



**Queensland University of Technology**  
Brisbane Australia

This is the author's version of a work that was submitted/accepted for publication in the following source:

Sa, Inkyu, Lehnert, Christopher, English, Andrew, McCool, Christopher, Dayoub, Feras, Upcroft, Ben, & Perez, Tristan  
(2017)

Peduncle detection of sweet pepper for autonomous crop harvesting:  
Combined colour and 3D information.

*IEEE Robotics and Automation Letters*. (In Press)

This file was downloaded from: <https://eprints.qut.edu.au/102962/>

© Copyright 2016 IEEE

**Notice:** *Changes introduced as a result of publishing processes such as copy-editing and formatting may not be reflected in this document. For a definitive version of this work, please refer to the published source:*

<https://doi.org/10.1109/LRA.2017.2651952>

# Peduncle Detection of Sweet Pepper for Autonomous Crop Harvesting - Combined Colour and 3D Information

Inkyu Sa<sup>1\*</sup>, Chris Lehnert<sup>2</sup>, Andrew English<sup>2</sup>, Chris McCool<sup>2</sup>, Feras Dayoub<sup>2</sup>, Ben Upcroft<sup>2</sup>, and Tristan Perez<sup>2</sup>

**Abstract**—This paper presents a 3D visual detection method for the challenging task of detecting peduncles of sweet peppers (*Capsicum annuum*) in the field. Cutting the peduncle cleanly is one of the most difficult stages of the harvesting process, where the peduncle is the part of the crop that attaches it to the main stem of the plant. Accurate peduncle detection in 3D space is therefore a vital step in reliable autonomous harvesting of sweet peppers, as this can lead to precise cutting while avoiding damage to the surrounding plant. This paper makes use of both colour and geometry information acquired from an RGB-D sensor and utilises a supervised-learning approach for the peduncle detection task. The performance of the proposed method is demonstrated and evaluated using qualitative and quantitative results (the Area-Under-the-Curve (AUC) of the detection precision-recall curve). We are able to achieve an AUC of 0.71 for peduncle detection on field-grown sweet peppers. We release a set of manually annotated 3D sweet pepper and peduncle images to assist the research community in performing further research on this topic.

## I. INTRODUCTION

Automating agricultural processes such as planting, harvesting, weeding and inspection will play a key role in helping to improve farm productivity, increase crop quality and reduce input costs [1]. In particular, the harvesting of high value crops within horticulture still demands a large amount of hand labour costs due to the dexterity required to achieve the task. For these reasons, automating the task of harvesting is of great interest to the horticultural and robotics industries. Recent research in robotics has made impressive progress towards the goal of developing viable broad-acre [2], [3] and horticultural robots [4]–[8].

This paper focuses on peduncle detection in order to improve autonomous harvesting. It is highly desirable to be able to identify the peduncle prior to performing crop removal (e.g., cutting the peduncle or pulling of the crop using a custom end-effector) as this is one of the most challenging steps of the harvesting process. Accurate estimation of the peduncle reduces the potential of damaging the crop and other branches of the plant. Additionally, retaining the proper peduncle maximises the storage life and market value of each pepper. More importantly, accurate peduncle detection can lead to higher success rates for crop detachment, which

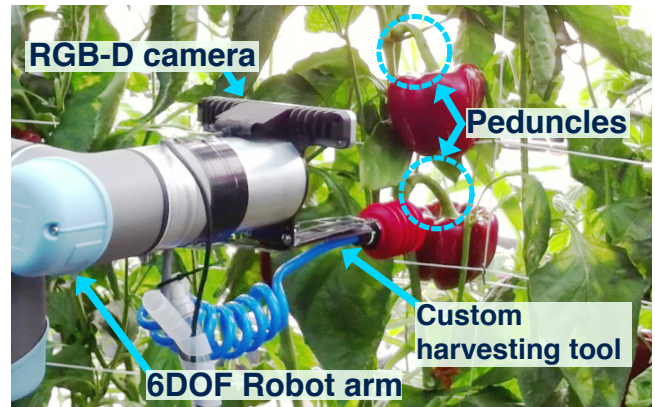


Fig. 1. Sweet pepper picking in operation showing a robotics arm equipped with an end-effector tool to harvest the pepper by cutting its peduncle. The photo highlights the presence of occlusions and varying lighting conditions of peduncles and sweet peppers grown in a field environment.

in turn yields more harvested crops. Peduncle detection is challenging because of varying lighting conditions and the presence of occlusions by leaves or other crops as shown in Fig. 1. Additionally, while peduncles are usually green, crops such as sweet peppers vary in colour from green through to red (with other variations possible), making it difficult to detect peduncles based on colour information alone.

