Article

Occluded Apple Fruit Detection and Localization with a Frustum-Based Point-Cloud-Processing Approach for Robotic Harvesting

Tao Li ¹ , Qingchun Feng ¹ , Quan Qiu ² , Feng Xie ^{1,3} and Chunjiang Zhao ^{4,*}

¹ Intelligent Equipment Research Center, Beijing Academy of Agriculture and Forestry Sciences, Beijing 100097, China; lit@nercita.org.cn (T.L.); fengqc@nercita.org.cn (Q.F.); 2111916009@stmail.uj.edu.cn (F.X.)

² BIPT Academy of Artificial Intelligence, Beijing Institute of Petrochemical Technology, Beijing 102617, China; qiuquan0110@ustc.edu

³ School of Agricultural Engineering, Jiangsu University, Zhenjiang 212013, China

⁴ National Engineering Research Center for Information Technology in Agriculture, Beijing 100097, China

* Correspondence: zhaojc@nercita.org.cn

Abstract: Precise localization of occluded fruits is crucial and challenging for robotic harvesting in orchards. Occlusions from leaves, branches, and other fruits make the point cloud acquired from Red Green Blue Depth (RGBD) cameras incomplete. Moreover, an insufficient filling rate and noise on depth images of RGBD cameras usually happen in the shade from occlusions, leading to the distortion and fragmentation of the point cloud. These challenges bring difficulties to position locating and size estimation of fruit for robotic harvesting. In this paper, a novel 3D fruit localization method is proposed based on a deep learning segmentation network and a new frustum-based point-cloud-processing method. A one-stage deep learning segmentation network is presented to locate apple fruits on RGB images. With the outputs of masks and 2D bounding boxes, a 3D viewing frustum was constructed to estimate the depth of the fruit center. By the estimation of centroid coordinates, a position and size estimation approach is proposed for partially occluded fruits to determine the approaching pose for robotic grippers. Experiments in orchards were performed, and the results demonstrated the effectiveness of the proposed method. According to 300 testing samples, with the proposed method, the median error and mean error of fruits' locations can be reduced by 59% and 43%, compared to the conventional method. Furthermore, the approaching direction vectors can be correctly estimated.

Keywords: agricultural robot; deep learning; fruit detection; point cloud; apple-harvesting robot; RGBD camera

Citation: Li, T.; Feng, Q.; Qiu, Q.; Xie, F.; Zhao, C. Occluded Apple Fruit Detection and Localization with a Frustum-Based Point-Cloud-Processing Approach for Robotic Harvesting. *Remote Sens.* **2022**, *14*, 482. <https://doi.org/10.3390/rs14030482>

Academic Editor: Ming-Der Yang

Received: 20 December 2021

Accepted: 17 January 2022

Published: 20 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 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

In the fresh fruit industry, harvest requires plenty of labor force, which usually undergoes a seasonal shortage. According to the existing literature [1], the cost of labor in the process of harvest constitutes over 50% of the total cost in apple orchards. To reduce the labor cost, robotic harvesters have been widely investigated over the past few decades. For a harvesting robot, locating fruits is one of the most challenging tasks of robotic perception in the complicated orchard environment. Visual sensors offer abundant information about the environment for robots, and in particular, Red Green Blue Depth (RGBD) cameras have pushed the boundaries of robot perception significantly and have been viewed as a promising technique. With the advantages of low cost, lightweight, and mini-type, the RGBD camera has become an essential component in agricultural robots, as well as in industrial robots and has attracted increasing attention.

One primary task of robotic harvesting is the recognition and localization of fruits. In orchards, the environmental factors affect the accuracy and robustness of stereo perception

with RGBD cameras. Densely arranged leaves and branches in front of fruits are very common in robotic harvesting, bringing difficulties to accurately locating and approaching fruits. Moreover, illumination conditions also make the appearances of fruits very different from morning to night. In this case, taking these working conditions of harvesting robots into account is of great significance. In other words, fruit-harvesting robots are required to adapt to the environment and to understand it to increase the rate of grasping success. In previous literature, fruit recognition, segmentation, and pose estimation have been extensively studied. The recognition and localization of fruits can be successfully implemented in the case of no occlusion or a small occlusion. In the field operation of orchards, a robotic harvester has to cope with many fruits with complicated situations of occlusion. In some cases, the centers of fruits usually are occluded by leaves or branches. Then, the bounding boxes detected and the corresponding depth measurements tend to be error-prone, affected by the obstacles, which results in the failures of grasps. Therefore, further development of the techniques to detect and locate the fruits with complicated occlusions are demanded.

In this work, a new method of fruit localization is proposed for robotic harvesting. The contributions can be summarized as follows:

- An instance segmentation network was employed to provide the location and geometry information of fruits in four cases of occlusions including leaf-occluded, branch-occluded, fruit-occluded, and non-occluded;
- A point-cloud-processing pipeline is presented to refine the estimations of the 3D positional and pose information of partially occluded fruits and to provide 3D approaching pose vectors for the robotic gripper.

The rest of the paper is organized as follows: Section 2 introduces the related works on the apple detection and localization algorithm. Section 3 compares different instance segmentation networks for fruit recognition and localization on the image-plane. Moreover, the point-cloud-processing method for the 3D localization of partially occluded fruits is introduced in this section. Section 4 gives the field experimental results and further discussions. Section 5 concludes the paper and provides future works.

