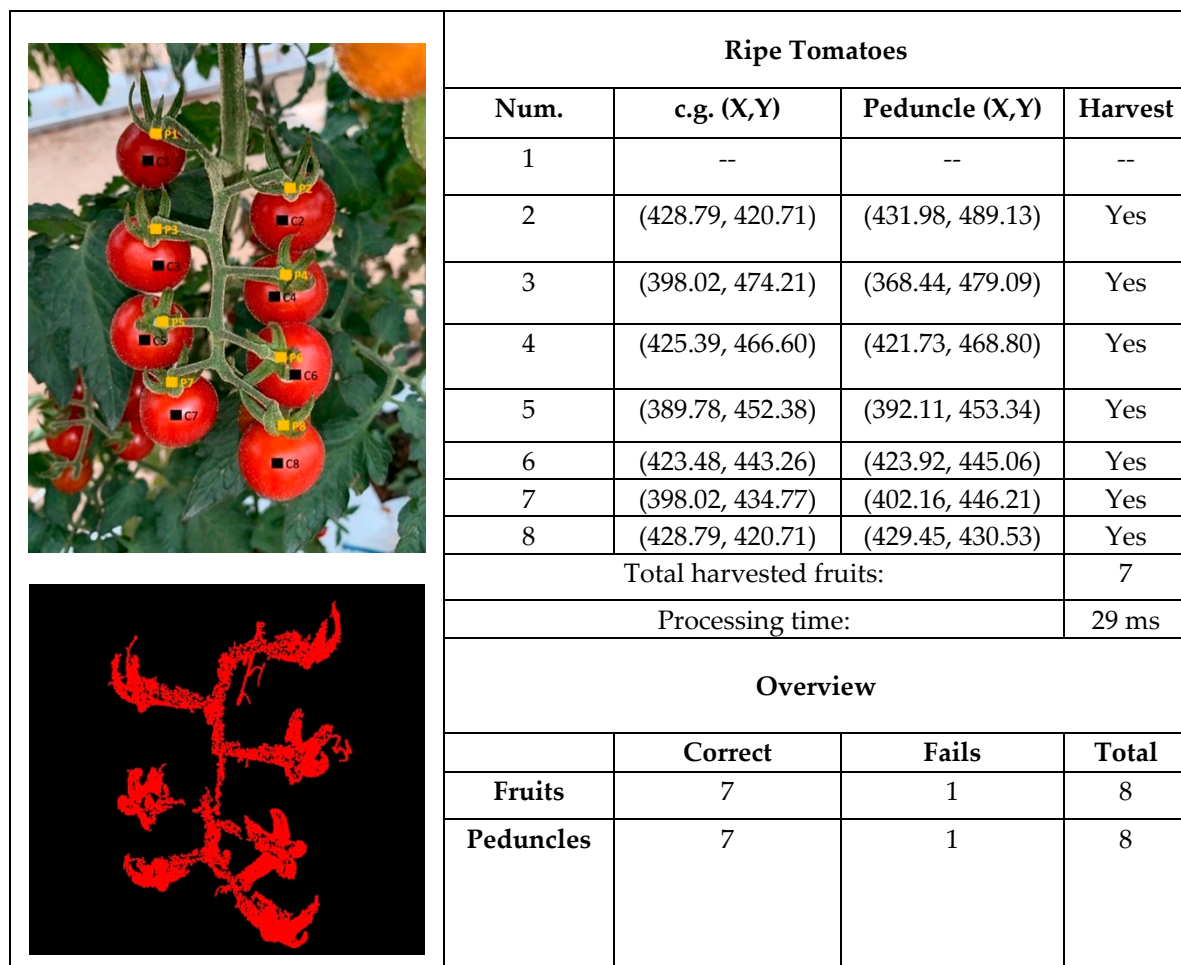
Ripe Tomatoes			
Num.	c.g. (X,Y)	Peduncle (X,Y)	Harvest
1	--	--	--
2	(428.79, 420.71)	(431.98, 489.13)	Yes
3	(398.02, 474.21)	(368.44, 479.09)	Yes
4	(425.39, 466.60)	(421.73, 468.80)	Yes
5	(389.78, 452.38)	(392.11, 453.34)	Yes
6	(423.48, 443.26)	(423.92, 445.06)	Yes
7	(398.02, 434.77)	(402.16, 446.21)	Yes
8	(428.79, 420.71)	(429.45, 430.53)	Yes
Total harvested fruits:			7
Processing time:			29 ms
Overview			
	Correct	Fails	Total
Fruits	7	1	8
Peduncles	7	1	8

Figure 21. Convex area centers, peduncle detection and results.

