

## Article

# Modeling Positions and Orientations of Cantaloupe Flowers for Automatic Pollination

Nguyen Duc Tai <sup>1</sup>, Nguyen Minh Trieu <sup>2</sup> and Nguyen Truong Thinh <sup>2,\*</sup>

<sup>1</sup> Department of Mechanical and Electro-Mechanical Engineering, National Sun Yat-sen University, Kaohsiung 804201, Taiwan; nguyenductai243@gmail.com

<sup>2</sup> Institute of Intelligent and Interactive Technologies, University of Economics Ho Chi Minh City—UEH, Ho Chi Minh City 70000, Vietnam; trieunm@ueh.edu.vn

\* Correspondence: thinhnt@ueh.edu.vn; Tel.: +84-903-675-673

**Abstract:** An automatic system for cantaloupe flower pollination in greenhouses is proposed to meet the requirements of automatic pollination. The system consists of a mobile platform, robotic manipulator, and camera that reaches the flowers to detect and recognise their external features. The main task of the vision system is to detect the position and orientation of the flower in Cartesian coordinates, allowing the manipulator to reach the pose and perform pollination. A comprehensive method to ensure the accuracy of the pollination process is proposed that accurately determines the position and orientation of cantaloupe flowers in real environments. The vision system is used to capture images, detect the flower, and recognise its state according to its external features, such as size, colour, and shape, thereby providing appropriate nozzle access during pollination. The proposed approach begins with a segmentation method designed to precisely locate and segment the target cantaloupe flowers. Subsequently, a mathematical model is used to determine the key points that are important for establishing the growth orientation of each flower. Finally, an inverse-projection method is employed to convert the position of the flower from a two-dimensional (2D) image into a three-dimensional (3D) space, providing the necessary position for the pollination robot. The experimental process is conducted in a laboratory and proves the efficacy of the cantaloupe flower segmentation method, yielding precision, recall, and F1 scores of 87.91%, 90.76%, and 89.31%, respectively. Furthermore, the accuracy of the growth-orientation prediction method reaches approximately 86.7%. Notably, positional errors in 3D space predominantly fall within the allowable range, resulting in a successful pollination rate of up to 83.1%.

**Keywords:** pollinating automation; computer vision; flower segmentation; flower model; smart agriculture; Vietnam agriculture



**Citation:** Duc Tai, N.; Minh Trieu, N.; Truong Thinh, N. Modeling Positions and Orientations of Cantaloupe Flowers for Automatic Pollination. *Agriculture* **2024**, *14*, 746. <https://doi.org/10.3390/agriculture14050746>

Academic Editors: Wei Hu, Zhiguo Zhou and Wenqing Zhao

Received: 5 April 2024

Revised: 5 May 2024

Accepted: 8 May 2024

Published: 10 May 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Currently, Vietnamese farmers are eager to integrate modern technologies into agricultural production. In particular, cantaloupe cultivation in greenhouses has become popular because cantaloupe flowers (*Cucumis melo*) are dioecious and incapable of self-pollination. Consequently, the robotic pollination of cantaloupes is essential for enhancing yield and improving fruit quality. Cantaloupe flowers typically rely on natural factors for pollination, such as wind and insects. However, these methods are fraught with uncertainty. In particular, inadequate pollination may be caused by poor weather or a lack of insects in greenhouses, leading to misshapen and undersized cantaloupes [1]. Hence, artificial pollination techniques, such as pollen spraying, pollen brushing, and pollen guns, have been employed to overcome the challenges associated with natural pollination methods. Although these methods have demonstrated greater efficiency than natural methods [2], their applicability is reduced when addressing the pollination needs of large cantaloupe farms because of their labour-intensive and time-consuming nature. Currently, rising labour costs due to an ageing population have increased difficulties in pollination, which

pose a significant challenge for cantaloupe pollination. To address these issues, experts and scientists have proposed automated pollination using robots to increase pollination success rates and fruiting efficiency [3]. This method ensures uniform pollination and more effective pollen utilisation [4]. The automatic pollination system involves two primary tasks: using computer vision with a camera to detect and locate flowers on cantaloupe plants, and manipulating a robotic arm to a suitable position for pollination without affecting trees, flowers, or other robotic systems.

The authors have previously studied various flower pollination methods, such as using drones, and concluded that robotic bees cannot fully replace biodiversity [5]. Moreover, the use of drones as bee substitutes increases pollination costs and lowers efficiency. One study proposed a robotic pollination system that utilised a pre-train model for the highly accurate detection of kiwifruit flowers and buds [6]. This study achieved a detection speed of 38.64 ms/image, and the recognised image size was  $4608 \times 3456$  px. However, this study focused solely on detecting flowers and buds and did not consider the pose of the flowers. Another study employed automatic kiwifruit pollination consisting of a wheeled platform, manipulator, and binocular camera system capable of capturing images of kiwifruit flowers and determining their positions [7]. A manipulator was used for pollination, and the location and direction of the flowers were determined based on the centres of gravity of the pistil and flower contour. Furthermore, Gao et al. [8] presented a novel pollinator that employed preferential flower selection and precise targeting. A manipulator was placed on a mobile platform with a tracking drive, and the flower identification process utilised an RGB-D camera vision system. The robotic system incorporated a three-degrees-of-freedom manipulator and tracking mobile platform. Natural pollination in greenhouses faces challenges because insects cannot access them. Therefore, artificial pollination that relies on robots and artificial intelligence (AI) is gaining traction. Light detection and ranging (LiDAR) technology has been employed for odometry and mapping to generate a three-dimensional (3D) point-cloud map [9]. The experimental dataset was collected in real time using LiDAR, inertial measurement units, and wheel odometry. Notably, these experiments were conducted in greenhouses and outdoor farms, yielding highly accurate results. In another study, a pollination system for tomatoes utilised robots and drones, leveraging a convolutional neural network (CNN) algorithm for flower-shape recognition [10]. The image analysis algorithm was tested in a tomato greenhouse and achieved an accuracy rate of at least 70%. A novel kiwifruit pollinating robot that offers enhanced efficiency, reliability, and cost-effectiveness has also been introduced [11]. This robot featured a mobile platform, a manipulator equipped with a novel air-assisted sprayer, and a vision system utilising a CNN algorithm. This system successfully recognised flowers and conducted pollination with an accuracy of 79.5%. A CNN model is proposed to detect the fruit in reference [12]. Furthermore, the Mask R-CNN model was employed to identify king flowers in apple orchards for pollination [13]. This model was based on a king flower segmentation algorithm for locating the king flower within a flower cluster, achieving accuracy in the range of 65.6% to 98.7% based on the flower's stage of development. Another study focused on designing an autonomous robotic navigation system for orchard harvesting by employing the master–slave mode [14]. This system consisted of two parts: an orchard transport robot as the master, and a picking robot as the slave. Given its outdoor use, the system was equipped with a global navigation satellite system in addition to the same sensor systems found in other harvesting robots. Current research on greenhouse automation focuses on moving platforms, manipulators, vision systems, and end effectors. For example, a 3D image-based tomato-harvesting system was proposed that utilised deep-learning algorithms to detect features in the Cartesian coordinate system [15]. A robotic system for harvesting sweet pepper fruits, known as SWEEPER, was developed, researched, and evaluated in a greenhouse setting [16]. This system consisted of a six-degrees-of-freedom robotic arm with a specially designed end effector and RGB-D camera system that was mounted on an autonomous cart. This robotic system was evaluated as being highly effective for harvesting sweet pepper fruits.

Cantaloupe is also known as rock or sweet melon and can typically weigh in the range of 0.5 to 5 kg. Cantaloupe is often used as a fresh fruit or dessert with ice cream or custard. In addition, its seeds can be consumed as snacks. The cantaloupe plant is usually grown at cool temperatures (22 °C to 33 °C) and high humidity (60% to 70%). However, the temperature in southern Vietnam is often hot and dry, so plants are often grown in greenhouses. The culturing temperature of a cantaloupe plant should not exceed 33 °C because the flowers fall off and fruit quality is poor. Cantaloupe plants have male and female flowers. In nature, pollination is usually carried out by insects or wind, but natural pollination activities are difficult to conduct in greenhouses. Therefore, pollination is often performed manually. However, this activity requires a large amount of manpower and meticulous operation. Therefore, the purpose of this study was to investigate the automatic pollination process. It should be noted that this study did not focus on presenting the structure of the robot. Instead, the goal of this study was to provide a method for identifying the position and developmental orientation of the flowers. The pollinating robotic system designed in this study consisted of a mobile platform, manipulator, visual identification system, and a central controller for the pollination spray, as shown in Figure 1. The pollinating robot system was equipped with a pan-tilt camera system that was attached to a mobile platform. The mobile platform moved based on the differential transmission, was made of aluminium, and included one passive and two active wheels. The platform carried the vision system and manipulator to the appropriate positions. The central controller controlled, analysed, and synthesised the data that was obtained from the vision system, sensor parameters of the arm, and moving platform based on a universal asynchronous receiver-transceiver. The flowers were randomly located and oriented in space. Therefore, the manipulator must be flexible so that it can direct the nozzle in its workspace. The manipulator had six degrees of freedom, including six rotational movements, to reach the positions and orientations required for flower pollination. The nozzle was fitted with a pollination system specifically designed for cantaloupe flowers.