2. Related Work

2.1. Fruit Recognition

The methods of fruit recognition can be divided into three kinds: single-feature-based, multi-feature-based, and deep-convolutional-neural-network-based. In a single-feature-based method, the differences among some shallow features to identify and detect fruit targets are employed, such as the color difference method with Otsu adaptive threshold segmentation, the Hough transform-based circle detection method, the optical flow method, etc. To improve robustness in the outdoors with variable lighting conditions and occlusion, the multiple-feature fusion method is proposed, which usually uses a combination of the Otsu adaptive threshold of colors, image morphology, and Support Vector Machine (SVM) to extract the areas of apple targets. In [2], fruit regions were segmented by combining red/green chromatic mapping and Otsu thresholding in the RGB color space, and a local binary pattern operator was employed to encode fruit areas to form a histogram for the classification by the SVM network. In [3], the fruits were segmented by a region-growing method, and then, the color and texture characteristics of the growing region were used for the classification and segmentation. Compared with the single-feature method, the core idea of the fruit-detection method based on multi-feature fusion is to encode pixels or local features of pixels to form descriptors and similarity measures and to select potential fruit regions based on super-pixels and region fusion algorithms. Then, region features are extracted as the inputs of the classifiers to complete region classification. Such methods can adapt to occlusion and changes in the external environment to a certain extent. However, the extracted features are designed artificially, and the process of feature extraction adopts an unsupervised way, which makes the selected features lack representativeness to distinguish various categories. In addition, the features contained in feature descriptors are usually

shallow features such as color, edge, and distance and lack advanced features such as spatial position and pixel correlation.

As deep learning theory has developed, the convolutional neural networks have become capable of extracting abstractly deep features to particular categories, which makes them qualified for complicated recognition tasks. Currently, the deep convolutional neural network (DCNN) has become the mainstream of fruit recognition methods. Many network models, i.e., YOLOv3 [4], LedNet [5], Faster RCNN [6,7], Mask RCNN [8], YOLACT [9], and YOLACT-edge [10], were successfully used in the recognition of apple fruits, which are summarized in Table 1.

Table 1. The summary of deep learning methods of apple detection.

Networks Model	Precision (%)	Recall (%)	mAP (%)	F1-Score (%)	Reference
Improved YOLOv3	97	90	87.71	—	[4]
LedNet	85.3	82.1	82.6	83.4	[5]
Improved R-FCN	95.1	85.7	—	90.2	[11]
Mask RCNN	85.7	90.6	—	88.1	[8]
DaSNet-v2	87.3	86.8	88	87.3	[12]
Faster RCNN	—	—	82.4	86	[13]
Improved YOLOV5s	83.83	91.48	86.75	87.49	[14]

mAP represents mean Average Precision; F1-score means the balanced F-score $F_1 = 2 \cdot \text{precision} \times \text{recall} / (\text{precision} + \text{recall})$.

In light of the DCNN-based method being easily adapted to multiple kinds of fruits and having the capability of generalization to complicated natural environments, in this paper, we employed the DCNN technique to propose the fruit recognition algorithm.

2.2. 3D Fruit Localization and Approaching Direction Estimation

In the visual sensing system of an agricultural robot, 3D localization refers to the process of extracting the 3D information of the target by visual sensors. The main task of 3D localization for a robotic harvester is to obtain the spatial coordinates of the fruit to further guide the robot's gripper approaching the targets. To accomplish such a task of targets, stereo cameras usually are employed, e.g., binocular vision systems based on optical geometry and consumer RGBD sensors based on Structured Light (SL), the Time-of-Flight (ToF) method, and Active Infrared Stereo (AIRS).

Based on the triangulation optical measurement principle [15], binocular cameras have been successfully used to identify tomatoes, sweet peppers, and apples [16–19]. Reference [20] developed a vision sensor system that used an over-the-row platform for apple crop load estimation. The platform acquires images from opposite sides of apple trees and identifies the targets on the images based on shape and color, then maps the detected apples from both sides together in a 3D space to avoid duplicate counting. The above optical-geometry-based stereo-vision system is cost-effective, but its accuracy of measurement and timeliness are usually limited in real applications [21], as the sensing modalities and algorithms have become sophisticated and time-consuming.

Consumer RGBD cameras have a simple and compact structure and can be used for many local tasks, such as the three-dimensional reconstruction of targets at specific locations. RGBD cameras, e.g., Microsoft Kinect V2 (ToF-based) [22–24] and Intel Realsense (AIRS-based) [12,25], have been widely used in harvesting robots with the merits of low cost, high measuring resolution of distance, adaptiveness to ambient light, and quick response. The technique of fruits' localization with RGBD cameras has been widely investigated in the field of harvesting robots. Reference [26] employed the depth map of the Fotonic F80 depth camera from the detected sweet pepper regions and transformed the region to the 3D location of the mass center to obtain the locations of fruits. Reference [27] employed the point cloud of apple trees of Kinect v2 by fusing the RGB information and depth information, then segmenting the fruit regions by the point-cloud-segmentation method of the ROI in the RGB images and achieved a segmented purity of 96.7% and 96.2% for

red and green apples. Reference [28] addressed the 3D pose estimation of peppers using Kinect Fusion technology from the Intel Realsense F200 RGBD camera to fit a superellipsoid from peppers' point clouds through a constrained non-linear least-squares optimization for the estimation of sweet pepper pose and grasp pose. In their follow-up work [29], the estimated pose of each sweet pepper was chosen from multiple candidates' poses during the harvesting by a utility function. In [30], tomatoes were fit by Random Sample Consensus (RANSAC), and the robotic manipulator grasped the tomatoes according to the fit centroid of the sphere. Likewise, Reference [31] used the RANSAC-based fitting method to model a guava fruit and an FCN network to localize a stem for robotic arm grasping. Reference [32] used an improved 3D descriptor (Color-FPFH) with a fusion of color features and 3D geometry features of the point clouds generated from RGBD cameras. Then, by the 3D descriptors and the classifiers optimized by support vector machine and the genetic algorithm, the objects of apples, branches, and leaves were divided. In [33], the 3D sphere Hough transform method was employed to model the apple fruits to compute the grasp pose of each fruit based on the fit sphere.