Prior work in peduncle detection has either identified the peduncle post-harvest or yielded accuracies that are too low to be of practical use. In [9], [10] the detection of the peduncle was performed using a Bayesian discriminant model of RGB colour information. However, such an approach is not practical in-situ for crops such as sweet pepper where the colour alone does not discriminate between peduncles, leaves and crops (see Fig. 1). In-situ detection of peduncle (multi-class detection) using multi-spectral imagery was performed for sweet peppers by Bac *et al.* [11]. Unfortunately their accuracy was too low to be of practical use. In order to address previous shortcomings, we use both colour and geometry shape features captured from multiple views by an RGB-D camera.

The aim and scope of this paper is peduncle detection from a 3D reconstructed model of a detected sweet pepper. In our previous work [12], we described a detection system based on deep neural networks which is capable of performing the detection step and in [6], we introduced a sweet pepper pose estimation method that combines point clouds from multiple viewpoints into a coherent point cloud using Kinect Fusion [13]. In this work, we thus assume that pre-registered

\* This work was done when Inkyu Sa was in QUT.

<sup>1</sup> Autonomous Systems Lab., Department of Mechanical and Process Engineering, ETH Zurich, Zurich, Switzerland. [inkyu.sa@mavt.ethz.ch](mailto:inkyu.sa@mavt.ethz.ch) <sup>2</sup> Science and Engineering Faculty, Queensland University of Technology, Brisbane, Australia. {[c.lehnert](mailto:c.lehnert), [a.english](mailto:a.english), [c.mccool](mailto:c.mccool), [feras.dayoub](mailto:feras.dayoub), [ben.upcroft](mailto:ben.upcroft), [tristan.perez](mailto:tristan.perez)}@qut.edu.au

3D models of the scene containing the peduncle and sweet pepper are already obtained and the presented method is deployed afterwards.

The main contribution of this paper is the use of 3D geometric features, namely Point Feature Histograms (PFH) [14] in addition to colour information for robust and accurate peduncle detection. The proposed detection system operates with any sweet pepper type (red, green, and mixed), and achieves high accuracy. To encourage reproducibility, we share the manually annotated set of 3D capsicum and peduncle models along with their annotated ground truth used to generate the results in this paper<sup>1</sup>.

The remainder of this paper is structured as follows. Section II introduces relevant related works. Section III describes sweet pepper peduncle detection and Section IV elaborates data collection environments and procedures. We present our experimental results in Section V. Conclusions are drawn in Section VII.

## II. RELATED WORK/BACKGROUND

This section reviews existing methods for detecting peduncles and crops using 3D geometry and visual features. Such techniques are widely used for autonomous crop inspection and detection tasks. Cubero *et al.* demonstrated the detection of various fruit peduncles using radius and curvature signatures [15]. The Euclidean distance and the angle rate change between each of the points on the contour and the fruit centroid are calculated. The presence of peduncles yields rapid changes in these metrics and can be detected using a specified threshold. Blasco *et al.* [9] and Ruiz *et al.* [10] presented peduncle detection of oranges, peaches, and apples using a Bayesian discriminant model of RGB colour information. The size of a colour segmented area was then calculated and assigned to pre-defined classes. These methods are more likely suitable for the quality control and inspection of crop peduncles after the crop have been harvested rather than for harvesting automation, as they require an inspection chamber that provides ideal lighting conditions with a clean background, no occlusions, good viewpoints, and high-quality static imagery.

There has also been significant progress in crop detection and plant classification for in-field crop harvesting [16]. Recently, a similar work for autonomous sweet pepper harvesting in greenhouses was presented by [17]. The harvesting method within this project used only the centroid of each sweet pepper, with a harvesting tool that encloses the fruit and cuts the peduncle without its detection. Within this project, multi-spectral images (robust under varying lighting conditions) were tested to classify different parts of sweet pepper plants [11]. This work achieved 40% multi-classification for parts of the plant. Although these results were deemed insufficient for use on a real robot, they demonstrated significant progress towards a plant classification system based on hyper-spectral information.

