International Conference of Agricultural Engineering
AgEng 2014 Zurich 6–10 July
Ref: C0229

Apple detection algorithm for robotic harvesting using a RGB-D camera

Tien Thanh Nguyen, Koenraad Vandevoorde, Erdal Kayacan,

Josse De Baerdemaeker, Wouter Saeys

MeBioS (Mechatronics, Biostatistics and Sensors) Division,

Biosystems Department, KU Leuven, Belgium

Tel: +32 163 21444; email: ***tienthanh.nguyen@biw.kuleuven.be***

Abstract

A fruit-by-fruit harvesting robot has not only to locate and characterize the fruits on a tree, but also to detect the obstacles for a collision free operation. Although many researchers have investigated this topic, recognizing and locating fruits on a tree is still a key challenge in developing such a robotic harvesting system due to the occlusion of the target fruits (by foliage, branches or other fruits) and the non-uniform and unstructured environment. Therefore, this study aimed at the development and validation of a system for the localization of apples and obstacles in the orchard based-on the color and depth images acquired with an RGB-D camera (Red, Green, Blue plus Depth). As an RGB-D camera provides both visual and 3D shape information, it is very suitable for 3D perception, which is particularly relevant for robots operating in unstructured environments. Using both color and depth information, an algorithm to detect apples on tree was developed. To separate the pixels belonging to the red or bicolored apples from the foliate background, the redness of each pixel in the color images was measured. Because of its simplicity and efficiency, the Euclidean clustering algorithm has been applied to the depth point cloud to extract the cluster for each apple. Finally, the RANdom Sample Consensus (RANSAC) algorithm has been used to fit each detected cluster to a pre-defined apple model. By using an iterative estimation method to the partially visible target object data (maximum 50% of the target object is observable from a single view-point), RANSAC provides the location of the center, the orientation and diameter of each detected apple. The accuracy of the elaborated algorithm was then tested on RGB-D images acquired in the apple orchard for which ground truth information had also been recorded. With 100% detection rate for the fully visible apples, 85% detection rate for the partially occluded apples and less than 1 cm localization error in a short run time (less than 1 second for the detection of 20 apples), the developed algorithm was found to be suitable for implementation in a robotic harvesting system. Moreover, thanks to its generic approach, this algorithm can also be applied for detecting pre-defined obstacles e.g. trunks, branches, wires. To minimize the memory consumption and calculation time for collision checking, the obstacles are represented in simple geometries such as spheres and cylinders, and stored in a single 3D map with respect to the global coordinates of the robot. The methodology presented in this paper has been tested on an apple picking robot and it has been observed that the developed algorithm can meet the requirements for the robotic harvesting system. Furthermore, the developed algorithm could also be used for yield estimation and orchard monitoring in a management decision support system.

Keywords: harvesting, robot, detection, apple

1 Introduction

Harvesting of seasonal fruits like apples requires intensives labor in a short period, which is typically done by seasonal workers. This labor cost comprises a significant part of the entire production cost. To reduce the labor demand, several mechanized solutions for fruit harvesting such as rotating beater bars or trunk shakers have been elaborated for mass harvesting methods. However, as most of these mechanized solutions lead to impact damage which makes the harvested products not suitable for the delicate fruits like table apples, bruising in the fruit must be avoided. Therefore mass harvesting methods are not applicable, and the selective harvesting method (fruit by fruit) in which fruits are picked softly by a manipulator is preferred. However, one of the bottle-necks in developing the selective harvesting machine system is the detection of the fruits on the trees. It has to deal with complex, unstructured and uncontrolled environment. Many studies were reported in fruit detection on the trees; most of them were using computer vision which can be classified into three groups according to the hardware:

- Spectral-based sensors such as color, thermal, multi-spectral and hyper-spectral cameras
- Range-based sensors such as stereo cameras, ToF (Time of Flight) cameras, laser scanner
- Combination of these 2 sensing devices using sensor fusion

The detection techniques using spectral-based sensors rely on the differences in the reflectance spectra between fruits and the background, while the detection techniques using range-based sensors depend on the specificity of the fruit shape.

In the category of the spectral-based sensors RGB cameras are by far the most often used. Tabb et al. (2006) segmented apple fruit using RGB camera with light cover by modelling the background's color properties. Ji et al. (2012) proposed an algorithm to recognize the apple fruit in real-time based on SVM (Support Vector Machine) with color and shape features from images acquired with an RGB camera in an uncontrolled environment without any light cover and artificial light source Apart from the RGB cameras, also multi- and hyperspectral cameras have been proposed. For example, Bac et al. (2013) used a multispectral camera to detect the stem and fruit in sweet peppers, while Gómez-Sanchis et al. (2008) used a hyper-spectral system to detect mandarin fruits and Stajinko et al. (2004) used a thermal camera for apple monitoring. While these spectral-based cameras can provide good contract for fruit detection in 2D, they are not sufficient to detect the 3D pose of the fruit required for robotic harvesting.