All aforementioned investigations usually assumed that the point clouds include an ideal surface of targets to employ a 3D descriptor and perform a fitting algorithm. Mostly, however, it is hard to acquire ideal point clouds of objects due to the unsatisfactory performance of the depth sensor, which is sensitive to external disturbance and prone to lose the filling rate of depth maps in outdoor conditions. However, there is little literature to address the locating problem in unsatisfactory point clouds, which are common in a real application.

3. Methods and Materials

The proposed detection and localization method for occluded apple fruits is based on deep learning and a point-cloud-processing algorithm. It is schematically shown in Figure 1. The flow of the proposed method comprises the acquisition of the sensing images of fruits, apple fruit recognition, fruit central line and frustum proposal, the generation of the point cloud of the visible parts of apples, and the determination of the centroid, size, and pose. Details of each step are presented in the following subsections.

Figure 1. The overall workflow of the proposed method for occluded apple localization.

3.1. Hardware and Software Platform

The implementation of the proposed method was based on the hardware platform, as shown in Figure 2a–c, comprising the following devices:

- Computing module: Intel NUC11 Enthusiast with CPU: Intel Core™ i7-1165G7 processor; GPU: Nvidia Geforce RTX 2060;
- RGBD camera: Intel Realsense D435i with a global shutter, ideal range: 0.3 m to 3 m, depth accuracy: <2% at 2 m, depth Field of View (FOV): $87^\circ \times 58^\circ$;
- Tracked mobile platform with autonomous navigation system: Global Navigation Satellite System (GNSS) + a laser scanner; electric drive: DC 48 V, two 650 W motors; working speed: 0–5 km/h; carrying capacity: 80 kg; size: 1150 × 750 × 650 mm (length × width × height);
- Robotic arm: Franka Emika 7-dof robotic arm;
- LiDAR: A Velodyne HDL-64E LiDAR is for the true value acquisition of fruits' 3D position coordinates.

The overall control system was implemented on the Robot Operation System (ROS) Melodic, with the operation system being Linux Ubuntu 18.04. The ROS accounts for robotic hardware abstraction, low-level devices' (e.g., cameras, robotic arm) control, the implementation of commonly used functionalities (e.g., motion planning), message-passing between processes, and package management. RGBD image acquisition was driven by Realsense SDK v2.40.0 with the ROS package. The training and real-time inference of the deep learning system were performed with the acceleration of CUDA and CUDNN with the Pytorch framework. The processing of the point cloud and computer vision were based on Open3D and OpenCV, respectively. The robotic arm control was implemented by the MoveIt motion-planning framework and franka_ros.

Figure 2. Hardware platform of the robotic system in the orchard experiments. (a–c) are the hardware platform of the robotic system; (d) is the verification platform with a Realsense D435i and a 64-line LiDAR to acquire the true values of fruits.

3.2. Data Preparation

To train the segmentation network for the proposed apple localization algorithm, two kinds of image datasets were employed in this work. One was the open-source MinneApple dataset [34], which contains 1000 images with over 41,000 labeled instances of apples. The other dataset (RGBD apple dataset) comprises over 600 images with over 4800 labeled instances of apples in 2 modern standard orchards in Haidian District and Changping District, Beijing, China, acquired by the Realsense D435i RGBD camera. Various conditions were considered in the preparation, as shown in Figure 3, including:

- Different illumination: front lighting, side lighting, back lighting, cloudy;
- Different occlusions: non-occluded, leaf-occluded, branch/wire-occluded, fruit-occluded;

- Different periods of the day: morning (08:00–11:00), noon (11:00–14:00), afternoon (14:00–18:00).

Figure 3. The apple fruits under different illuminations and occlusions.

The dataset was split into 4:1, for the training and validation sets, respectively. In the collected training set, there were 50.97%, 33.49%, 7.95%, and 7.59% fruits of non-occlusion, leaf-occlusion, branch/wire-occlusion, and fruit-occlusion, respectively. In the validation sets, there were 42.47%, 35.67%, 14.01%, and 7.86% fruits of the four occlusions, respectively.

To prevent overfitting, we employed the following methods to augment the dataset: (a) photometric distortion including random contrast, random saturation, random hue, and random brightness, (b) random mirror operation, (c) random sample crop, (d) random flip, and (e) random expand.

3.3. Image Fruit Detection and Instance Segmentation

To select the instance segmentation network for our task, 4 network models were compared: Mask-RCNN, Mask Scoring RCNN (MS RCNN), YOLACT, and YOLACT++. We employed pre-trained models to implement transfer learning in this work. The comparison results are given in Table 2.

Table 2. Performance comparisons of different networks on apple instance segmentation.

Average Precision (AP)										
Network Model	Backbone	Non-Occlusion		Leaf-Occlusion		Branch-Occlusion		Fruit-Occlusion		FPS
		Bbox	Mask	Bbox	Mask	Bbox	Mask	Bbox	Mask	
Mask	ResNet-50	38.14	40.12	29.56	28.14	9.1	6.14	13.2	8.85	17.3
RCNN	ResNet-101	38.39	39.62	25.13	25.96	7.61	5.23	9.96	7.88	14.5
MS	ResNet-50	38	38.12	27.04	25.27	4.88	6.36	9.03	8.29	17.1
RCNN	ResNet-101	39.09	40.66	27.98	24.97	7.52	7.05	10.99	10.36	13.6
YOLACT	ResNet-50	43.53	44.27	26.29	26.08	16.67	13.35	15.33	14.61	35.2
	ResNet-101	39.85	41.48	28.07	27.56	11.61	11.31	21.22	21.88	34.3
YOLACT++	ResNet-50	42.62	44.03	32.06	35.49	18.17	13.8	21.48	20.28	31.1
	ResNet-101	43.98	45.06	34.23	36.17	15.46	13.49	11.52	13.33	29.6