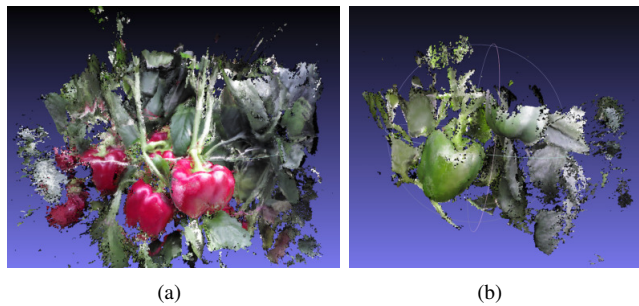


Fig. 2. Reconstructed red (a) and green (b) sweet peppers using Kinect Fusion. The texture and colour information is projected onto the 3D model as an average weighting from multiple viewpoints of an RGB-D camera.

Using an accurate laser scanner would be beneficial for plant classification. Paulus *et al.* [18] showed that the point feature histograms obtained from different plant structures, such as leaves and stems, are highly distinguishable. They, in turn, could easily classify plant organs with high accuracy using a supervised classification algorithm. Wahabzada *et al.* [19] demonstrated grapevine, wheat, and barley organ classification using data acquired from a 2D laser scanner mounted on a robotic arm. A 3D point cloud with an accuracy of 45  $\mu\text{m}$  was produced from the scan using a  $k$ -means clustering algorithm. Our work differs from this approach in that we use a consumer-level affordable RGB-D camera with significant measurement noise and apply it in a field site.

Strawberry peduncle detection was reported by [20]. The region of interest (ROI) was pre-defined using prior knowledge and the boundary point between a fruit and peduncle was detected using colour information. The inclination of the peduncle - the angle between the vertical axis and the boundary point - was computed. It is possible to easily distinguish the boundary point for red strawberry but is a challenge to exploit this approach for green sweet pepper.

## III. METHODOLOGY

Detecting the peduncles of sweet peppers is a challenging task. Unlike other crops, such as apples and mangoes, which have straight and vertical peduncles, the peduncles of sweet peppers are highly curved and sometimes even flattened against the top of the fruit as shown in Fig. 1 and Fig. 2.

In our work, we combine colour information that can distinguish red sweet peppers from their green peduncles with geometry features [21] that discriminate between peppers and peduncles that are both green. In this Section, we present a feature vector representation and the use of a supervised classifier for peduncle detection.

### A. Feature representation

It is important to determine the discriminative features of an object we wish to detect. To achieve this, we utilise two features: colour and geometry shape.

<sup>1</sup> Available at: <http://goo.gl/8BtcQX>

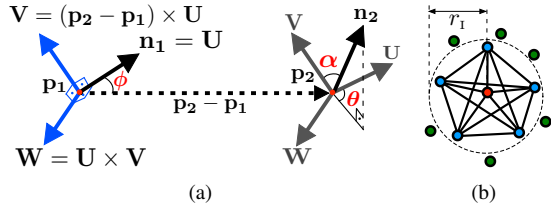


Fig. 3. (a) Illustration of three estimated angular features for a pair of points  $\mathbf{p}_1$  and  $\mathbf{p}_2$  and their associated normals  $\mathbf{n}_1$  and  $\mathbf{n}_2$  [14]. (b) The influence region  $r_1$  of a Point Feature Histogram. The red and green indicates the query point and its  $k$ -neighbours respectively.

1) *Colour feature (HSV)*: As shown in Fig. 2 (a), a highly distinguishable feature between a red sweet pepper and its peduncle is colour. Selecting a colour space for object detection is non-trivial [22] and we use the Hue, Saturation, and Value (HSV) colour space that expresses visual information in the dimensions of colour (H), lightness (S) and intensity (V). Since the HSV space has a component accounting for pixel brightness, its colour components are not significantly affected by varying shades of the same colour. This is useful for visual crop detection, given that 3D objects reflect light differently depending on the angle of incidence onto a surface as shown in Fig. 2 (a). Furthermore, using the RGB colour space may not be appropriate for peduncle detection due to the high correlation between its R, G, and B dimensions. This can result in problems under certain conditions where light reflects off differently from solid-colour objects and shadows obscured by the plants.

2) *Geometry feature (Point feature histograms [14])*: Although colour cues provide good distinguishable features for detecting red sweet pepper peduncles, they are only of limited use for green peppers as shown in Fig. 2 (b) since the fruit and its peduncle have similar colour responses. Furthermore, using only colour, it is difficult to distinguish between green peduncles and other plant parts, such as leaves or stems, which are also green.