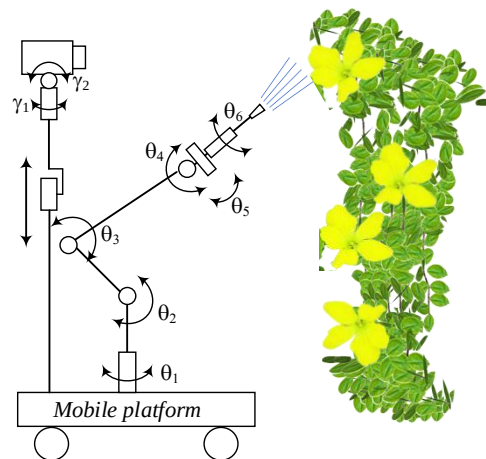


**Figure 1.** Schematic of designed pollinating robotic system.

Currently, research laboratories and companies have conducted many studies on pollinator robots. Studies have also focused on determining the position and orientation of flowers. However, flower orientation is based on pistil characteristics. In the authors' previous studies, mathematical modelling helped determine the position and orientation of many different flowers, independent of the biological and geometrical properties of the pistil. The objective and main contribution of this study was to create a comprehensive method for determining the position and orientation of flowers, which is important for accurate pollination, optimal use of pollen, and cost reduction. This study presents a cantaloupe pollination robot that captures the pollination targets and performs efficient pollination. Simultaneously, a strong foundation is established for integrated innovation research on intelligent cantaloupe pollination devices in later stages. In addition to the

benefits and contributions of this study, it can also be applied to other automated pollination processes for fruits and vegetables, allowing for more effective use of pollen in other automated pollination operations.

The remainder of this paper is organised as follows. The proposed methodology and a diagram of the proposed method are presented in Section 2. The position modelling and orientation of flowers based on algorithms, including flower segmentation, flower position determination, and key points used to predict flower growth orientation are presented in Section 3. The depth estimation methods and reverse-projection techniques for determining the positions of objects in 3D space from two-dimensional (2D) images are provided in Section 4. The experiments are presented in Section 5, and the conclusions of this study are provided in Section 6.

## 2. Proposed Methodology

Greenhouse environments used for planting are typically unstructured. Achieving successful pollination of flowers without any collisions or interactions with neighbouring flowers presents a significant challenge for robotic pollination. This study introduces a pollination method for determining the position and orientation of flowers that is based on images obtained using a vision system. For robots, implementing inverse kinematics is crucial for controlling the pollen nozzle with precision and ensuring that it reaches the required position and direction for effective pollination. The images provided by the vision system are 2D; however, robot control requires 3D control, specifically in the Cartesian space. Cantaloupe flowers are scattered irregularly in space, with many being obscured by fruits, leaves, and other flowers. Therefore, determining the precise 3D positions of these flowers is essential for successful pollination. Additionally, the orientation of each flower significantly influences the accuracy and efficiency of the pollination process. The nozzle must approach the pistil of the flower at an optimal distance during pollination. Thus, determining the orientation of the flower aids the robot in identifying the pistil of the target flower and understanding its growth orientation. Consequently, determining the position and orientation of the flowers plays a pivotal role in planning the optimal end-effector trajectory.

The procedure for determining the position and orientation of cantaloupe flowers is illustrated in Figure 2. Initially, the system camera captures two images, denoted as the  $i$ th and  $(i + 1)$ th images. These images are used to construct a depth map to determine the distance of the object from the camera. The depth map relies primarily on the  $i$ th image, which is considered the baseline image. Subsequently, a mathematical camera model is used to calculate the distance to the target object. Concurrently, the  $i$ th image undergoes segmentation using the proposed thresholding method to identify regions containing cantaloupe flowers. This stage also involves determining the key points of the flower that are required to ascertain the growth orientation of the flowers, such as the pistil and centre of gravity points. Subsequently, the acquired 2D information is transformed into 3D data within the camera-coordinate system using a mathematical model of the reverse-projection method. Ultimately, this process yields information regarding the position and growth orientation of the cantaloupe flowers, which serve as inputs for the controller to control the robot's actuators that are responsible for pollinating the flowers.

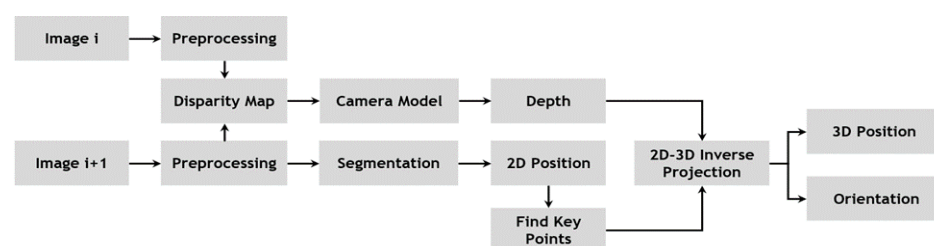


**Figure 2.** Diagram of the proposed method.

### 3. Modelling Position and Orientation of Flowers

This Section presents the image-processing techniques and mathematical methods used to determine the position and orientation of flowers within an image frame. It is necessary to define two reference points within the image frame to accurately ascertain the growth orientation of a cantaloupe flower. Cantaloupe flowers typically exhibit a round shape and are found growing on leaf axils. These flowers typically feature five yellow or pale yellow petals, with the colour of the pistil consistently darker than that of the petals. The petals are normally evenly arranged around the central pistil when viewed directly. Therefore, the shape of the petals will be unevenly arranged when the orientation of the flower changes. A method for determining the position and orientation of cantaloupe flowers within the image frame was proposed by leveraging their biological characteristics.

#### 3.1. Cantaloupe Flower Instance Segmentation Based on a Multi-Level Thresholding Algorithm

##### 3.1.1. Two-Level Thresholding

The implementation of cantaloupe flower segmentation is difficult in garden environments because of the influence of various factors, such as lighting conditions, varied colour and density distributions, environmental noise, and distortion. In addition to the difficulties caused by environmental factors, it is necessary to ensure the implementation of an algorithm with low computational resource consumption for applications in pollinating robots. Therefore, a method using multiple thresholds was proposed and implemented to address these conditions.

Two-level thresholding involves determining two different thresholds in the input image that are used to divide the image into parts based on similar properties. In this case, objects in an image that have close intensities and a large difference in the frequency transition at the contours probably belong to the same cluster. The intensity gradient between nearby pixels is a useful statistical parameter for estimating these features. Thus, a gradient image  $M_g(x, y)$  was created by calculating the difference in intensity between each pixel from the original image and the reference pixel value  $\lambda_a$ . Although this reference pixel value  $\lambda_a$  can be effective as a threshold in basic binary segmentation cases, it is insufficient in more sophisticated segmentation cases because it ignores the higher-frequency gradients (e.g., edges) of the elements in the image. The arithmetic mean value of the gradient image  $\lambda_g$  was calculated and the reference pixel value  $\lambda_a$  in the image was then adjusted to integrate the gradient information into the obtained threshold. The corresponding calculations are expressed as Equations (1)–(3).

$$\lambda_a = \frac{1}{W \times H} \sum_{x=0}^W \sum_{y=0}^H M(x, y) \quad (1)$$

$$M_g = M(x, y) - \lambda_a \quad (2)$$

$$\lambda_g = \frac{1}{W \times H} \sum_{x=0}^W \sum_{y=0}^H (M(x, y) - \lambda_a) \quad (3)$$

Two thresholds  $\mu_1$  and  $\mu_2$  were formed to divide the image into three groups  $k_1$ ,  $k_2$ , and  $k_3$  because  $\lambda_g$  was smaller than  $\lambda_a$ . In addition, the edges or border gradients that marked the object boundary could have various intensity levels in  $M_g(x, y)$  depending on their distance from  $\lambda_a$ . The first threshold  $\mu_1$  is a minus offset from the reference point value  $\lambda_a$ , while the second threshold  $\mu_2$  is an additional offset from the reference point  $\lambda_a$  with a value of  $\lambda_g$ . As a result, the thresholds and groups can be determined utilising (4) and (5).

$$\begin{cases} \mu_1 = \gamma_a - \gamma_g \\ \mu_2 = \gamma_a + \gamma_g \end{cases} \quad (4)$$

$$\begin{cases} k_1, \text{ if } 0 \leq M \leq \mu_1 \\ k_2, \text{ if } \mu_1 < M \leq \mu_2 \\ k_3, \text{ if } \mu_2 < M \leq L - 1 \end{cases} \quad (5)$$

where  $L$  denotes the intensity levels.