From the comparison results, it can be seen that the YOLACT++ network with ResNet-101 had better FPS performance than Mask RCNN and MS RCNN due to its network structure. Besides, due to the use of deformable convolutional networks, YOLACT++ can tune the size and shape of the convolution kernel adaptively according to the learning rate. Consequently, YOLACT++ (ResNet-101) showed better Average Precision (AP) performance than the counterparts. The comparative test results of the networks are given in Figures 4–6. From Figure 4, it can be seen that ResNet-50 output an incorrect bounding box in the dotted red circle. In contrast, ResNet-101 correctly detected and segmented all fruits. In Figures 5 and 6, it can be seen that ResNet-50 had missing detections, compared with ResNet-101. Based on the above analysis, we chose YOLACT++ (ResNet-101) as the instance segmentation network for apples in this work, and some test results are demonstrated in Figure 7.

Figure 4. Performance of the detection and segmentation of YOLACT++ network in cloudy conditions. (a) ResNet-50; (b) ResNet-101.

Figure 5. Performance of the detection and segmentation of YOLACT++ network in front light in sunny conditions. (a) ResNet-50; (b) ResNet-101.

Figure 6. Performance of the detection and segmentation of YOLACT++ network in back lighting in sunny conditions. (a) ResNet-50; (b) ResNet-101.

Besides, we employed the hybrid strategy, which switches from Adam to Stochastic Gradient Descent (SGD) optimization [35] to achieve better generalization and to prevent overfitting. By the combination of Adam and SGD, a balance between convergence speed and learning performance can be obtained. Moreover, we froze the BN layer during fine-tuning to prevent large value changes of Gamma and Beta.

Figure 7. Detection results of YOLACT++ (ResNet-101) under different illuminations and occlusions. The numbers on the bounding boxes denote the classes of the targets, Class 1: non-occluded, Class 2: leaf-occluded, Class 3: branch/wire-occluded, Class 4: fruit-occluded.

In practical application, it can be very common that fruits are partially occluded by branches, leaves, and other fruit. Such a situation brings difficulties in point cloud processing. Moreover, the sensitivities of RGBD cameras to an external disturbance in the outdoor conditions result in noise and a low fill rate of the depth maps. This also leads to the low quality of point clouds generated from RGBD images. For instance, based on our previous experimental point cloud data acquired in real applications (see Figure 8), the point clouds of the partially occluded fruits were usually fragmentary and could not be used for the 3D reconstruction of the targets, making the apples' location hard to determine. From Figure 8, it can be observed that both Target A and Target B had a poor filling rate of depth to different extents, which can be seen in two different views. Moreover, point clouds on the surface of Target A and Target B were not spherical, leading to difficulties in morphological feature analysis.

To address this problem, we now propose a pipeline for fruit high-precision localization with occlusions.

3.4. Fruit Central Line and Frustum Proposal

Determining the fruit's center is a fundamental problem for a robotic harvest. In practice, however, due to the characteristics of sensors and the inaccuracy of mask segmentation, there are deficiencies and outliers in the point clouds, resulting in that the acquired point clouds are fragmentary and distorted; see Figure 9. In this case, the point cloud inside of the bounding box cannot represent the whole fruit, and the inference of centroid based on such point clouds might be inaccurate. Many conventional methods directly use point clouds inside of the 2D bounding box to calculate the centroid of fruits, leading to a large location error. To address this issue, we propose a frustum-based method to determine the centroid. Before introducing the frustum, it is necessary to extract the central line of fruits. We used 2D bounding boxes obtained from the RCNN network to propose a complete fruit region along with a center on the RGB image. For the quasi-symmetry of spherical fruits, the

center of the bounding box overlaps the center of the fruit. Using such a property, we may determine the center's position on the x -axis and y -axis in the frame of an RGBD camera.

Figure 8. Distortion and fragments of the point clouds of the partially occluded fruits (Target A and Target B).

Figure 9. Distortion and fragments of the targets' point clouds. (a) Non-occluded fruits' point clouds with little distortion and good completeness; (b) leaf-occluded fruits' point clouds with considerable distortion and good completeness; (c) leaf-occluded fruits' point clouds with considerable distortion and fragments.

To further obtain the coordinate on the z -axis, the fruits' bounding boxes on the RGB image can be lifted to a frustum (with near and far planes specified by the depth range in the 2D bounding box on the depth image) and a 3D central line (starting at the origin of the camera frame and ending at the center of the frustum's far plane), by using the RGB camera's intrinsic parameter matrix and the aligned depth image, as shown in Figure 10.

Figure 10. 2D bounding boxes on RGB images and their corresponding point cloud frustums.

3.5. Point Cloud Generation of Visible Parts of Occluded Apples

After obtaining the frustum and the central line, it is necessary to further determine the specific position of the fruit's center on the line. As shown in Figure 11, there are point clouds of non-fruit in the frustum, such as the background, leaves, branches, etc. It is necessary to further filter the point cloud belonging to the fruit from objects of non-fruit. To this end, there are two steps to be performed.

Figure 11. The point cloud in a frustum of partially occluded fruits, including point clouds of fruits (in blue), leaves, the background, and noise. View 1 and View 2 are two different angles of view of the point cloud: View 1 (left front); View 2 (front).