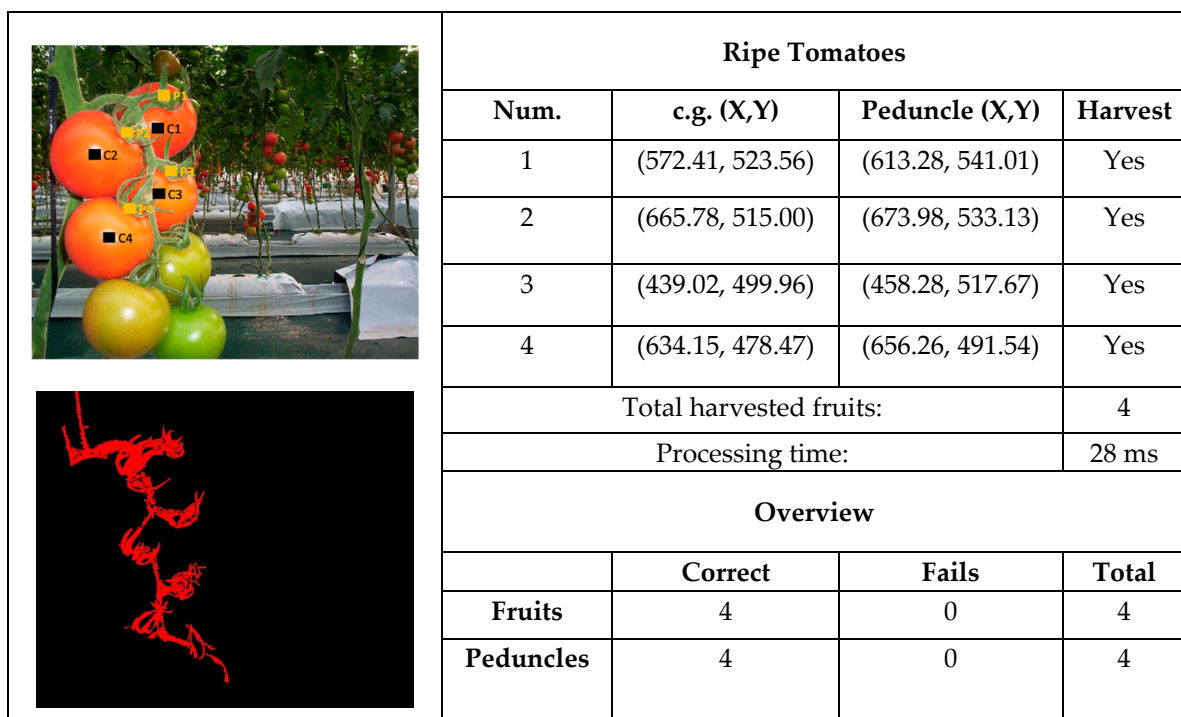| Ripe Tomatoes			
Num.	c.g. (X,Y)	Peduncle (X,Y)	Harvest
1	(572.41, 523.56)	(613.28, 541.01)	Yes
2	(665.78, 515.00)	(673.98, 533.13)	Yes
3	(439.02, 499.96)	(458.28, 517.67)	Yes
4	(634.15, 478.47)	(656.26, 491.54)	Yes
Total harvested fruits:			4
Processing time:			28 ms
Overview			
	Correct	Fails	Total
Fruits	4	0	4
Peduncles	4	0	4

Figure 22. Convex area centers, peduncle detection and results.