### 3.1.2. Global Thresholding

When a single threshold needs to be used, the suggested two-level threshold can be enhanced to provide image binarisation. The probability density function of the pixel value distribution of a greyscale image is used to accomplish this. Only one of the thresholds obtained in the two-level threshold ( $\mu_1$  or  $\mu_2$ ) can be utilised to split the pixels into two groups. The probability density function was used to determine the offset value from the reference point, which resulted in greater information acquisition after the image was divided into two groups. The cumulative total of the probability density function of pixels between the reference point value  $\lambda_a$  and the two points  $\mu_1$  and  $\mu_2$  can be calculated if  $\mu_1$  and  $\mu_2$  are rounded up to the closest integer. The probability density function  $P_i(n_i)$  can be determined if  $n_i$  is defined as the frequency of a pixel having  $i$  intensity in an image  $M(x, y)$  of size  $W \times H$ , which is expressed as (6). The two cumulative totals were compared to determine the global threshold  $T$  and implement image binarisation using  $T$  as (7).

$$P_i(n_i) = \frac{n_i}{WH} \quad (6)$$

$$T = \begin{cases} \mu_1, \text{ if } \sum_{i=\mu_1}^{\lambda_a} P_i(n_i) < \sum_{i=\lambda_a}^{\mu_1} P_i(n_i) \\ \mu_2, \text{ otherwise} \end{cases} \quad (7)$$

An illustration of an image that has been binarised using the proposed global thresholding method is shown in Figure 3.

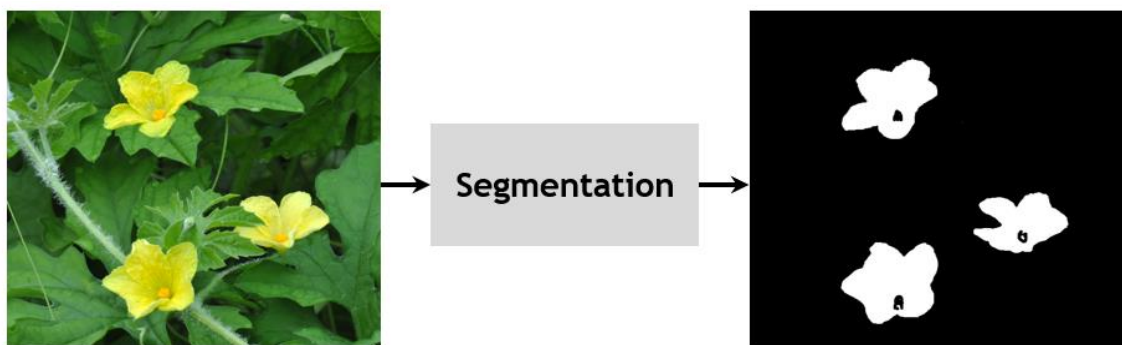


**Figure 3.** Result of the flower instance segmentation.

### 3.2. Determine the Key Points and Growth Orientation of the Flower in the Image Frame

An arrow was formed that connected the two key points of the cantaloupe flower (the point on the pistil of the female flower and point at the centre of gravity) that served as an indicator of the flower's growth orientation. Initially, image-processing techniques were employed to determine the pistil point of the female flower (point A). This determination was based on the inherent biological properties of the cantaloupe flower, in which the colour of the stigma consistently appears darker than that of the petals. Subsequently, an ellipse was fitted to the petal contour, with its centre serving as the centre of gravity of the petal contour (point B), as illustrated in Figure 4. The arrow connecting points A and B provided an approximate representation of the growth orientation of the cantaloupe flower.

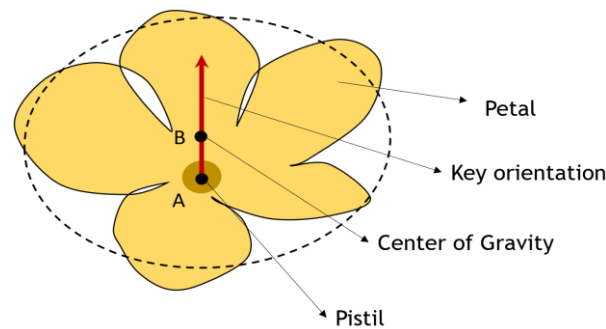


**Figure 4.** Model of the cantaloupe flower.

The following process is executed to obtain the pistil point of the female flower. First, the flower instance segmentation of each image (as discussed in Section 3.1) must be completed. Second, an ROI is created for each image consisting of a contour of the petals. Third, the original colour image is transformed into a greyscale image. The Gaussian blur algorithm is applied during this step to reduce noise in the greyscale image [17]. At this stage, an adaptive threshold is calculated based on the segmented binary and greyscale images to isolate the pistil region of the flower. Employing certain morphological techniques is crucial for enhancing detection accuracy owing to the consistent biological characteristics of all cantaloupe flowers, where the colour of the stigma is always darker than that of the petals. Fourth, the erosion and dilation method is immediately applied to the binary image after thresholding, resulting in a refined binary image that exclusively represents the pistil region of the female flower [18]. Finally, the centre of the circular contours within this binary image is identified as the first key point (referred to as point A). An illustration of the process of locating points on a pistil is illustrated in Figure 5.

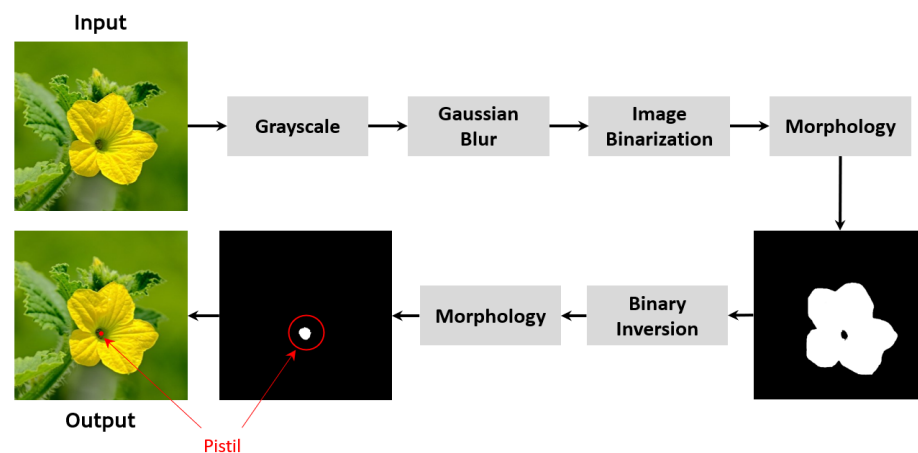


**Figure 5.** Method to determine key points on a cantaloupe flower.

Obtaining a point on the pistil is important for determining the orientation of flower growth. The growth orientation of the flower can be determined after establishing the second point (point B, the centre of gravity) based on the arrow connecting the two obtained points. Based on the biology of cantaloupe flowers, the shape distribution of each flower is primarily circular or elliptical. Accordingly, an ellipse was approximated that aligned with the flower shape. The centre of the ellipse corresponds to the second point belonging to the arrow, which indicates the orientation of the flower's development.

Several steps were performed to determine the centre of gravity. First, the extracted cantaloupe flower image was segmented to obtain a binary image representing the characteristics of each flower. Second, morphological processing techniques were applied to remove noise from the binary images. Third, bounding recognition was implemented to extract the features of the petal area from the binary image. Lastly, the contour areas of the

petals were detected and an ellipse was fitted to each contour. The problem solved in this Section is expressed as follows.

A set of 2D points  $S = \{p_i\}_{i=1}^n$  is obtained, where  $p_i = (x_i, y_i)$ . A set of curves  $C(k)$  is parameterised by the vector  $k$ . Then, curve  $C(k)$  is determined to be the best fit on the point dataset when finding the minimum  $k$  value, at which point the error function  $\varepsilon^2(k) = \sum_{i=1}^n \delta(C(k), p_i)$  reaches its global minimum. Here,  $\delta(C(k), p_i)$  denotes the measured distance from  $k$  points  $p$  to curve  $C(k)$ . The centre of the curve  $C(k)$  is also the flower's centre of gravity. The proposed approximate mean square (AMS) method was used to solve this problem [19]. An ellipse with a basic set of  $K = (x^2, xy, y^2, x, y, 1)$  has a set of  $A^T = (A_{xx}, A_{xy}, A_{yy}, A_x, A_y, A_0)$  coefficients.

The AMS method applies a condition  $A^T(D_x^T D_x + D_y^T D_y)A = 1$  to limit the fit to the elliptical curves, where the matrix  $D_x, D_y$  is the partial derivative of the matrix  $D$  with respect to  $x$  and  $y$ . The matrix formed in rows is applied to each point in the point set, which is expressed as (8). The AMS method optimises the following cost function to achieve the minimum value that is calculated in (9).

$$D(i, :) = \{x_i^2, x_i y_i, y_i^2, x_i, y_i, 1\} \quad (8)$$

$$\varepsilon^2 = \frac{A^T D^T D A}{A^T (D_x^T D_x + D_y^T D_y) A^T} \quad (9)$$

Accordingly, the obtained curve was the best-fit curve for the 2D set of points. The result of this process is shown in Figure 6a. An arrow was formed based on the two points found in the flower image, representing the growth orientation of the flower. The orientation of this arrow is the direction from the point on the pistil to the centre of the fitted ellipse. The results of orientation determination are shown in Figure 6b. In these figures, the green circle is an ellipse fitting the flower. The red dot marks the pistil, while the red line shows the ellipse's major and minor axes.