1. Generation of point clouds under the fruits' masks. According to the fruits' masks detected by the RCNN network and combining their corresponding depth map, the point clouds inside the masks can be generated, as shown in Figure 12. Ideally, the point cloud(s) generated from the fruit's mask is (are) supposed to be distributed on the surface of the fruit sphere. In practice, however, due to the nonideal characteristics of sensors and the inaccuracy of mask segmentation, there are deficiencies and outliers in the point clouds, resulting in the acquired point clouds being fragmentary and distorted; see Figure 9. Therefore, the following step is necessary;
2. Selection of the most likely point cloud. To sort out the cluster on the target's surface, we used Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to cluster the target point cloud. By DBSCAN, the point clouds generated from the masks can be clustered. Then, through counting how many points each cluster holds, the point cloud holding more points is selected as the best cluster. Taking the cluster as the most likely point cloud belonging to the target's surface, we may remove the other point clouds in the frustum.

Figure 12. The generating process of point clouds and frustum from RGBD images.

The reasons for choosing DBSCAN in this work lied in the following aspects: (1) DBSCAN does not need to know the number of clusters in the point clouds a priori, as opposed to k-means; (2) DBSCAN is robust to noise in point clouds; (3) There are only two parameters needed in DBSCAN, and it is insensitive to the order of points in the database, which suits clustering point clouds; (4) In the orchard sensing tasks, the parameters of DBSCAN, a minimum number of points (minPts), and the neighborhood (ϵ) can be easily determined according to artificial experience.

3.6. Centroid Determination and Pose Estimation

By the above two steps, the outliers can be removed and the target point cloud on the surface of the fruit obtained. Then, determining the fruit's center is necessary. Due to the distortion of the point clouds, as shown in Figure 9, fitting a sphere (such as by RANSAC and 3D Hough transform) is not feasible in this case. Consequently, the estimations of the center and the approaching vector of fruits need the following steps:

1. Obtaining the centroid of the filtered point cloud, denoted by p_c ;
2. Calculating the radius from the 2D bounding box of the target and camera intrinsic parameters by:

$$r = \frac{K_u \Delta u}{2z'}, \quad (1)$$

where $\Delta u = u_r - u_l$ is the width of the bounding box and u_l, u_r are the pixel positions of the left side and the right side of the bounding box on the U -axis, respectively; z' are the z -axis coordinate of the point p_c ; r denotes the radius of the sphere; K_u is the scale factor on the U -axis of the camera;

3. Constructing a sphere with radius r by taking p_c as a spherical center, two points of intersection between the sphere and the central line of the frustum were obtained;
4. Taking the far point of the above two as the fruit's centroid p_o ;
5. Taking the direction vector from p_o to p_c as the approaching vector.

4. Experimental Results and Discussion

4.1. Results of Localization and Approaching Vector Estimation

To verify the performance of the proposed method, experiments in an orchard were conducted. We set up a reference system with an RGBD camera and a LiDAR, aiming at providing the ground truth positions of fruit in order to evaluate the performance of the proposed method, as shown in Figure 2d. Such a reference system is not essential in the implementation of robotic harvest with the proposed method. Consequently, the standalone reference system with the LiDAR and Realsense camera was used. To extract

the true values of the positions of a fruit's surface, the positional measurements of the LiDAR were employed due to its high resolution of distance and robustness to sunlight.

Once having obtained the point clouds from the LiDAR, we manually determined the centroid's position of fruit according to its point cloud. Besides, the true values of fruits' sizes were calculated by the centroid's positions and the image pixel positions, according to $r = K_u \Delta u / 2z_c$, where z_c is the true value of the center's position on the z-axis.

We extracted 50 frames of RGBD image pairs of the robotic sensing system and the corresponding point cloud generated from the LiDAR as well, where the target fruits belonged to six different apple trees.

In the experiments, the travel speed of the tracked mobile platform to acquire images was 0.3 m/s along the fruit tree line and the approaching speed toward fruits of the RGBD camera was 0.3 m/s. The total pipeline processing speed of the proposed method was 0.1–0.12 s per frame, namely around 10 Frames Per Second (FPS) on our hardware system, which is competent to perform vision-based harvesting robot control.

We conducted three groups of tests at the following distances: <0.6 m (Group 1), 0.6–0.9 m (Group 2), and >0.9 m (Group 3) from the row of trees and at the view of the left, middle, and right of the target tree, respectively, as shown in Figure 13. In these tests, the numbers of the four different conditions of occlusions are as follows: non-occlusion: 91 (30.33%), leaf-occlusion: 171 (56.99%), branch/wire-occlusion: 24 (8%), and fruit-occlusion: 14 (4.67%).

Figure 13. The fruits of the comparison tests at three different distances and views.

To verify the effectiveness of the proposed method, we took the bounding-box-based (bbx mtd.) method [36] as the reference scheme.

Table 3 presents the estimating errors of the center and radius with the proposed method (our mtd.) and the bounding-box-based method (bbx mtd.) in the different testing groups and the total amount. Moreover, the errors included the Maximum error (Max. error), Minimum error (Min. error), Median error (Med. error), Mean error (Mean error), and Standard error (Std. error). From Table 3, it can be observed that the values of the median error, mean error, and standard error of the centers by the proposed method were considerably reduced by 67%, 50%, and 10% in Group 1 and 59%, 43%, and 9% in total. The median error, mean error, and standard error of the radius by the proposed method were reduced by 75%, 80%, and 96% in Group 1 and 70%, 70%, and 78% in total. This demonstrated that the proposed method can reduce the errors of both the center localization and the fruit size estimation.

Table 3. The experimental results of the locating performance with the proposed method and the bounding-box-based method.