Using geometry features, such as surface normal and curvature estimates, can be useful for identifying different classes with distinguishable 3D geometries. Fig 4 shows surface normals of red (left) and green (right) sweet peppers. Here, it can be qualitatively observed that the peduncle and the body have different curvature and surface normal directions. In other words, the curvature radius for a peduncle can be generally easily distinguished from that of a sweet pepper body.

Point Feature Histograms (PFHs) [14] are an example of a descriptor that can capture geometry characteristics from a 3D point cloud. They represent primary surface properties given a sample point  $\mathbf{p}$  by combining geometrical information such as Euclidean distance and normals between the query point  $\mathbf{p}$  and its  $k$  neighbours as shown in Fig. 3(b). Given two points  $\mathbf{p}_1$  and  $\mathbf{p}_2$  from Fig. 3(a) and their normals  $\mathbf{n}_1$  and  $\mathbf{n}_2$ , the quadruplet consisting of  $\langle \alpha, \phi, \theta, d \rangle$  can

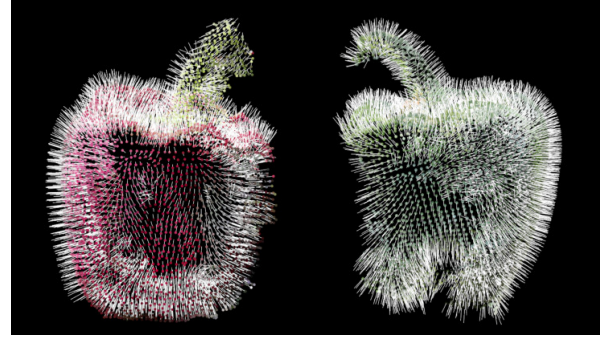


Fig. 4. Surface normal visualisation for a red (left) and a green (right) sweet pepper. This step is important to inspect and select the right search radius  $r_N$  for calculating point normals. This  $r_N$  should be properly chosen based on the level of detail required by the application. In this paper, we choose  $r_N$  as 0.01 m in order to capture the curvature of a sweet pepper body and a peduncle.

be formulated as:

$$\begin{aligned} \alpha &= \mathbf{V} \cdot \mathbf{n}_2 \\ \phi &= \mathbf{U} \cdot \frac{(\mathbf{p}_2 - \mathbf{p}_1)}{d} \\ \theta &= \arctan(\mathbf{W} \cdot \mathbf{n}_2, \mathbf{U} \cdot \mathbf{n}_2) \\ d &= \|\mathbf{p}_2 - \mathbf{p}_1\|_2 \end{aligned} \quad (1)$$

where  $\mathbf{U}$ ,  $\mathbf{V}$ , and  $\mathbf{W}$  determine an oriented orthonormal basis of the curve (Darboux frame) shown in Fig. 3(a). The quadruplets are calculated for each pair of points within the influence region  $r_1$  (the dotted line from Fig. 3(b)) and then binned into a histogram of size 33 [23] that divides the feature space. It is important to mention that the performance of feature extraction is affected by parameters such as the normal search radius and the influence region radius as illustrated in Fig. 3(b). These parameters should be appropriately selected depending on the application. If these values are set too large, then the surface normals of a peduncle and body will be identical since the set of points within the radius cover points from neighbouring surfaces. However, if these values are too small, then the normal will be dominated by random noise and will not properly represent surface curvature and geometry shape. In this paper, we set  $r_1$ , and  $r_N = 0.01$  based on empirical evaluations such as those shown in Fig. 4.

### B. Support Vector Machines (SVMs) classifier tuning

At this stage, we can build a feature vector for a single 3D point with 36 float elements extracted from HSV (3) and PFH (33). A concatenated feature vector ( $n \times 36$ ) where  $n$  is the number of points, serves as an input for a support vector machine (SVM) classifier. SVMs are one of the most successful and popular binary supervised learning algorithms (i.e. two-class classification) due to their simplicity, flexibility, effectiveness in dealing with high-dimensional spaces and ease of training. We provide insight into our experimental datasets in the following section, as well as choosing a kernel function and its parameters in section V-B. A more in-depth explanation of SVMs can be found in [24] chapter 4.