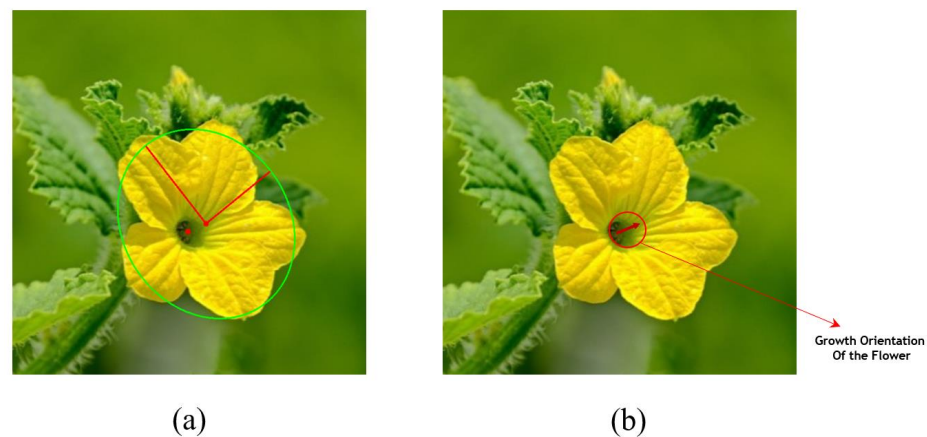


**Figure 6.** (a) Result of fitting an ellipse to the point set of the flower. (b) Result of determining the growth orientation of flowers.

### 3.3. Position of the Cantaloupe Flower in Cartesian Space

Camera parameter estimation is crucial for determining the relationship between a 3D point in the real world and its corresponding coordinates in an image frame. To project a 3D object onto an image frame, the world coordinate system of the point must first be converted to the camera coordinate system using an extrinsic matrix (rotation and translation matrices). Subsequently, the point is projected onto the image plane using the intrinsic matrix of the camera (focal length and distortion coefficients). The conversion of the camera coordinate system is shown in Figure 7.

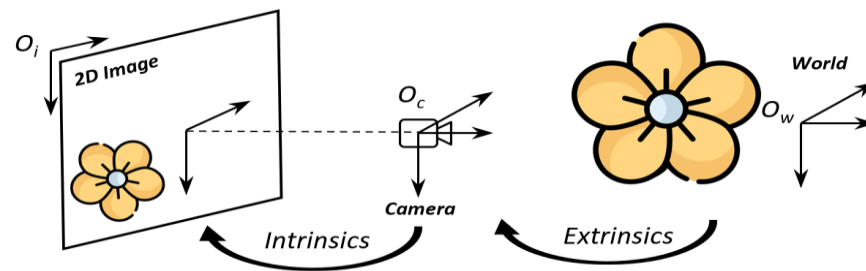


**Figure 7.** Conversion of the camera coordinate system.

The process of estimating the camera parameters or camera calibration is outlined as follows. Initially, the world coordinate system for a 3D object is established using chequerboards of a known size. The camera is then used to capture multiple images of the chequerboard from various angles. Subsequently, a set of test images with a chequerboard pattern must be inserted after the camera is linked to the computer. To ensure that the test images match the calibrator criteria, the calibration method finds the corners of the chequerboard using the `cv.findChessboardCorners` function in OpenCV to identify the coordinates of the corresponding 3D points in the image frame. Finally, the camera parameters are determined based on the `cv.calibrateCamera` function using real 3D and 2D point coordinates. The intrinsic parameters of the camera are obtained after completing the camera calibration process, as listed in Table 1. Calculations are established to create a linear model to find the position of the 3D flower in the real world. However, the distortion coefficient parameters have nonlinear properties. Therefore, image transformation to remove this distortion was performed independently based on an OpenCV algorithm after completing the calibration process.

**Table 1.** Parameters of the calibrated camera.

| Parameters              | Values                                                                                  |
|-------------------------|-----------------------------------------------------------------------------------------|
| Intrinsic matrix        | $\begin{bmatrix} 833.42 & 0 & 316.25 \\ 0 & 833.73 & 241.62 \\ 0 & 0 & 1 \end{bmatrix}$ |
| Distortion coefficients | $[0.01436 \quad -0.17403 \quad -0.08447 \quad -0.06694 \quad 0.12799]$                  |
| Principle Point         | $\begin{bmatrix} 316.25 & 241.62 \end{bmatrix}$                                         |
| Focal Length            | $\begin{bmatrix} 833.42 & 833.73 \end{bmatrix}$                                         |

A projection method was implemented to determine the position of flowers in 3D space from 2D images. In this Section, the position of the flower in the camera coordinate system (point  $P_c$ ) is obtained from the position of the flower in the image frame (point  $p_i$ ) using the 2D to 3D inverse-projection method. The focus in this study was on modelling the position of the cantaloupe flower relative to the camera's position. In addition, the extrinsic matrix parameters  $[R | t]$  were considered as  $[I | 0]$  to convert the world coordinate system to the camera coordinate system. First, the forward projection method calculates the position of point  $p_i = (a, b)$  in the image frame from point  $P_c = (x, y, z)$  in the camera coordinate system in 3D space, which is expressed as (10).

$$p_i = K[R|t] * P_c \quad (10)$$

where  $K$  represents the intrinsic matrix and  $R$  and  $t$  represent the rotational and translational elements of the extrinsic matrix, respectively.  $P_c$  denotes the point of the camera frame in the 3D space and  $p_i$  denotes the coordinate of the corresponding point  $P_c$  in the 2D image frame.

Matrix  $[R | t]$  can be written as  $[I | 0]$  because this calculation converts the image from the camera frame to the image frame. Consequently, Equation (8) can be rewritten as (11).

$$\varepsilon \begin{bmatrix} a \\ b \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (11)$$

The coordinates  $x$ ,  $y$ , and  $z$  denote the 3D position of a point within the camera coordinate system. This information helps to locate objects in 3D space. In addition,  $a$  and  $b$  represent the 2D coordinates of a point in the image frame, typically measured in pixels. The intrinsic parameter matrix  $K$  was obtained using the camera calibration process. The focal length of the camera is denoted as  $f$ . The focal lengths along the  $x$  and  $y$  axes ( $f_x$  and  $f_y$ , respectively) and principal point coordinates ( $c_x$ ,  $c_y$ , respectively) represent the centre of the image in pixels, where  $e$  is the scaling factor. After the algebraic transformations, the following equation can be obtained from Equation (12).

$$\begin{bmatrix} \varepsilon a \\ \varepsilon b \\ \varepsilon \end{bmatrix} = \begin{bmatrix} f_x x + y + c_x z \\ f_y y + c_y z \\ z \end{bmatrix} \quad (12)$$

Subsequently, Equation (10) can be rewritten as Equation (13).

$$\varepsilon = z \quad (13)$$

To convert the images from 2D to 3D, Equation (8) can be rewritten as (14).

$$\varepsilon \begin{bmatrix} a \\ b \\ 1 \end{bmatrix} = K[R|t] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (14)$$

The conversion is only performed between the camera coordinate system and image frame. Therefore, the transformation matrix  $[R | t]$  is written as  $[I | 0]$ . By applying algebraic operations, Equation (14) can be rewritten as (15).

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \varepsilon K^{-1} \begin{bmatrix} a \\ b \\ 1 \end{bmatrix} \quad (15)$$

From Equations (11) and (13), the position of a point in the 3D space in the camera coordinate system is based on the position of the pixel in the image frame. This is calculated using (16).

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = z K^{-1} \begin{bmatrix} a \\ b \\ 1 \end{bmatrix} \quad (16)$$

In Equation (14), the position of any pixel in the image frame can be obtained using 3D information regarding its position in the camera coordinate system when the matrix intrinsic  $K$  and depth  $z$  are determined. The  $z$ -depth value was determined based on the method presented in the Section below.