		Group 1		Group 2		Group 3		Total	
		bbx mtd.	our mtd.	bbx mtd.	our mtd.	bbx mtd.	our mtd.	bbx mtd.	our mtd.
Center	Max. error (mm)	49.65	47.24	49.01	49.07	48.46	42.80	49.65	49.07
	Min. error (mm)	2.24	0.05	4.79	0.19	0.67	1.65	0.67	0.05
	Med. error (mm)	17.16	5.69	24.43	8.25	21.44	14.94	19.77	8.03
	Mean error (mm)	18.36	9.15	24.43	13.73	22.70	17.74	21.51	12.36
	Std. error (mm)	11.09	9.99	13.55	13.10	13.15	10.38	12.74	11.59
Radius	Max. error (mm)	14.47	1.21	27.11	6.72	14.84	5.44	27.11	6.72
	Min. error (mm)	0.01	0.54	0.08	1.18	0.14	0.26	0.01	0.26
	Med. error (mm)	4.01	0.98	3.40	1.37	4.26	2.22	3.97	1.18
	Mean error (mm)	4.78	0.96	4.77	1.48	4.61	2.41	4.74	1.43
	Std. error (mm)	3.20	0.14	4.89	0.69	3.83	1.04	3.90	0.84

To clarify the relationship between distances and center/radius error, Figures 14 and 15 are presented, where the x -axis denotes the errors divided into five groups and the y -axis denotes the number of samples accordingly. From the comparisons of the distributions between the bounding-box-based method (Figure 15a,b) and the proposed method (Figure 15c,d), the majority of testing samples were distributed at the section of 0–10 mm/0–5 mm in the proposed method, outperforming the bounding-box-based method. In Figure 14c,d, one may see that the locating and estimating precisions were significantly affected by the sensing distances, denoted by the bars in different colors. This means that the density of the point clouds was one of the key factors of the performance.

Figure 14. Quantities' distributions of the center errors and radius errors with different sensing distances. (a,b) are with the bounding-box-based method; (c,d) are with the proposed method.

Figure 15. Quantities' distributions of the center errors and radius errors with the proposed method and the bounding-box-based method. (a,b) are the results of Group 1; (c,d) are the results of Group 2; (e,f) are the results of Group 3.

Intuitively speaking, the abundant information of the RGBD pixels of the fruit improved the accuracy of location, which can be seen in Figure 15. The nearer sensing distance offered higher accuracy, but lost sensing range, and such a trade-off has to be made. Besides, Figures 16 and 17 demonstrate some experimental results of localization and approaching vector estimations. In Figure 17, the sphere-fitting-based methods, including RANSAC [30,31], 3D descriptor (Color-FPFH) [32], and 3D Hough transform [33], failed to generate accurate approach vectors because these sphere-fitting-based methods failed to extract the position of the fruits' surface accurately. In contrast, the success rate of detachment with the proposed method increased significantly. This demonstrated that the derived approach vectors by the proposed method can achieve better robustness on the point cloud without an ideal sphere surface of fruits.

Figure 16. The fit spheres of fruits and their approaching vectors at 800 mm.

Figure 17. The localizations and estimations of approaching vectors of fruits with the proposed method. (a–d) demonstrate four samples of experimental results with different apple trees.

4.2. Discussion

In the experiments in Figures 14 and 15, it can be observed that the proposed method showed good accuracy of localization in different distances. In the 3D-bounding-box-based method, the minimum and maximum positions on each axis of all points inside the 3D bounding box were employed to construct the edges. As shown in Figure 16, the front face of the cube had a large offset on the z-axis due to the leaf. This illustrates that the 3D-bounding-box-based method is sensitive to the outliers of the fruits' point clouds.

It was found that the performance of fruit location accuracy also depended on the accuracy of central line extraction based on 2D bounding boxes and 2D masks in our experiments. Namely, the detection accuracy had an influence on the centroid positioning along the XOY-plane; while the mask affected the performance on the z-axis. Consequently,

the accuracy of detection and segmentation on 2D images is still a prerequisite. Some excellent studies provided potential solutions to this issue. For instance, neural rendering [37] and the Cycle-Generative Adversarial Network (C-GAN) [38] have been employed to restore occluded or unseen plant organs in the field of phenotypic data robotic collection and automatic fruit yield estimation. Besides, the symmetry-based 3D-shape-completion method [39] has been successfully applied to locate strawberries with the incomplete visible part of the targets and showed good effectiveness. To completely solve the problem of occluded fruit detection, multiple techniques are expected, which is also our research interest in the future.

Moreover, the experimental results showed that the filling rate of the depth data underwent a degradation in the back lighting condition, due to the influence of RGBD sensor performance. By the proposed method, the localization accuracy under good illumination conditions outperformed the case with the back lighting illumination. In practical applications, the outdoor filling rate of the RGBD image is still the main limitation to be solved in the future by the development of sensor technology. On the other hand, the improvement obtained in this paper was on the precision of localization and 3D approaching pose. Such an improvement was at the cost of the processing rate, compared with the conventional image-segmentation-based fruit localization algorithm, which is usually 25 FPS. However, compared with a harvesting cycle (around 6 s in our case), such a time delay can be neglected and the success rate is much more important than such a tiny delay for the overall harvesting time.

5. Conclusions

This paper investigated the problem of apple fruits' localization for harvesting robots under occluded conditions. A network for apple fruit target instance segmentation and a frustum-based processing pipeline for point clouds generated from RGBD images were proposed in this paper. To segment the fruit part from occluding objects with partial occlusions, YOLACT++ (ResNet-101) was used to improve the accuracy of bounding boxes' detection and masks' estimation in the case of partial occlusion. To address the issue of the low fill rate and noise of the depth image in an outdoor environment, we constructed a frustum by the bounding box and extracted the central line of the fruits. After that, to generate the point clouds of fruits, we used the segmented masks of fruits on RGBD images. Based on the central line and point cloud of fruits, DBSCAN was used to search the points that belong to the target fruit in the distorted and fragmentary point clouds. Meanwhile, the fruit's radius and the central position of the target fruit can be estimated from the point of intersection accordingly. Moreover, the proposed method provided the approaching direction vector for robots, calculated from the center of the fruit to the mass center of the obtained cluster.