Range-based devices such as stereo vision cameras, ultrasonic sensors, laser scanners and Time of Flight cameras measure the distance from the sensor to the observed objects. Time-of-flight cameras and laser scanner provide accurate range information in real-time and are consistent with varying lighting condition, are used more widely in agricultural machines nowadays. As fruits have a specific 3D shape they can be recognized in the distance images. Tanigaki et al. (2008) and Van Henten et al. (2003) used stereo vision and artificial light sources to respectively locate cherry and cucumber fruits in the canopy. However, range-based devices often have difficulties with the similarity of shapes between fruits and other objects, and the occlusion where fruits are partially covered by other fruits and leaves.

Another option to obtain distance information is by combining multiple 2D images acquired while moving towards the target. Baeten et al. (2008) and De-An et al. (2011) used an RGB camera mounted on the end-effector of a harvesting robot to calculate the location of the fruits while moving the camera. Instead of choosing between the discrimination power of spectral-based sensors and the distance information from range-based sensors, both sensor types can be combined. Adhikari & Karee (2011) developed a 3D machine vision system consisting of a ToF camera and a color vision camera to reconstruct apple trees for automatic pruning. Therefore, the main objective of this study was to develop a multiphase algorithm to detect and localize apple fruits by combining an RGB-D camera and point cloud processing techniques. The outline of this paper is as follows. In section 2 the working environment, the sensor system and the detection algorithm are described. The results are presented and discussed in Section 3. Finally, in section 4, conclusions are drawn from this study and suggestions for the future work are proposed.

2 Materials and methods

2.1 Working environment

The RGB-D images were acquired in high density, experimental apple orchards with espalier trellis system which had been pruned into a planar canopy to simplify manual or mechanized harvesting. Pruning was performed as such that the fruit trees formed a ‘fruit wall’ where the fruit are attached closely to the trunk or the main branches, as shown in Figure 1(a). The advantages of using this planting system are a reduction of the occlusion of the fruits by branches and leaves and the better-structured environment. In Belgium, a well maintained orchard of three year old ‘Jonared’ was used for testing in Sint-Truiden. The moving platform can run over the tree row with a plastic shield which covers the working space and an artificial illumination mounted inside, as shown in Figure 1(b).

a) b) c)

Figure 1: Espalier apple tree, measurement platform and sensor system

2.2 Sensor system

The sensor system for fruit detection in the orchard is a RGB-D sensor (Xtion PRO Live, ASUS, Taipei, Taiwan) consisting of a color image sensor, an IR image sensor and an IR light source for depth measurement. The basic principle of the depth measurement of the RGB-D sensor is the emission of an IR pattern and the simultaneous image capture of the IR image with a CMOS camera and IR-pass filter (Andersen et al, 2012). The image processor of the RGB-D sensor uses the relative positions of the dots in the pattern to calculate the depth displacement at each pixel position in the image. For the RGB-D sensor, the actual depth value of a point is the distance from the sensor-plane to this point rather than the distance from the sensor itself. Moreover, each pixel in the image acquired by the color camera has been registered with each pixel of the IR camera of the RGB-D sensor. Therefore, the output acquired by the RGB-D camera is a colored point cloud which is a set of data points in (x, y, z) coordinates with color information (R, G, B).

Table 1: Specifications of Asus Xtion PRO Live sensors

Distance of Use	Between 0.8 m and 3.5 m
Field of View	$58^o H, 45^o V, 70^o D$ (Horizontal, Vertical, Diagonal)
Sensor	RGB & Depth & Microphone x 2
Depth Image Size/Frame rate	VGA (640x480) : 30 fps & QVGA (320x240): 60 fps
RGB Image Size	SXGA (1280x1024)
Nominal spatial resolution	3 mm (at 2 m distance)
Nominal depth resolution	1 cm (at 2 m distance)

In our sensor system, two ASUS Xtion PRO Live sensors have been mounted on both sides of the first joint of the robotic manipulator illustrated in Figure 1(c). In this way, both sensors can be moved vertically to acquire images from the whole tree, from two different viewpoints. The main specifications of the ASUS Xtion PRO Live sensor are summarized in Table 1.