#### 4. Depth Estimation Based on Multi-View

A method was proposed to obtain the position of the flower in 3D space that helped the robot successfully pollinate. A depth-estimation approach based on multi-view observations was used to obtain 3D information about an object in the 3D space [20]. A camera was used to capture two images from different angles using the triangulation concept, and

the depth information was recovered by computing the disparity between the image pairs. The optical centre positions of the two images indexed as  $i$  and  $i + 1$  are denoted as  $O_l$  and  $O_r$ , respectively, as illustrated in Figure 8.  $O_l-X_lY_lZ_l$  and  $O_r-X_rY_rZ_r$  denote the coordinate systems of the image frames indexed as  $l$  and  $i + 1$ , respectively. The baseline distance  $d$  represents the horizontal separation between the optical centres when capturing images at the  $i$ th and  $(i + 1)$ th positions. In the image frames indexed as  $i$  and  $i + 1$ , the projected point values of  $P(X, Y, Z)$  are denoted as  $p_l(x_l, y_l)$  and  $p_r(x_r, y_r)$ , respectively. The projection model in the OXZ plane is shown in Figure 8. The following can be calculated based on the properties of similar triangles.

$$\frac{z}{f} = \frac{x}{x_l} \quad (17)$$

$$\frac{z}{f} = \frac{x - d}{x_r} \quad (18)$$

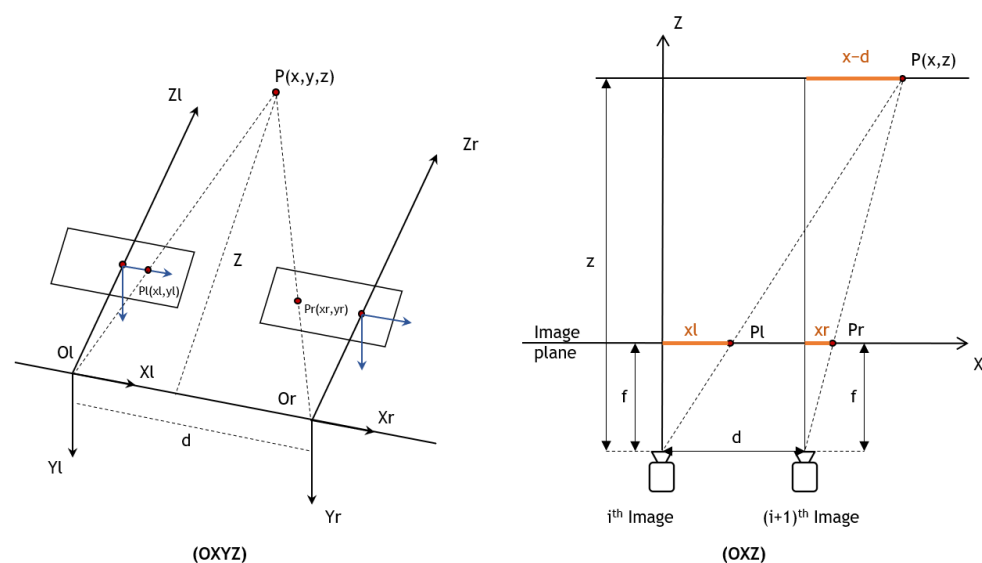


**Figure 8.** Multi-view model for the target object.

The following can be calculated from Equations (17) and (18), it is presented by (19).

$$\frac{x}{x_l} = \frac{x - d}{x_r} \quad (19)$$

Performing some transformations on Equation (19) results in Equation (20). However, considering Equations (17) and (20) results in (21).

$$\frac{d}{x_l - x_r} = \frac{x}{x_l} \quad (20)$$

$$z = f \left( \frac{d}{x_l - x_r} \right) \quad (21)$$

The  $x_l - x_r$  denotes the disparity value according to Equation (21), which is the displacement of point  $P$  at the projection point with respect to the plane of the images indexed at  $i$  and  $i + 1$ . Therefore, when the parameters  $d$  and  $f$  and the coordinates of point  $P$  in each image plane are determined, the depth  $z$  of that point is obtained. Therefore, it is important to calculate the positions of points  $p_l(x_l, y_l)$  and  $p_r(x_r, y_r)$  in each respective image frame for the images indexed at  $l$  and  $i + 1$ , respectively, to obtain information about the depth of point  $P$  in 3D space. Calculating the disparity  $x_l - x_r$  of a pixel in the  $i$ th image compared to that of the  $(i + 1)$ th image is necessary to determine the  $z$ -depth of that pixel in practice. Hence, the census transformation method was used to calculate

and build a disparity map [21]. A traditional local approach called the census transform determined the matching cost by converting the greyscale values of the pixels into bit strings. To accomplish this, the grey values of each pixel were compared to the grey value of the window's central pixel. This was achieved when both the window dimensions were odd numbers ( $u$  and  $v$ ). Following this comparison, the Boolean result was converted into a bit string, which was eventually transformed into the census transformation value of the central pixel. The calculation for this process is expressed as (22).

$$C(x, y) = \bigotimes_{i=-u'}^{u'} \bigotimes_{j=-v'}^{v'} \zeta(M(x, y), M(x + i, y + j)) \quad (22)$$

where  $u'$  and  $v'$  represent the maximum integers that do not exceed half the values of  $u$  and  $v$ , respectively, and  $\bigotimes$  denotes a bitwise connection operation.  $M(x, y)$  denotes the greyscale value of the image at the point to be matched, specifically at coordinates  $(x, y)$ .  $(x + i, y + j)$  represents the greyscale values of neighbouring points within the local area centred around the point to be matched. The  $\zeta$  operation is expressed as (23).

$$\zeta(M(x, y), M(x + i, y + j)) = \begin{cases} 1, & \text{if } M(x, y) \leq M(x + i, y + j) \\ 0, & \text{otherwise} \end{cases} \quad (23)$$

After applying the census transformation, the pixel values within the window are replaced with a binary bit string composed of zeroes and ones. The sorting of these bit strings depends solely on the values of the central and neighbouring pixels within the window. The ultimate cost calculation was performed using these bit strings in the left  $C_l(x, y)$  and right  $C_r(x - d, y)$  images using the Hamming distance, which is expressed as (24).

$$C(x, y, d) = \text{Hamming}(C_l(x, y), C_r(x - d, y)) \quad (24)$$

The Hamming distance quantifies the dissimilarity between two distinct bit strings. It is computed by performing an 'exclusive or' operation on the two-bit strings, and subsequently tallying the number of bits that differ from one in the result. A larger Hamming distance indicates a lower matching accuracy for the corresponding bit pixels. The census transformation and Hamming distance for the  $3 \times 3$  windows are shown in Figure 9. The disparity map generated using this algorithm is shown in Figure 10. Therefore, the depth can be determined relative to the camera object based on the disparity value obtained for any pixel in the  $i$ th image. The  $z$ -depth is the final value used to determine the 3D position of the flower in the real world in the camera's coordinate frame.

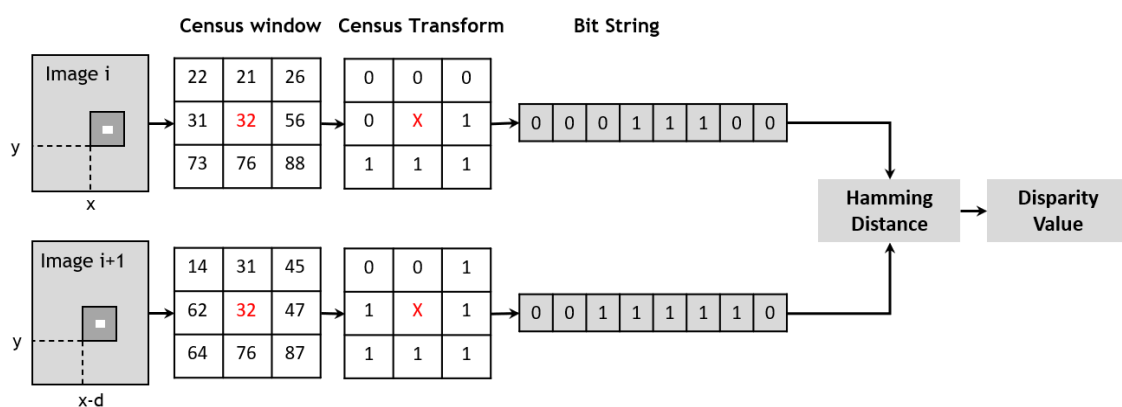


Figure 9. Census transformation method.

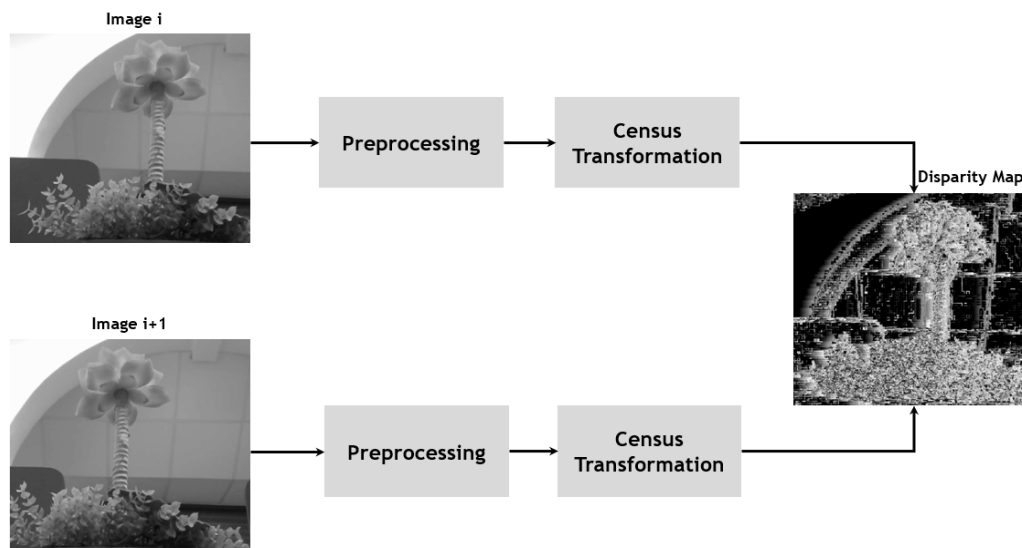


Figure 10. Method of generating the disparity map.