The experimental results in orchards showed that the proposed method improved the performance of fruits' localization in the case of partial occlusion. The superiority of the proposed method can be summarized as follows: (a) robustness to the low fill rate and noise of the depth map in an outdoor environment; (b) accuracy when a partial occlusion exists; (c) providing the approaching direction for the reference of the robotic gripper to pick. According to 300 testing samples, the values of the center locations' median error, mean error, and standard error could be considerably reduced by 59%, 43%, and 9% in total with the proposed method, and the approaching direction vectors could be correctly estimated.

Author Contributions: Conceptualization, T.L.; methodology, T.L.; software, T.L. and F.X.; validation, T.L. and F.X.; formal analysis, T.L.; investigation, T.L.; resources, Q.F. and Q.Q.; data curation, F.X.; writing—original draft preparation, T.L.; writing—review and editing, T.L. and Q.Q.; visualization, F.X.; supervision, C.Z.; project administration, Q.F.; funding acquisition, C.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Beijing Science and Technology Plan Project (Z2011000080-20009), the China Postdoctoral Science Foundation (2020M680445), the Construction Project of Beijing Key Laboratory of Agricultural Intelligent Equipment Technology in 2021 (PT2021-15), and the Postdoctoral Science Foundation of Beijing Academy of Agriculture and Forestry Sciences of China (2020-ZZ-001).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, Z.; Heinemann, P.H. Economic analysis of a low-cost apple harvest-assist unit. *HortTechnology* **2017**, *27*, 240–247. [\[CrossRef\]](#)
2. Zhuang, J.; Hou, C.; Tang, Y.; He, Y.; Luo, S. Computer vision-based localisation of picking points for automatic litchi harvesting applications towards natural scenarios. *Biosyst. Eng.* **2019**, *187*, 1–20. [\[CrossRef\]](#)
3. Ji, W.; Zhao, D.; Cheng, F.; Xu, B.; Zhang, Y.; Wang, J. Automatic recognition vision system guided for apple harvesting robot. *Comput. Electr. Eng.* **2012**, *38*, 1186–1195. [\[CrossRef\]](#)
4. Zhao, D.; Wu, R.; Liu, X.; Zhao, Y. Apple positioning based on YOLO deep convolutional neural network for picking robot in complex background. *Trans. Chin. Soc. Agric. Eng.* **2019**, *35*, 164–173.
5. Kang, H.; Chen, C. Fast implementation of real-time fruit detection in apple orchards using deep learning. *Comput. Electron. Agric.* **2020**, *168*, 105108. [\[CrossRef\]](#)
6. Gené-Mola, J.; Vilaplana, V.; Rosell-Polo, J.R.; Morros, J.R.; Ruiz-Hidalgo, J.; Gregorio, E. Multi-modal deep learning for Fuji apple detection using RGBD cameras and their radiometric capabilities. *Comput. Electron. Agric.* **2019**, *162*, 689–698. [\[CrossRef\]](#)
7. Fu, L.; Majeed, Y.; Zhang, X.; Karkee, M.; Zhang, Q. Faster R-CNN-based apple detection in dense-foliage fruiting-wall trees using RGB and depth features for robotic harvesting. *Biosyst. Eng.* **2020**, *197*, 245–256. [\[CrossRef\]](#)
8. Gené-Mola, J.; Sanz-Cortielles, R.; Rosell-Polo, J.R.; Morros, J.R.; Ruiz-Hidalgo, J.; Vilaplana, V.; Gregorio, E. Fruit detection and 3D location using instance segmentation neural networks and structure-from-motion photogrammetry. *Comput. Electron. Agric.* **2020**, *169*, 105165. [\[CrossRef\]](#)
9. Quan, L.; Wu, B.; Mao, S.; Yang, C.; Li, H. An Instance Segmentation-Based Method to Obtain the Leaf Age and Plant Centre of Weeds in Complex Field Environments. *Sensors* **2021**, *21*, 3389. [\[CrossRef\]](#)
10. Liu, H.; Soto, R.A.R.; Xiao, F.; Lee, Y.J. YolactEdge: Real-time Instance Segmentation on the Edge. *arXiv* **2021**, arXiv:cs.CV/2012.12259.
11. Dandan, W.; Dongjian, H. Recognition of apple targets before fruits thinning by robot based on R-FCN deep convolution neural network. *Trans. Chin. Soc. Agric. Eng.* **2019**, *35*, 156–163.
12. Kang, H.; Chen, C. Fruit detection, segmentation and 3D visualisation of environments in apple orchards. *Comput. Electron. Agric.* **2020**, *171*, 105302. [\[CrossRef\]](#)
13. Zhang, J.; Karkee, M.; Zhang, Q.; Zhang, X.; Yaqoob, M.; Fu, L.; Wang, S. Multi-class object detection using faster R-CNN and estimation of shaking locations for automated shake-and-catch apple harvesting. *Comput. Electron. Agric.* **2020**, *173*, 105384. [\[CrossRef\]](#)
14. Yan, B.; Fan, P.; Lei, X.; Liu, Z.; Yang, F. A Real-Time Apple Targets Detection Method for Picking Robot Based on Improved YOLOv5. *Remote Sens.* **2021**, *13*, 1619. [\[CrossRef\]](#)
15. 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. [\[CrossRef\]](#)
16. Buemi, F.; Massa, M.; Sandini, G.; Costi, G. The agrobot project. *Adv. Space Res.* **1996**, *18*, 185–189. [\[CrossRef\]](#)
17. Kitamura, S.; Oka, K. Recognition and cutting system of sweet pepper for picking robot in greenhouse horticulture. In Proceedings of the IEEE International Conference Mechatronics and Automation, Niagara Falls, ON, Canada, 29 July–1 August 2005; Volume 4, pp. 1807–1812.
18. Xiang, R.; Jiang, H.; Ying, Y. Recognition of clustered tomatoes based on binocular stereo vision. *Comput. Electron. Agric.* **2014**, *106*, 75–90. [\[CrossRef\]](#)
19. Plebe, A.; Grasso, G. Localization of spherical fruits for robotic harvesting. *Mach. Vis. Appl.* **2001**, *13*, 70–79. [\[CrossRef\]](#)
20. Gongal, A.; Silwal, A.; Amatya, S.; Karkee, M.; Zhang, Q.; Lewis, K. Apple crop-load estimation with over-the-row machine vision system. *Comput. Electron. Agric.* **2016**, *120*, 26–35. [\[CrossRef\]](#)
21. Grosso, E.; Tistarelli, M. Active/dynamic stereo vision. *IEEE Trans. Pattern Anal. Mach. Intell.* **1995**, *17*, 868–879. [\[CrossRef\]](#)
22. Liu, Z.; Wu, J.; Fu, L.; Majeed, Y.; Feng, Y.; Li, R.; Cui, Y. Improved kiwifruit detection using pre-trained VGG16 with RGB and NIR information fusion. *IEEE Access* **2019**, *8*, 2327–2336. [\[CrossRef\]](#)
23. Tu, S.; Pang, J.; Liu, H.; Zhuang, N.; Chen, Y.; Zheng, C.; Wan, H.; Xue, Y. Passion fruit detection and counting based on multiple scale faster R-CNN using RGBD images. *Precis. Agric.* **2020**, *21*, 1072–1091. [\[CrossRef\]](#)
24. Zhang, Z.; Pothula, A.K.; Lu, R. A review of bin filling technologies for apple harvest and postharvest handling. *Appl. Eng. Agric.* **2018**, *34*, 687–703. [\[CrossRef\]](#)