Fig. 5. QUT’s Harvey, a sweet pepper harvester prototype. In this paper, we only use the robotic arm and RGB-D sensor for peduncle detection. System integration with the proposed peduncle detection and the custom end-effector to perform autonomous sweet pepper harvesting will be tackled in future work.

#### IV. SYSTEM SETUP AND DATA COLLECTION

Image data training and system testing was performed using a prototype robotic sweet pepper harvester, “Harvey”, we developed at QUT. As shown in Fig. 5, this device possesses a 6 DoF robotic arm from Universal Robots (UR5) mounted on a scissor lift platform. The robot arm has a customised harvesting tool mounted on its end effector that both grips each pepper with a suction cup and cuts the peduncle with an oscillating cutting blade. An Intel i7 PC records incoming RGB-D data ( $640 \times 480$ ) from a eye-in-hand depth camera<sup>2</sup>.

Fig. 6 shows the internal peduncle detection pipeline. The reconstruction of a dense sweet pepper point cloud from multiple views using Kinect fusion is implemented based on our previous work [6] (green). The point clouds are extremely large and noisy, consequently we use a statistical outlier remover and voxel grid down sampler (blue) supported from Point cloud library (PCL) [25]. Surface normals are calculated and fed into the PFH feature extractor. The PFH features are then concatenated with HSV colour information to create the 36 dimensional feature vector (red) used for classification. Finally, peduncle detection is performed using a SVM with the RBF kernel and the trained model from Section V-B.1 (black).

The Robot Operating System (ROS, Indigo) [26] is used for system integration on an Ubuntu 14.04 Linux system. The methods in the green, blue, and red boxes shown in Fig. 6 are implemented in C++ using PCL, while peduncle detection (black) is written in Python 2.7 [27].

##### A. Training/testing data collection

Field trials for data collection were conducted on a farm in North Queensland (Australia) over a 10-day period within a protected cropping system. Such a system involves growing plants under a structure (e.g., greenhouse) that aims to create

<sup>2</sup>Intel Real Sense F200, shorter range (0.2 m-1.2 m)

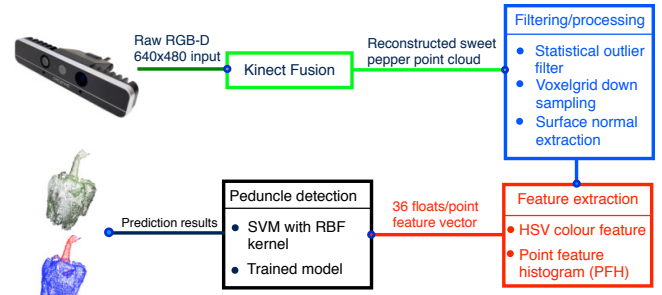


Fig. 6. Software system pipeline for the peduncle detection. Each box denotes a ROS node and its output.

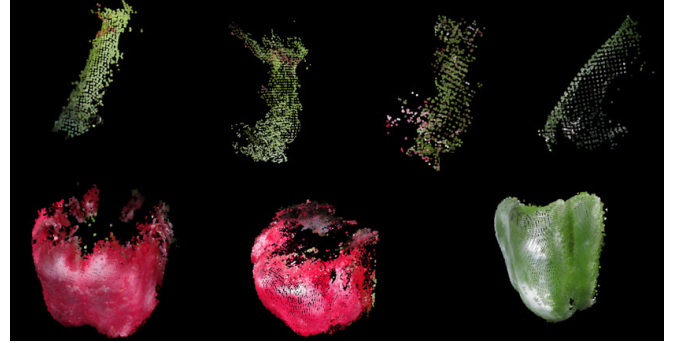


Fig. 7. Instances of manually annotated ground truth datasets for peduncles (top row) and red and green sweet peppers (bottom row). These datasets are available<sup>1</sup>.

an environment closer to the optimum for maximum plant growth and production as shown in Fig. 8. This facility provides relatively even lighting conditions by diffusing incoming sunlight through a semi-transparent plastic/mesh roof and wall.

Data is captured while the robot arm moves through a predefined triangle scanning trajectory around the centroid of a detected red pepper in order to acquire multiple viewpoints (see time of 28 s-55 s in the demonstration video<sup>1</sup>). The scan takes 15 s. During this process, not only is the red sweet pepper (in the centre) reconstructed in real-time but also the adjacent green crops. For each pepper, an RGB-D camera was moved in a scanning motion to view it from multiple directions and the KinFu software library [25] was used to reconstruct a single point cloud.