2.3 Apple fruit detection algorithm

In this section, the fruit detection algorithm to locate apples in 3D (x, y, z) coordinates is described. The inputs are colored point clouds acquired with the ASUS Xtion Live sensors which are placed 1m away from the tree row and pointed straight to the tree row. The detection algorithm exploits both the color and shape properties to detect and localize the apple fruits on the trees. The implementation of the detection algorithm is based on the Point Cloud Library (PCL) (Rusu et al., 2011) and completely written in C++.

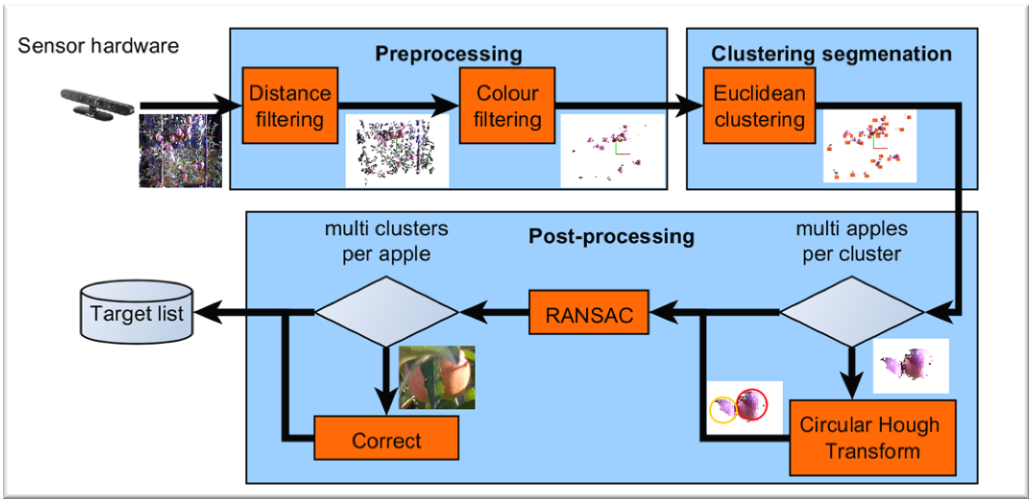


Figure 2: Pipeline of detection algorithm

The detection pipeline which represents the detailed procedure of the algorithm is schematically illustrated in Figure 2. As can be seen in the figure, our detection algorithm consists of three phases. The first phase – preprocessing: pruning of the point cloud – aims at selecting the data points inside the region of interest which are suspected to belong to apples. The second phase - clustering segmentation – segments the ‘red’ point cloud resulting from the first phase into multiple clusters where each cluster is supposed to belong to one single apple. The last phase - post-processing - is for estimating the location and diameter of each fruit, and for solving the problem of single clusters including multiple apples..

2.3.1 Preprocessing: Pruning of the point cloud

The preprocessing phase aims to select only the interesting data points, by removing the data points corresponding to data points from leaves, trunks and branches. In this way, the overall processing time of the detection procedure can be significantly reduced. To reach this goal two filters have been implemented: a distance filter and a color filter. The distance filter is applied to the depth data of the point cloud, and limits the further processing to data points within predefined distance boundaries. These boundaries are defined as such that only fruits from the targeted tree row hanging at the same side of the tree row as where the camera is positioned are detected.

The color filter method proposed by (Bulanon et al., 2002) for segmenting apple fruit from the background has also been used here. In this approach, two coefficients r and g represent the normalized red and green features, respectively:

$$r = \frac{R}{R + G + B} \quad and \quad g = \frac{G}{R + G + B}$$

By combining these features two decision functions, d_1 and d_2 can be calculated to separate the fruits from others classes in the feature space :

$$\begin{bmatrix} d_1 \\ d_2 \end{bmatrix} = \begin{bmatrix} 0.09 & -0.11 \\ 0.12 & -0.06 \end{bmatrix} \begin{bmatrix} r \\ g \end{bmatrix}$$

The decision functions are applied to all point cloud data P_s, to determine if the point $p_i \in \boldsymbol{P}_s$ is high enough in red and low enough in green color to belong to the apple point cloud $\boldsymbol{P}_r$.