In Figure 21, eight ripe tomatoes appear. All are ready for harvest because they are mature tomatoes. There is a tomato (c1) in the shade that is ready to pick but was not detected by the system. The algorithm detects and correctly locates the other seven tomatoes for harvesting. The peduncles of the seven detected tomatoes are visible, and the system manages to locate them. These results could be improved significantly if we could make the images invariant against a set of transformations like light intensity and an affine transformation composed of translations, rotations and size changes. Table 2 shows the results for cluster tomatoes.

The average processing time per image for the visible ripe tomatoes and harvested tomatoes was 29 ms. The worst results were obtained with this type of tomato (17%); this might be due to having worked with 50% fewer images than were needed to meet the established requirements.

4. Discussion

In this work, the following objectives were achieved:

- Detection of ripe tomatoes: the system detected those ripe tomatoes located in the foreground of the image whose surfaces were not occluded by the plant or the fruit that surround it, or at least, not so much that they could not be collected. Specifically, it detected the "candidate" tomatoes to be collected, representing each of them by a single region (convex area) separated from the rest.
- Location of the ripe tomatoes in XY: once detected, the system located the ripe tomatoes in the XY plane of the image by calculating the position of their centers.
- Location of the tomato peduncle in XY: for each ripe tomato detected, the system indicated whether or not its peduncle was visible from the position where the image was captured. If the peduncle was visible, the system located it by providing its position in the image's XY plane and informed us that the tomato could be collected. If the peduncle was not visible, the system advises as such, and informs us that the tomato cannot be collected.

5. Conclusions

It is rarely a simple task, in a given field of study, to find the sequence of processes needed to enhance and segment images. In our case, it has been particularly complex. We consider that the main novelty and contributions of this work are:

1. The identification and location of the ripe tomatoes and their peduncles;
2. The computing time we achieved for the processing (identification and location) of an image was of the order of milliseconds, while in other works [18,24,27], it was of the order of seconds;
3. The use of flash to acquire the images minimized the illumination variations effects;
4. Another very important contribution of this vision system was that it can be used for any tomato-harvesting robot, without having to develop a new vision system for each end-effector prototype, because it locates the needed tomato parts for the different types of harvesting: cutting or embracing/ absorbing.

Furthermore, as we noted in this paper, this is only a first, yet important step, leading to other tasks that will complete the harvesting automation process (calculating the z-position and cutting or suctioning the tomatoes, improving the detection of tomatoes in poor lighting, etc.).

Consequently, the objectives proposed in this work were successfully achieved although there are numerous lines of research that could be followed in the future, both to improve the performance of the system that is already implemented and to expand its computer-vision functionality and versatility in detecting commercial fruit, or sorting the tomatoes by quality criteria such as color or size, or using other algorithms (for example, applying CNNs for tomato detection). In addition, the design of the robotic part of the project and the integration of the robot and computer-vision subsystems (the z-coordinate calculation, developing the cut-end effector, exploring pressure systems and picking tomatoes) should be studied.