## 5. Experiments and Discussions

### 5.1. Verify the Efficiency of the Cantaloupe Flower Segmentation Process

The experiments were conducted on cantaloupe plants in a greenhouse at the UEH University laboratory. A dataset was collected that included 89 images of the greenhouse cantaloupe flowers captured with different shooting angles and brightness to facilitate the experimental process. The developed system used the proposed algorithm to determine the position and orientation of flowers in a greenhouse. The first part of the experiment evaluated the segmentation efficiency for cantaloupe flowers. Performance metrics, such as precision, recall, and *F1*-score were used to assess the effectiveness of the proposed thresholding approach. The metrics were computed for binarization using true positive (*TP*), false positive (*FP*), true negative (*TN*), and false negative (*FN*) attributes. In contrast to background pixels that are logical zeros, foreground pixels with logical zeros were considered positive. Precision is the percentage of *TP* pixels out of all the positive pixels present in the predicted binary image. Precision provides a probabilistic measurement of the number of positively predicted pixels. Recall is the percentage of positive pixels in the ground-truth image that are *TP* pixels. The harmonic mean between the recall and precision is known as the *F1*-score. The precision, recall, and *F1* calculations are, respectively, expressed as (25)–(27).

$$precision = \frac{TP}{TP + FP} \quad (25)$$

$$recall = \frac{TP}{TP + FN} \quad (26)$$

$$F1 - score = 2 \times \frac{precision \times recall}{precision + recall} \quad (27)$$

The flower segmentation performance of the Otsu [22] and proposed methods are listed in Table 2. Cases 1–4 in Table 2 represent typical images from the test dataset that was composed of 89 images. Visually, the proposed model outperformed the classic Otsu method. In specific cases, such as Case 3 where the image was less affected by environmental factors, the performance of the classic method was similar to that of the proposed method. However, the Otsu method exhibited poor results for images that were significantly affected by lighting conditions and noise. In contrast, the proposed method could separate the border of the flower area and filter out noise in the resulting image, which created favourable conditions for the following process. The flower segmentation

process in the test set was evaluated, and the results are listed in Table 3. The values in this Table are the average values of all 89 cases. The precision, recall, and *F1*-score values of the Otsu method were 75.23%, 79.61%, and 77.36%, whereas those of the proposed method were 87.91%, 90.76%, and 89.31%, respectively. These results proved that the proposed method was superior to the Otsu method. In this experiment, the proposed method produced nearly 90% accurate results even when the environmental conditions were changed. Therefore, it is feasible to apply the proposed technique to a greenhouse environment.

**Table 2.** Examples of segmentation results from the proposed and Otsu methods.

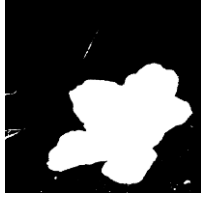
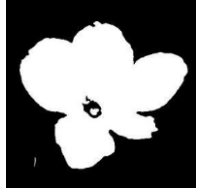
| Case | Original                                                                            | Otsu Method                                                                           | Proposed Method                                                                       |
|------|-------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| 1    |    |    |    |
| 2    |   |   |   |
| 3    |  |  |  |
| 4    |  |  |  |

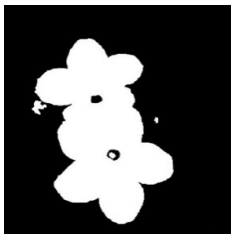
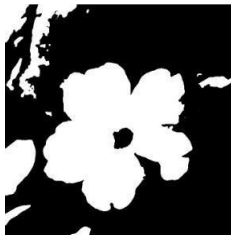
**Table 3.** Results on the segmentation accuracy of the proposed method.

| Method          | Precision | Recall | <i>F1</i> -Score |
|-----------------|-----------|--------|------------------|
| Otsu            | 0.7523    | 0.7961 | 0.7736           |
| Proposed Method | 0.8791    | 0.9076 | 0.8931           |

During the cantaloupe flower segmentation process, there are instances where flower segmentation may be missed or the binary image may contain excessive noise. Examples of error segmentation cases are listed in Table 4. Specifically, Case 1 demonstrates two flowers that are fused together. Here, the proposed approach was unable to segment them and considered them as part of the same flower. Case 2 demonstrates changes in lighting conditions that resulted in an image with varying light and dark areas, which led to a poor segmentation outcome and a binary image with significant noise. The causes of missed flower segmentation were analysed as follows. (1) In Case 1, cantaloupe flowers grew in a climbing form because of their biological characteristics, which often causes overlapping

where one flower covers another. This overlap results in outer flowers having significantly more visible characteristics than those of the inner or obscured flowers. The biological characteristics of these two flowers were almost identical, which posed challenges for the segmentation method that led to missed segmentation. (2) In Case 2, natural lighting conditions strongly affected the segmentation process. The proposed method struggled to extract the characteristics of the flowers when they were photographed in poor or complex lighting environments with alternating light and dark areas. This resulted in segmentation problems and missed segmentation. These cases pose challenges in determining the positions and orientations of flowers in subsequent processes, highlighting the limitations of our method. For example, variations in lighting conditions or the influence of wind can alter the position and shape of flowers, creating uncertainties that significantly impact the identification and location processes. These uncertainty factors reduce the overall accuracy of the proposed method, indicating the need for further research to develop solutions and address existing issues in future studies.

**Table 4.** Examples of error segmentation cases of the proposed method.

| Case | Original                                                                             | Proposed Method                                                                       |
|------|--------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| 1    |   |   |
| 2    |  |  |

## 5.2. Evaluating the Accuracy of Determining the Growth Orientation of Cantaloupe Flowers

Determining the growth orientation of flowers is based on identifying their key points. It is possible to derive the pistil and centre of gravity points through mathematical approximation and image processing. An arrow representing the growth orientation of the flower can be derived from these two points, as shown in Figure 11. The red bounding box and label represent the positions and names of the individual cantaloupe flowers. The orange and black arrows represent the orientation of flower growth predicted by the proposed method and the axis containing the correct orientation of each labelled flower, respectively.

The predicted results obtained using the proposed method were compared with the actual results to evaluate the effectiveness of predicting the flower growth orientation. A statistical analysis of the data was also conducted. The evaluation scheme was implemented as follows. First, the proposed method was applied to determine the arrows representing the growth orientation of the flower, which were then compared to the labelled actual arrows. Subsequently, ImageJ software version 1.54 was used to measure and record the angle of deviation between the two arrows. The prediction of the flower's growth orientation was considered to be accurate when the deflection angle was less than  $5^\circ$ . A dataset was developed consisting of 89 images containing flowers to conduct this experimental process, and the results are presented in Table 5. The mean deviation angle between the arrow formed by the proposed method and the actual angle was  $3.89^\circ$ , indicating that most of the deviation angles were below the specified threshold of  $5^\circ$ . The accuracy of the method was

approximately 86.7%. Therefore, the proposed method effectively predicted the growth orientation of flowers within an acceptable margin of error.



**Figure 11.** Determination of the growth orientation of the cantaloupe flower.

**Table 5.** Statistical results for determining the accuracy of growth orientation of the flower.

| Case | Number of Flowers in Image | Deviation Angle (°) | Mean Value of All of Deviation Angle Values | Accuracy (%) |
|------|----------------------------|---------------------|---------------------------------------------|--------------|
| 1    | 1                          | 2.9                 | 3.89                                        | 86.7         |
| 2    | 2                          | 3.7; 4.1            |                                             |              |
| 3    | 2                          | 4.4; 6.7            |                                             |              |
| 4    | 3                          | 4.8; 3.1; 3.7       |                                             |              |
| 5    | 2                          | 2.9; 3.7            |                                             |              |
| ...  | ...                        | ...                 |                                             |              |
| 85   | 3                          | 3.2; 3.1; 2.9       |                                             |              |
| 86   | 2                          | 2.1; 1.9            |                                             |              |
| 87   | 1                          | 6.3                 |                                             |              |
| 88   | 3                          | 4.1; 6.6; 5.2       |                                             |              |
| 89   | 1                          | 3.5                 |                                             |              |

### 5.3. Evaluating the Accuracy of Determining the Position of Cantaloupe Flowers in 3D Space

Obtaining the position of the flower in 3D space was based on the 2D to 3D inverse-projection and depth-estimation methods presented in Section 4. First, the position of a point in the image frame was converted to a point in 3D space with a position relative to the camera frame using algebraic calculations. Here, the depth of the point was obtained using the depth estimation method. The results are shown in Figures 12 and 13, where the pistil position was estimated in 3D space using three parameters ( $x$ ,  $y$ , and  $z$ ) relative to the camera frame. Here, the camera is the origin of the coordinate at the time of taking the  $i$ th image. The results of the proposed method were compared with the real values to comprehensively evaluate the effectiveness of the method. Experiments were conducted in a laboratory environment and greenhouse. For the laboratory experiment, a cantaloupe flower was attached to a specialised angle-measuring device to change the angle of the flower relative to the camera. The process of setting up the experimental subjects in the laboratory is shown in Figure 12.