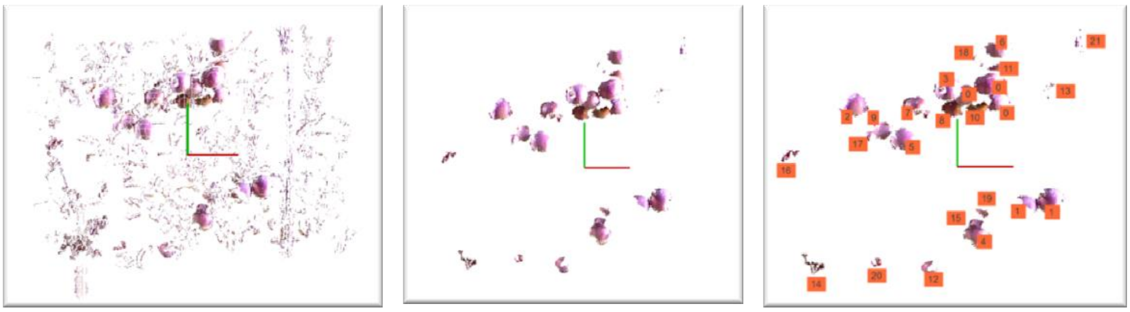
2.3.2 Point cloud clustering segmentation

Point cloud clustering segmentation is the main procedure of our fruit detection algorithm. By using the Euclidean clustering technique to the unorganized point cloud model $\boldsymbol{P}_r$ is divided into multiple clusters $\boldsymbol{P}^i_{apple}$ which each are supposed to belong to an apple number i. A point cloud cluster is defined as follows. Let $\boldsymbol{O}_i = \{p_i \in \boldsymbol{P}_i\}$ be a distinct point cluster $\boldsymbol{O}_j = \{p_i \in \boldsymbol{P}_j\}$ if:

$$min\|p_i - p_j\| \geq d_{th}$$

where d_{th} is a maximum imposed distance threshold. The above equation states that if the minimum distance between a set of points $p_i \in \boldsymbol{P}_i$ and another set $p_j \in \boldsymbol{P}_j$ is larger than a predefined distance value, then the points in p_i are set to belong to a point cluster $\boldsymbol{O}_i$ and the ones in p_j to another distinct point cluster $\boldsymbol{O}_j$.

Although this Euclidean clustering algorithm is well able to detect the apples, it cannot deal with the situation of multiple apples hanging close together in a bushel, as shown in Figure 4(a). In this situation, two or more apples are really close or touching each other. This makes that the minimum distance between the data points belonging to two different apples is smaller than the distance threshold d_{th}. So, in the point cloud, the apple bushel will be recognized as one single cluster. This will lead to an underestimation of the number of apples on the tree and an error in the positioning of the individual apples. To overcome this limitation, a post-processing step has been included in the algorithm.



a) b) c)

Figure 3: Illustration of different steps of Euclidean clustering segmentation of the distance and color filtered point cloud: a) original point cloud, b) detected clusters, c) labeled clusters

2.3.3 Post-processing

After the clustering, the pruned point cloud has been segmented in multiple clusters which each are supposed to belong to one apple fruit. The post-processing phase of the detection algorithm aims for two purposes:

- for solving the problem of single clusters including multiple apples
- for estimating the location and diameter of each fruit

First, the point cloud of the suspected cluster is converted into a binary image based on the 2D location of each point in the point cloud. By applying the *HoughCircles* function available in the OpenCV library to the binary image regions which represent apples are detected. The effect of this algorithm is illustrated in Figure 4(b) where the bushel cluster has been correctly separated in two separate apple clusters.

a) single bushel cluster b) separate apple clusters

Figure 4: Illustration of the post-processing algorithm for separating a single bushel cluster identified by the Euclidean clustering algorithm into separate apple clusters (circles with different color)

In a final step, the RANSAC algorithm is applied to the detected apple clusters to estimate the location of the center of gravity and the diameter of each fruit. RANSAC (Fischler & Bolles, 1981) is an iterative approach for model recognition that is used to estimate parameters of a mathematical model from a data set containing outliers. In our apple detection algorithm, a simple sphere was selected as the model to be fitted to the apple clusters such that the parameters to be estimated are the center location in 3D and the radius of the sphere.

3 Results and Discussion

The detection algorithm was tested in the apple orchard in Sint-Truiden, Belgium. The testing cultivar is Jonared which is a red apple cultivar. From the 100 apples present in the acquired point clouds 90% was correctly detected. For the completely visible fruits, the recognition percentage was 100%, while the recognition percentage for the occluded fruits was only 85%. An analysis of the un-recognized fruits revealed that these were the highly occluded fruits and the fruits at the edges of the image which had a very small visible part.