References

1. Food and Agriculture Organization of the United Nations (FAO). Available online: http://www.fao.org/ (accessed on 10 September 2019).
2. Valera, D.L.; Belmonte, L.J.; Molina-Aiz, F.D.; López, A.; Camacho, F. The greenhouses of Almería, Spain: Technological analysis and profitability. *Acta Hortic.* **2017**, *1170*, 219–226. [CrossRef]
3. Callejón-Ferre, Á.J.; Montoya-García, M.E.; Pérez-Alonso, J.; Rojas-Sola, J.I. The psychosocial risks of farm workers in south-east Spain. *Saf. Sci.* **2015**, *78*, 77–90. [CrossRef]
4. Sistler, F.E. Robotics and Intelligent Machines in Agriculture. *IEEE J. Robot. Autom.* **1987**, *RA-3*, 3–6. [CrossRef]
5. Sarig, Y. Robotics of Fruit Harvesting: A State-of-the-art Review. *J. Agric. Eng. Res.* **1993**, *54*, 265–280. [CrossRef]
6. Ceres, R.; Pons, J.L.; Jimenez, A.R.; Martin, J.M.; Calderon, L. Design and implementation of an aided fruit—Harvesting robot. *Ind. Robot* **1998**, *25*, 337–346. [CrossRef]
7. Pons, J.L.; Ceres, R.; Jiménez, A. Mechanical Design of a Fruit Picking Manipulator: Improvement of Dynamic Behavior. In Proceedings of the IEEE International Conference on Robotics and Automation, Minneapolis, MN, USA, 22–28 April 1996.
8. Bulanon, D.M.; Kataoka, T.; Okamoto, H.; Hata, S. Development of a real-time machine vision system for the apple harvesting robot. In Proceedings of the SICE Annual Conference, Sapporo, Japan, 4–6 August 2004; Hokkaido Institute of Technology: Sapporo, Japan, 2004.
9. Gotou, K.; Fujiura, T.; Nishiura, Y.; Ikeda, H.; Dohi, M. 3-D vision system of tomato production robot. In Proceedings of the IEEE/ASME Int. Conference on Advanced Intelligent Mechatronics, Kobe, Japan, 20–24 July 2003; pp. 1210–1215.
10. Bac, C.W.; van Henten, E.J.; Hemming, J.; Edan, Y. Harvesting robots for high-value crops: State-of-the-art review and challenges ahead. *J. Field Robot* **2014**, *31*, 888–911. [CrossRef]
11. Bachche, S. Deliberation on Design Strategies of Automatic Harvesting Systems: A Survey. *Robotics* **2015**, *4*, 194–222. [CrossRef]
12. Zhao, Y.; Gong, L.; Huang, Y.; Liu, C. A review of key techniques of vision-based control for harvesting robot. *Comput. Electron. Agric.* **2016**, *127*, 311–323. [CrossRef]
13. Pereira, C.; Morais, R.; Reis, M. Recent Advances in Image Processing Techniques for Automated Harvesting Purposes: A Review. In Proceedings of the Intelligent Systems conference, London, UK, 7–8 September 2017.
14. Mavridou, E.; Vrochidou, E.; Papakostas, G.A.; Pachidis, T.; Kaburlasos, V.G. Machine Vision Systems in Precision Agriculture for Crop Farming. *J. Imaging* **2019**, *5*, 89. [CrossRef]
15. Schillaci, G.; Pennisi, A.; Franco, F.; Longo, D. Detecting Tomato Crops in Greenhouses Using a Vision-Based Method. In Proceedings of the III Int. Conference SHWFA (Safety Health and Welfare in Agriculture and in Agro-food Systems), Ragusa, Italy, 3–6 September 2012; pp. 252–258.
16. Ji, C.; Zhang, J.; Yuan, T.; Li, W. Research on Key Technology of Truss Tomato Harvesting Robot in Greenhouse. *Appl. Mech. Mater.* **2014**, *442*, 480–486. [CrossRef]
17. Feng, Q.; Wang, X.; Wang, G.; Li, Z. Design and Test of Tomatoes Harvesting Robot. In Proceedings of the IEEE International Conference on Information and Automation, Lijiang, China, 8–10 August 2015; pp. 949–952.
18. Zhao, Y.; Gong, L.; Zhou, B.; Huang, Y.; Liu, C. Detecting tomatoes in greenhouse scenes by combining AdaBoost classifier and colour analysis. *Biosyst. Eng.* **2016**, *148*, 127–137. [CrossRef]
19. Zhao, Y.; Gong, L.; Zhou, B.; Huang, Y.; Liu, C. Robust Tomato Recognition for Robotic Harvesting Using Feature Images Fusion. *Sensors* **2016**, *16*, 173. [CrossRef] [PubMed]

20. Li, B.; Ling, Y.; Zhang, H.; Zheng, S. The Design and Realization of Cherry Tomato Harvesting Robot Based on IOT. *iJOE* **2016**, *12*, 23–26. [CrossRef]

21. Zhao, Y.; Gong, L.; Zhou, B.; Liu, C.; Huang, Y. Dual-arm Robot Design and Testing for Harvesting Tomato in Greenhouse. *IFAC-PapersOnLine* **2016**, *49*, 161–165. [CrossRef]

22. Taqi, F.; Al-Langawi, F.; Abdulraheem, H.; El-Abd, M. A Cherry-Tomato Harvesting Robot. In Proceedings of the 18th International Conference on Advanced Robotics (ICAR), Hong Kong, China, 10–12 July 2017; pp. 463–468.