The results obtained in the laboratory are presented in Table 6, and those in the greenhouse are presented in Table 7. In laboratory experiments, environmental conditions were fixed by placing a black curtain behind the objects and fixed lighting and wind. Meanwhile, experiments in the greenhouse, the experiment time is in the morning at about 9:00 a.m, without major changes in light and wind conditions. As shown in Tables 6 and 7, 3D position determination of the flowers could determine the position of the flowers relative to the camera coordinate system of the  $i$ th image. The proposed method exhibited a certain margin of error within the range of 0.1 to 1.6 cm. In fact, considering the biology of the flower and structure of the robot's nozzle, it was deemed acceptable for the discrepancy

error between the proposed method and real value on the  $x$  and  $y$  axes to be less than 0.3 cm for accurate pollination. Similarly, an error of less than 1 cm on the  $z$ -depth axis was considered acceptable for successful pollination.

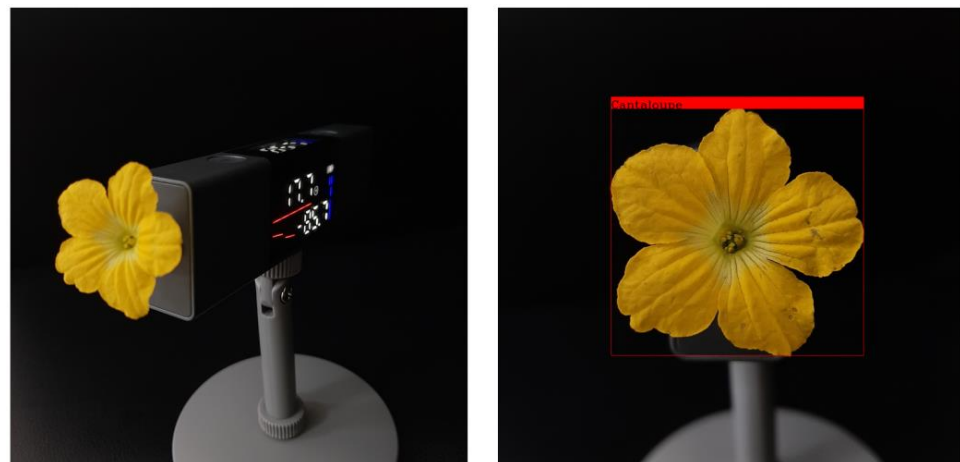


Figure 12. Flower experiments in the laboratory.

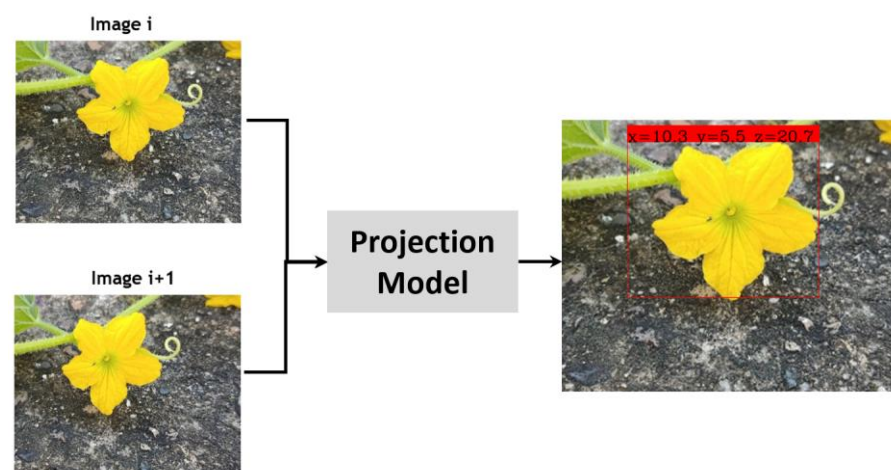


Figure 13. Result of determining the 3D position of the flower.

Table 6. Statistical results for determining the 3D position of a flower in the laboratory.

| No. | Proposed Method ( $x, y, z$ ) (cm) | Ground Truth ( $x, y, z$ ) (cm) | Position Error ( $dx, dy, dz$ ) (cm) | Pollination | Success Rate (%) |
|-----|------------------------------------|---------------------------------|--------------------------------------|-------------|------------------|
| 1   | 8.5; 6.3; 24.5                     | 8.7; 6.6; 25.3                  | 0.2; 0.3; 0.8                        | Success     | 86.5%            |
| 2   | 6.9; 7.3; 29.6                     | 7.1; 7.5; 30.4                  | 0.2; 0.2; 0.8                        | Success     |                  |
| 3   | 7.6; 8.1; 21.6                     | 7.8; 8.4; 22.3                  | 0.2; 0.3; 0.7                        | Success     |                  |
| 4   | 7.8; 9.2; 27.1                     | 7.9; 8.9; 26.2                  | 0.1; 0.3; 0.9                        | Success     |                  |
| 5   | 6.2; 8.7; 31.7                     | 6.3; 6.0; 30.5                  | 0.2; 0.3; 1.2                        | Fail        |                  |
| ... | ...                                | ...                             | ...                                  | ...         |                  |
| 85  | −8.6; 5.4; 31.1                    | −8.3; 5.5; 29.7                 | 0.3; 0.1; 1.4                        | Fail        |                  |
| 86  | −7.8; 6.1; 25.6                    | −7.6; 6.2; 26.5                 | 0.2; 0.1; 0.9                        | Success     |                  |
| 87  | 9.1; −6.4; 24.3                    | 9.4; −6.6; 25.2                 | 0.3; 0.2; 0.9                        | Success     |                  |
| 88  | 7.3; 10.8; 29.5                    | 7.6; 10.6; 30.3                 | 0.3; 0.2; 0.8                        | Success     |                  |
| 89  | 8.9; 9.2; 26.6                     | 9.1; 9.4; 27.5                  | 0.2; 0.2; 0.9                        | Success     |                  |

The laboratory was set up to determine the location of flowers. Here, a flower was attached to a highly accurate angle-measuring device, and the object to be measured was placed in front of a black background to minimise image noise. Experiments involving various scenarios were conducted by adjusting the angle of the flower and its actual

distance from the camera. The maximum value observed on the  $x$ - and  $y$ -axes was 11.2 cm, whereas the maximum value on the  $z$ -axis was 35.7 cm. The results revealed only two instances of pollination failure, where the values on either the  $x$ - or  $y$ -axis exceeded 0.3 cm. In contrast, pollination failure primarily stemmed from errors in the  $z$ -axis values, with all errors exceeding 1 cm occurring at distances greater than 30 cm. Conducting these experiments in a laboratory environment facilitated significant reductions in interference from brightness and complex background noise, resulting in highly practical findings. In fact, the location was pinpointed with an accuracy of up to 86.5%, making this study valuable for practical applications.

**Table 7.** Statistical results for determining the 3D position of a flower in the greenhouse.

| No. | Proposed Method<br>( $x, y, z$ ) (cm) | Ground Truth<br>( $x, y, z$ ) (cm) | Position Error<br>( $dx, dy, dz$ ) (cm) | Pollination | Success Rate<br>(%) |
|-----|---------------------------------------|------------------------------------|-----------------------------------------|-------------|---------------------|
| 1   | 10.3; 5.5; 21.1                       | 10.1; 5.4; 21.8                    | 0.2; 0.1; 0.7                           | Success     | 83.1%               |
| 2   | −6.6; 7.1; 23.3                       | −6.3; 7.0; 23.7                    | 0.3; 0.1; 0.4                           | Success     |                     |
| 3   | −7.2; 7.5; 22.2                       | −7.4; 7.8; 22.8                    | 0.2; 0.3; 0.6                           | Success     |                     |
| 4   | 5.4; −9.0; 31.6                       | 5.4; −9.4; 33.1                    | 0; 0.4; 1.5                             | Fail        |                     |
| 5   | 6.1; 6.3; 29.8                        | 6.3; 6.0; 30.5                     | 0.2; 0.3; 0.7                           | Success     |                     |
| ... | ...                                   | ...                                | ...                                     | ...         |                     |
| 85  | 9.1; −6.4; 37.1                       | 9.4; −6.5; 36.5                    | 0.3; 0.1; 1.6                           | Fail        |                     |
| 86  | 5.8; −8.1; 20.7                       | 5.6; −7.9; 21.1                    | 0.2; 0.2; 0.4                           | Success     |                     |
| 87  | −8.1; −6.8; 21.3                      | −8.0; −6.6;<br>21.9                | 0.1; 0.2; 0.6                           | Success     |                     |
| 88  | 6.2; 13.8; 19.5                       | 5.9; 13.2; 20.2                    | 0.3; 0.7; 0.7                           | Fail        |                     |
| 89  | 9.1; 8.4; 27.8                        | 9.4; 8.2; 26.9                     | 0.3; 0.2; 0.9                           | Success     |                     |

The results obtained from the data in Table 7 reveal the following patterns for the experiments performed in the greenhouse. Errors for the  $x$ - and  $y$ -axes mostly remained within an acceptable range ( $<0.3$  cm) for small values of approximately 10 cm. However, the errors on the  $x$ - and  $y$ -axes also increased as the values gradually increased beyond 10 cm, leading to some instances of pollination failure ( $>0.3$  cm). Similarly, in the case of depth estimation ( $z$  axis), values near 20 cm typically exhibited acceptable errors ( $<1$  cm). Nonetheless, the errors increased significantly for larger depth values of approximately 30 cm. The calculations indicated that approximately 72% of failures could be attributed to high errors in depth values in the majority of pollination failure cases. Despite these errors resulting in some pollination failures, a statistical analysis revealed up to 83.1% of pollination cases were successful. The results in the greenhouse were slightly poorer because it had a slightly more chaotic environment than that of the laboratory. Overall, most cases met the pollination requirements, demonstrating the feasibility of applying this method to determine flower positions in 3D space in a real environment.