The error of the diameter estimation has been determined by measuring the diameter of the detected apples. The location ground truth for the detected apples was measured by using the robotic manipulator. The manipulator will move to the point which is in the plane of global coordination of the manipulator and the detected center of the fruit, and has the distance to the center of the fruit equal to the estimated diameter plus added value for measurement. The goal point and the center of the fruit make a line that is perpendicular with the sensor plane. The error of the location will be calculated as the error distance from the tool center point of the manipulator minus the error of the diameter estimation and the added value for measurement. The diameter estimation was quite large with an approximated error of 1.7 cm bigger than the measured diameter. The reason is the RANSAC algorithm still have problem with partial data and nonuniform model. For the 10 apples for which the location ground truth has been measured the root-mean-square location deviation was 0.9 cm. This location error below 1cm is considered acceptable for robotic harvesting as the grasping margin of the gripper is larger than 1cm.

Although the fruit detection does not have to be in real-time to allow robotic harvesting, it should not be too slow as this would increase the cycle time. In this study, the time needed to detect 20 apples in one scene was approximately 1s. As the aim is to pick one fruit per 7s, this detection time of 50ms per fruit is considered acceptable and the fruit detection algorithm is efficient enough for the realistic application.

4 Conclusions

In this study, a system for detection and localization of red and bicolored apples on the trees based on color and range data captured by RGB-D sensors has been proposed and tested.

The algorithm achieved 100% correct detection for fully visible apples and 85% for occluded apples. The maximum errors for the location estimation were less than 1 cm which is manageable for the robot system. With an average calculation time of 50 ms per apple, the algorithm is also efficient enough for the autonomous harvesting application. Moreover, the detection algorithm is straightforward and simple and could also be used to detect other tree fruits such as peaches or plums. Apart from the robotic harvesting application, this detection algorithm could also be used for orchard monitoring and yield prediction.

However, it should be noted that the algorithm is limited to the detection of red and bi-colored apples. To be able to also detect green apple, the color filtering should be improved or replaced by an alternative filtering strategy. This might be achieved by using machine learning methods which can classify color regions in the point cloud rather than individual pixels and/or by using other features like object texture.

5 Acknowledgements

The authors gratefully acknowledge the support of the colleagues of INIA (Chile): Dr. Stanley Best, Lorenzo Leon, Fabiola Flores. The authors wish to thank Ir. Tom Deckers and his colleagues of PCFruit in Sint-Truiden (Belgium) for their valuable advices and preparation of the testing orchards.

This research was funded by the European Commission in the 7th Framework Programme (CROPS GA no.246252). The opinions expressed in this document do by no means reflect the official opinion of the European Union or its representatives.

6 References

Adhikari, B., & Karee, M. (2011). 3d reconstruction of apple trees for mechanical pruning. *Proceedings of the American Society of Agricultural and Biological Engineers (ASABE) Annual International Meeting.* Kentucky.

Andersen et al. (2012). *Kinect Depth Sensor Evaluation for Computer Vision Applications.* Aarhus University.

Bac, C.W. et al. (2013). Robust pixel-based classification of obstacles for robotic harvesting of sweet-pepper. *Computers and Electronics in Agriculture, 96*, 148-162.

Baeten et al. (2008). Autonomous fruit picking machine: A robotic apple harvester. *Field and Service Robotics - Springer Tracts in Advanced Robotics, 42*, 531-539.

Bulanon et al. (2002). A color model for recognition of apples by a robotic harvesting system. *Journal of the Japanese Society of Agricultural Machinery, 64*(5), 123-133.

Fischler, M. A., & Bolles, R. C. (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *SRI International*, 381-395.

Gómez-Sanchis et al. (2008). Hyperspectral system for early detection of rottenness caused by penicillium digitatum in mandarins. *Journal of Food Engineering, 89*(1), 80-86.

Ji et al. (2012). Automatic recognition vision system guided for apple harvesting robot. *Computers and Electrical Engineering, 38*(5), 1186-1195.

Rajendra et al. (2009). Machine vision algorithm for robots to harvest strawberries in tabletop culture greenhouses. *Engineering in Agriculture, Environment and Food, 2*(1), 24-30.

Rusu et al. (2011). 3D is here: Point Cloud Library (PCL). *IEEE International Conference on Robotics and Automation.* Shanghai, China.

Stajinko et al. (2004). Estimation of number and diameter of apple fruits in an orchard during the growing season by thermal imaging. *Computers and Electronics in Agriculture, 42*(1), 31-42.

Tabb et al. (2006). Segmentation of apple fruit from video via background modeling. *Proceedings of the American Society of Agricultural and Biological Engineers (ASABE) Annual International Meeting.* Portland.

Tanigaki et al. (2008). Cherry-harvesting robot. *Computers and Electronics in Agriculture, 63*(1), 65-72.

Van Henten et al. (2003). Field test of an autonomous cucumber picking robot. *Biosystems Engineering, 86*(3), 305-313.