We manually annotated a dataset with a total of **72** sweet peppers, example images are provided in Fig. 7. It can be seen that these images have a substantial amount of noise present, this is due in part to the low-cost RGB-D sensor as well as shades/reflections from the object surfaces. We randomly divide these 72 samples into two even groups for model training and testing. The output from our classifier is a binary prediction for each 3D point (i.e., 0 is for sweet pepper and 1 for peduncle).

Table I presents the first set split into a **50-50** ratio of training and test data whereas Table II demonstrates a **79-21** split. These datasets are used for system evaluation in Section V-A.

TABLE I  
NUMBER OF ANNOTATED 3D MODELS FOR 50-50 RATIO FIELD TRIP DATASET.

	Train (peduncle+pepper)	Test (peduncle+pepper)	Num. 3D models
Field Trip 1	14 (50%)	14 (50%)	28
Field Trip 2	22 (50%)	22 (50%)	44
<b>Total</b>	<b>36 (50%)</b>	<b>36 (50%)</b>	<b>72</b>

TABLE II  
NUMBER OF ANNOTATED 3D MODELS FOR 79-21 RATIO FIELD TRIP DATASET.

	Train (peduncle+pepper)	Test (peduncle+pepper)	Num. 3D models
Field Trip 1	21 (75%)	7 (25%)	28
Field Trip 2	36 (82%)	8 (18%)	44
<b>Total</b>	<b>57 (79%)</b>	<b>15 (21%)</b>	<b>72</b>

## V. EXPERIMENTAL EVALUATION AND RESULTS

In this section, we present qualitative and quantitative peduncle detection results. To evaluate the performance of the 3D point segmentation, we use the Area-Under-the-Curve (AUC) measure of the precision-recall curve.

### A. Peduncle detection results

Fig. 9 shows the quantitative results of peduncle detection. We note that Fig. 9(a) uses 50% of the available data for training and testing (Table I) whereas Fig. 9(b) has an uneven ratio of 79% for training and 21% for testing (Table II). It can be seen that Field Trip 1 yields a consistent AUC result of 0.73 but that Field Trip 2 varies significantly from 0.54 to 0.67. This is mainly due to the higher number of more challenging green and mixed sweet peppers in the latter. Fig. 9(a) contains 4 green sweet peppers in the training set and 5 green and 2 mixed in the testing set. Fig. 9(b) encapsulates 5 green and 2 mixed for training and 4 greens for testing. Table III summarises different types of sweet peppers (colour coded: mixed=blue) mixed in the training and testing data for various experiments in the two field trips. It is challenging to estimate the optimal ratio of training-testing datasets, especially when the number of samples is relatively low [28]. As a rule-of-thumb in the community, we present results with ratios of 50-50 and 80-20.

We also perform an analysis over three different types of sweet peppers red, green, and mixed colour (red and green). Using the model trained with the 50-50 ratio dataset, sweet peppers in the test dataset are clustered by their colour (red: 29, green: 5, and mixed: 2). There are two mixed sweet peppers which have different amounts of red and green colouration (red=32.5% and 75%, green=67.5% and

TABLE III  
TYPES OF SWEET PEPPERS FOR EACH FIELD TRIP. (THE TEXT COLOUR INDICATES THE SWEET PEPPER COLOUR, MIXED=BLUE).

	Train		Test	
	50-50	79-21	50-50	79-21
Field Trip 1	14, 0, 0	21, 0, 0	14, 0, 0	7, 0, 0
Field Trip 2	18, 4, 0	29, 5, 2	15, 5, 2	4, 4, 0

25% respectively). We then perform the detection shown in Fig. 10. As expected, while red sweet peppers can be easily distinguished, detection results are poorer for green ones (AUC: 0.52). This supports our previous argument on the performance drop for Field Trip 2. Mixed sweet peppers show the intermediate result of only 2 testing samples.

1) *Qualitative evaluation of detection*: Fig.11 displays the qualitative results of our sweet pepper peduncle detection method. We assess whether these outcomes can be utilised in the field. The columns indicate different types of sweet peppers and rows are colour and detection views. For red and mixed sweet peppers, the proposed algorithm reports impressive results. Although sufficient performance is achieved to detect the peduncles of green sweet peppers, there are noticeable false positives,  $F_N$ , that mainly stem from a small margin of sweet peppers and peduncles in colour and PFH features. This paper only presents one view point of 3D model, but the accompanying video shows entire views for further inspection.