25. Milella, A.; Marani, R.; Petitti, A.; Reina, G. In-field high throughput grapevine phenotyping with a consumer-grade depth camera. *Comput. Electron. Agric.* **2019**, *156*, 293–306. [\[CrossRef\]](#)
26. 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. [\[CrossRef\]](#)
27. Zhang, Y.; Tian, Y.; Zheng, C.; Zhao, D.; Gao, P.; Duan, K. Segmentation OF apple point clouds based on ROI in RGB images. *Inmateh Agric. Eng.* **2019**, *59*, 209–218. [\[CrossRef\]](#)
28. Lehnert, C.; Sa, I.; McCool, C.; Upcroft, B.; Perez, T. Sweet pepper pose detection and grasping for automated crop harvesting. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 2428–2434. [\[CrossRef\]](#)
29. Lehnert, C.; English, A.; Mccool, C.; Tow, A.W.; Perez, T. Autonomous Sweet Pepper Harvesting for Protected Cropping Systems. *IEEE Robot. Autom. Lett.* **2017**, *2*, 872–879. [\[CrossRef\]](#)
30. Yaguchi, H.; Nagahama, K.; Hasegawa, T.; Inaba, M. Development of an autonomous tomato harvesting robot with rotational plucking gripper. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea, 9–14 October 2016; pp. 652–657.
31. Lin, G.; Tang, Y.; Zou, X.; Xiong, J.; Li, J. Guava Detection and Pose Estimation Using a Low-Cost RGBD Sensor in the Field. *Sensors* **2019**, *19*, 428. [\[CrossRef\]](#)
32. Tao, Y.; Zhou, J. Automatic apple recognition based on the fusion of color and 3D feature for robotic fruit picking. *Comput. Electron. Agric.* **2017**, *142*, 388–396. [\[CrossRef\]](#)
33. Kang, H.; Zhou, H.; Chen, C. Visual Perception and Modeling for Autonomous Apple Harvesting. *IEEE Access* **2020**, *8*, 62151–62163. [\[CrossRef\]](#)
34. Häni, N.; Roy, P.; Isler, V. MinneApple: A benchmark dataset for apple detection and segmentation. *IEEE Robot. Autom. Lett.* **2020**, *5*, 852–858. [\[CrossRef\]](#)
35. Keskar, N.S.; Socher, R. Improving generalization performance by switching from adam to sgd. *arXiv* **2017**, arXiv:1712.07628.
36. Sahin, C.; Garcia-Hernando, G.; Sock, J.; Kim, T.K. A review on object pose recovery: From 3d bounding box detectors to full 6d pose estimators. *Image Vis. Comput.* **2020**, *96*, 103898. [\[CrossRef\]](#)
37. Magistri, F.; Chebrolu, N.; Behley, J.; Stachniss, C. Towards In-Field Phenotyping Exploiting Differentiable Rendering with Self-Consistency Loss. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 13960–13966. [\[CrossRef\]](#)
38. Bellocchio, E.; Costante, G.; Cascianelli, S.; Fravolini, M.L.; Valigi, P. Combining domain adaptation and spatial consistency for unseen fruits counting: A quasi-unsupervised approach. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1079–1086. [\[CrossRef\]](#)
39. Ge, Y.; Xiong, Y.; From, P.J. Symmetry-based 3D shape completion for fruit localisation for harvesting robots. *Biosyst. Eng.* **2020**, *197*, 188–202. [\[CrossRef\]](#)