23. Wang, L.; Zhao, B.; Fan, J.; Hu, X.; Wei, S.; Li, Y.; Zhou, Q.; Wei, C. Development of a tomato harvesting robot used in greenhouse. *Int. J. Agric. Biol. Eng.* **2017**, *10*, 140–149.

24. Zhang, L.; Jia, F.; Gui, G.; Hao, X.; Gao, W.; Wang, M. Deep Learning Based Improved Classification System for Designing Tomato Harvesting Robot. *IEEE Access* **2018**, *6*, 67940–67950. [CrossRef]

25. Kamilaris, A.; Prenafeta-Boldú, F.X. Deep learning in agriculture: A survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [CrossRef]

26. Feng, Q.C.; Zou, W.; Fan, P.F.; Zhang, C.F.; Wang, X. Design and test of robotic harvesting system for cherry tomato. *Int. J. Agric. Biol. Eng.* **2018**, *11*, 96–100. [CrossRef]

27. Malik, M.H.; Zhang, T.; Li, H.; Zhang, M.; Shabbir, S.; Saeed, A. Mature Tomato Fruit Detection Algorithm Based on improved HSV and Watershed Algorithm. *IFAC PapersOnLine* **2018**, *51*, 431–436. [CrossRef]

28. Zhao, Y.; Gong, L.; Zhou, B.; Liu, C.; Huang, Y.; Wang, T. Dual-arm cooperation and implementing for robotic harvesting tomato using binocular vision. *Robot. Auton. Syst.* **2019**, *114*, 134–143.

29. Lin, G.; Tang, Y.; Zou, X.; Cheng, J.; Xiong, J. Fruit detection in natural environment using partial shape matching and probabilistic Hough transform. *Precis. Agric.* **2019**. [CrossRef]

30. Yuan, T.; Lin, L.; Zhang, F.; Fu, J.; Gao, J.; Zhang, J.; Li, L.; Zhang, C.; Zhang, W. Robust Cherry Tomatoes Detection Algorithm in Greenhouse Scene Based on SSD. *Agriculture* **2020**, *10*, 160. [CrossRef]

31. Yoshida, T.; Fukao, T.; Hasegawa, T. A Tomato Recognition Method for Harvesting with Robots using Point Clouds. In Proceedings of the 2019 IEEE/SICE International Symposium on System Integration (SII), Paris, France, 14–16 January 2019; pp. 456–461. [CrossRef]

32. González, R.; Woods, R. *Digital Image Processing*, 4th ed.; Pearson: New York, NY, USA, 2018.

33. Rodríguez, F.; Moreno, J.C.; Sánchez, J.A.; Berenguel, M. Grasping in agriculture: State-of-the-art and main characteristics. In *Grasping in Robotics*; Springer: London, UK, 2013; pp. 385–409.

34. Kondo, N.; Yata, K.; Iida, M.; Shiigi, T.; Monta, M.; Kurita, M.; Omori, H. Development of an end-effector for a tomato cluster harvesting robot. *Eng. Agric. Environ. Food* **2010**, *3*, 20–24. [CrossRef]

35. Ling, P.P.; Ehsani, R.; Ting, K.C.; Chi, Y.T.; Ramalingam, N.; Klingman, M.H.; Draper, C. Sensing and end-effector for a robotic tomato harvester. In Proceedings of the 2004 ASAE Annual Meeting, American Society of Agricultural and Biological Engineers, Golden, CO, USA, 4–6 November 2004; p. 1.

36. Monta, M.; Kondo, N.; Ting, K.C. End-effectors for tomato harvesting robot. In *Artificial Intelligence for Biology and Agriculture*; Springer: Dordrecht, The Netherlands, 1998; pp. 1–25.

37. Siciliano, B.; Khatib, O. *Springer Handbook of Robotics*; Springer: Berlin/Heidelberg, Germany, 2016.