### B. Processing time and parameter tuning

Our testing methodology highlighted issues in processing time. For example, it takes about 40 seconds to predict the 6,094 input points, which is unlikely to be practical for autonomous sweet pepper picking. Since predicting input features is fully independent, we thus apply parallel processing, which can submit jobs across multiple processes [29] as shown in Fig. 12. Feature vectors are evenly divided between the number of processes and quadruple speed-ups can be achieved with quad cores.

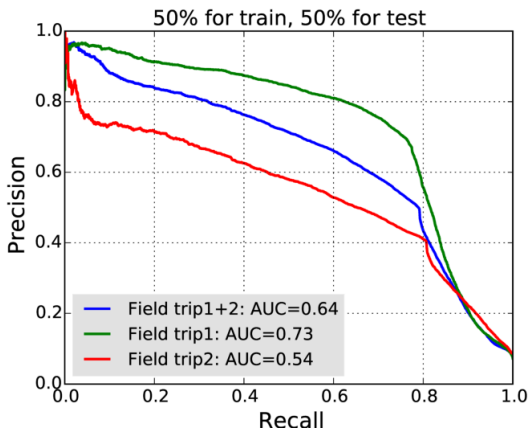
1) *Model (Kernel function) selection and parameter tuning*: The aim of this study is to identify the contributions of features (i.e., colours and 3D point clouds). To achieve this, we set a dataset only with red sweet peppers and vary kernels and combinations of features. Note that the dataset used in this model selection study differ to those in Table II.

We use the Machine Learning Python package [27] that supports a variety of kernel function options such as linear, polynomial, radial basis function (RBF), and even custom kernel functions. It is challenging to pick the optimal kernel and associated parameter sets for all possible combinations since training is very time-consuming (e.g., model training takes about 10 hours for linear SVMs with 50K sample points on a modern Intel i7 computer.). We thus evaluate 16 different cases for 3 common kernel functions (SVC, RBF, and linear) and 2 dominant parameters such as Gaussian kernel and regularisation terms<sup>3</sup> (named  $\gamma$  and  $C$  respectively). Precision-recall curves are calculated for all cases and AUC is used as their comparison metric, with higher AUC numbers implying better detection performance. The RBF kernels with  $\gamma=0.01$  and  $C=100$  report the best AUC score of 0.971, where  $\gamma$  is the inverse of the radius of a single training point accounting for hyperplane estimation and  $C$  defines the impact of mis-classification on the cost

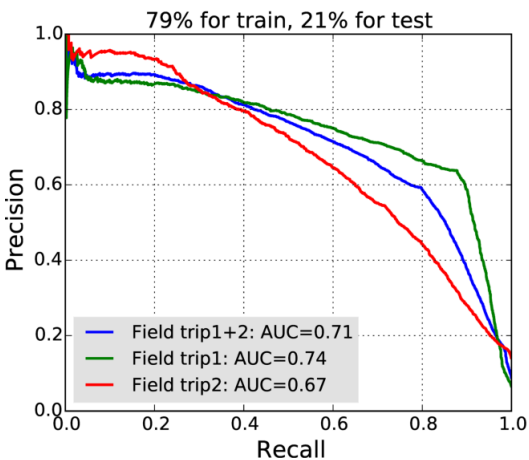
<sup>3</sup>[http://scikit-learn.org/stable/auto\\_examples/svm/plot\\_rbf\\_parameters.html](http://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html)



Fig. 8. A panoramic view of a sweet peppers within a protected cropping facility (left) and the crop layout within it (right). It is evident that the brightness of the left and right regions is significantly different.



(a)



(b)

Fig. 9. Quantitative detection results of precision-recall curves and their AUC when using 50% training and 50% testing dataset (a) and 79%-21% ratio dataset (b).

function. Again, we empirically choose these parameters by performing parameter sweeping (16 evaluations with varying parameters for each kernel (i.e., SVC, RBF, and linear)).