## 6. Conclusions

In this study, we proposed a novel approach utilizing threshold-based segmentation methods and advanced image processing techniques to accurately detect the position and orientation of cantaloupe flowers. Additionally, a multi-view imaging method was developed to generate depth maps of the flower images. By integrating this information into a reverse-projection model, we successfully determined the coordinates of the flowers in three-dimensional space. The results of the experiments demonstrated the effectiveness and feasibility of our method compared to existing approaches. Our findings built the way for the development of innovative technologies for smart devices aimed at automating cantaloupe flower pollination. Furthermore, this research lays the groundwork for the advancement of equipment capable of precisely locating targets and conducting accurate pollination tasks. Moreover, this study offers a comprehensive solution to the challenge of transferring cognitive information to robots and automated systems involved in fruit and vegetable pollination, leading to potential cost savings through efficient pollen utilization. Moving forward, our future research will focus on addressing the remaining limitations

and introducing new algorithms to further enhance the accuracy of flower detection based on an AI platform.

**Author Contributions:** Conceptualization, N.T.T.; Methodology, N.D.T. and N.T.T.; Software, N.D.T. and N.M.T.; Validation, N.D.T.; Formal analysis, N.D.T. and N.T.T.; Investigation, N.M.T.; Writing—original draft, N.D.T.; Writing—review & editing, N.T.T.; Visualization, N.M.T.; Supervision, N.M.T. and N.T.T.; Project administration, N.T.T.; Funding acquisition, N.T.T. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work belongs to the project “Research, Design, and Development of the High-Technology Greenhouse” grant No: B2024-KSA-01 funded by Vietnam Ministry of Education and Training and hosted by University of Economics Ho Chi Minh City—UEH, Vietnam.

**Data Availability Statement:** No new data were created or analysed in this study. Data sharing is not applicable to this article.

**Acknowledgments:** This research is funded by University of Economics Ho Chi Minh City-UEH, Vietnam.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Toni, H.C.; Djossa, B.A.; Ayanan MA, T.; Teka, O. Tomato (*Solanum lycopersicum*) pollinators and their effect on fruit set and quality. *J. Hortic. Sci. Biotechnol.* **2021**, *96*, 1–13. [\[CrossRef\]](#)
2. Sáez, A.; Negri, P.; Viel, M.; Aizen, M.A. Pollination efficiency of artificial and bee pollination practices in kiwifruit. *Sci. Hortic.* **2019**, *246*, 1017–1021. [\[CrossRef\]](#)
3. Abdel-Raziq, H.; Petersen, K. Automated Monitoring of Pollinators with Agricultural Robots. In Proceedings of the 2022 8th International Conference on Automation, Robotics and Applications (ICARA), Prague, Czech Republic, 18–20 February 2022; pp. 86–90.
4. Hao, B.; Zhao, J.; Du, H.; Wang, Q.; Yuan, Q.; Zhao, S. A search and rescue robot search method based on flower pollination algorithm and Q-learning fusion algorithm. *PLoS ONE* **2023**, *18*, e0283751. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Potts, S.G.; Neumann, P.; Vaissière, B.; Vereecken, N.J. Robotic bees for crop pollination: Why drones cannot replace biodiversity. *Sci. Total Environ.* **2018**, *642*, 665–667. [\[CrossRef\]](#) [\[PubMed\]](#)
6. Li, G.; Suo, R.; Zhao, G.; Gao, C.; Fu, L.; Shi, F.; Dhupia, J.; Li, R.; Cui, Y. Real-time detection of kiwifruit flower and bud simultaneously in orchard using YOLOv4 for robotic pollination. *Comput. Electron. Agric.* **2022**, *193*, 106641. [\[CrossRef\]](#)
7. Li, K.; Huo, Y.; Liu, Y.; Shi, Y.; He, Z.; Cui, Y. Design of a lightweight robotic arm for kiwifruit pollination. *Comput. Electron. Agric.* **2022**, *198*, 107114. [\[CrossRef\]](#)
8. Gao, C.; He, L.; Fang, W.; Wu, Z.; Jiang, H.; Li, R.; Fu, L. A novel pollination robot for kiwifruit flower based on preferential flowers selection and precisely target. *Comput. Electron. Agric.* **2023**, *207*, 107762. [\[CrossRef\]](#)
9. Yang, C.; Watson, R.M.; Gross, J.N.; Gu, Y. Localization algorithm design and evaluation for an autonomous pollination robot. In Proceedings of the International Meeting of The Satellite Division of the Institute of Navigation 2019, Miami, FL, USA, 16–20 September 2019; pp. 2702–2710.
10. Hiraguri, T.; Kimura, T.; Endo, K.; Ohya, T.; Takanashi, T.; Shimizu, H. Shape classification technology of pollinated tomato flowers for robotic implementation. *Sci. Rep.* **2023**, *13*, 2159. [\[CrossRef\]](#) [\[PubMed\]](#)
11. Williams, H.; Nejati, M.; Hussein, S.; Penhall, N.; Lim, J.Y.; Jones, M.H.; Bell, J.; Ahn, H.S.; Bradley, S.; MacDonald, B.; et al. Autonomous pollination of individual kiwifruit flowers: Toward a robotic kiwifruit pollinator. *J. Field Robot.* **2020**, *37*, 246–262. [\[CrossRef\]](#)
12. Minh Trieu, N.; Thinh, N.T. Quality Classification of Dragon Fruits Based on External Performance Using a Convolutional Neural Network. *Appl. Sci.* **2021**, *11*, 10558. [\[CrossRef\]](#)
13. Mu, X.; He, L.; Heinemann, P.; Schupp, J.; Karkee, M. Mask R-CNN based apple flower detection and king flower identification for precision pollination. *Smart Agric. Technol.* **2023**, *4*, 100151. [\[CrossRef\]](#)
14. Mao, W.; Liu, H.; Hao, W.; Yang, F.; Liu, Z. Development of a combined orchard harvesting robot navigation system. *Remote Sens.* **2022**, *14*, 675. [\[CrossRef\]](#)
15. Jun, J.; Kim, J.; Seol, J.; Kim, J.; Son, H.I. Towards an efficient tomato harvesting robot: 3d perception, manipulation, and end-effector. *IEEE Access* **2021**, *9*, 17631–17640. [\[CrossRef\]](#)
16. Arad, B.; Balendonck, J.; Barth, R.; Ben-Shahar, O.; Edan, Y.; Hellström, T.; Hemming, J.; Kurtser, P.; Ringdahl, O.; Tielen, T.; et al. Development of a sweet pepper harvesting robot. *J. Field Robot.* **2020**, *37*, 1027–1039. [\[CrossRef\]](#)
17. Afshari, H.H.; Gadsden, S.A.; Habibi, S. Gaussian filters for parameter and state estimation: A general review of theory and recent trends. *Signal Process.* **2017**, *135*, 218–238. [\[CrossRef\]](#)
18. Zhang, Y.; Tong, R.; Song, D.; Yan, X.; Lin, L.; Wu, J. Joined fragment segmentation for fractured bones using GPU-accelerated shape-preserving erosion and dilation. *Med. Biol. Eng. Comput.* **2020**, *58*, 155–170. [\[CrossRef\]](#)

19. Zhou, G.; Hu, Z.; Chen, X.; Liu, Q. Direct ellipse fitting by minimizing the L0 algebraic distance. In Proceedings of the 2023 3rd International Conference on Consumer Electronics and Computer Engineering (ICCECE), Guangzhou, China, 6–8 January 2023; pp. 175–180.
20. Mertan, A.; Duff, D.J.; Unal, G. Single image depth estimation: An overview. *Digit. Signal Process.* **2022**, *123*, 103441. [[CrossRef](#)]
21. Hou, Y.; Liu, C.; An, B.; Liu, Y. Stereo matching algorithm based on improved Census transform and texture filtering. *Optik* **2022**, *249*, 168186. [[CrossRef](#)]
22. Ma, G.; Yue, X. An improved whale optimization algorithm based on multilevel threshold image segmentation using the Otsu method. *Eng. Appl. Artif. Intell.* **2022**, *113*, 104960. [[CrossRef](#)]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.