Given this optimised kernel function and parameters, we investigate the impact of each feature on peduncle detection in Fig. 13. We examine the impact of using just colour information (HSV, denoted as green and black in Fig. 13) and colour combined with geometry features (HSV+PFH, in red and blue). Moreover, we study the effects of different kernels with the same features. The plots demonstrate that RBF kernels are superior to their linear counterparts, and colour is the dominant feature for red sweet pepper detection. PFH contributes to improving the detection performance (see

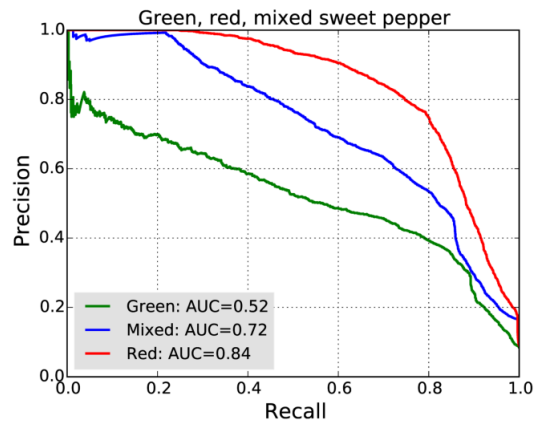


Fig. 10. Quantitative detection results of precision-recall curves and their AUC for different types of sweet peppers; green, red, and mixed colour.

	Red	Green	Mixed
<b>3D model</b>			
<b>Detection</b>			

Fig. 11. Qualitative detection results. The top row indicates the input point clouds and the bottom depicts segmentation results for peduncles (red) and sweet peppers (blue). Each column represents different sweet pepper types.

Fig. 13 at recall level around 0.1). Further experimental results on different sweet peppers types, such as green and mixed colours, are presented in Section V-A.

## VI. DISCUSSION

This paper presents initial, and encouraging, results of peduncle point-level segmentation, which aims at enhancing the performance of a sweet pepper robotic harvester. The integration of the segmentation with the actual harvester and the assessment of its performance is outside the scope of the paper and part of on-going work.

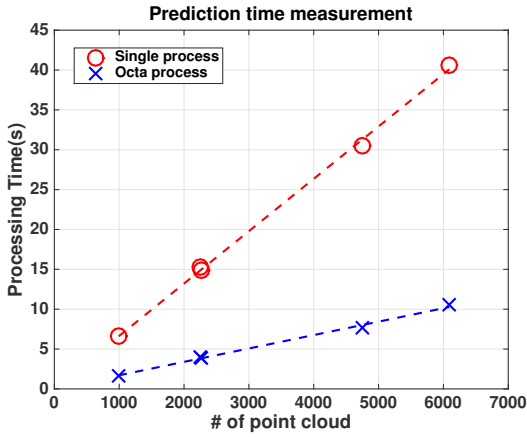


Fig. 12. Processing speed comparison for a single (red) and octa (blue) processes. A quadruple speed improvement is achieved with parallel processing on a quad-core computer.

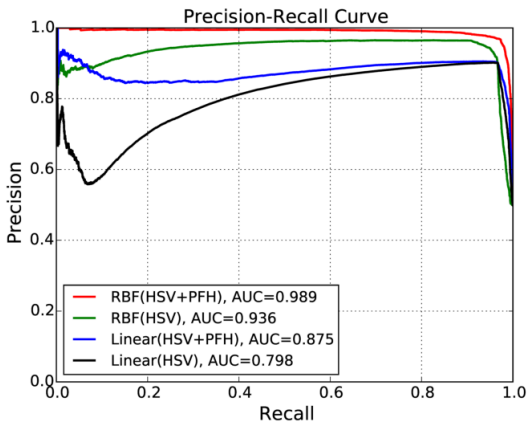


Fig. 13. The impact of PFH features using different kernels for red sweet peppers. Colour and PFH features complement each other and improve detection rate.

A noticeable attribute of the proposed method is that the trained model yields false positives for other stems and leaves that exhibit similar colour and shape to the peduncle as shown in Fig. 14. To overcome this issue, future work could examine the potential of using context features that account for the geometrical relationship between a sweet pepper and a peduncle (i.e., a sweet pepper usually grows the direction of gravity and locates the top of the fruit). The orientation of each segmented peduncle point and its bounding volume can then be computed. With this information, the supervised learning algorithm may be able to determine that peduncles are usually above the centroid within a range of distances, reducing false positive rates. The accompanying video demonstration displays qualitative results with different viewing points<sup>1</sup>.

## VII. CONCLUSIONS AND FUTURE WORK

We present a detection system for sweet pepper peduncles in a farm field site. Colour and shape obtained from a RGB-D sensor, are utilised as dominant features for the detection and we discuss the impact of selecting features and their tuning approaches.

The novel use of PFHs provides further discriminative

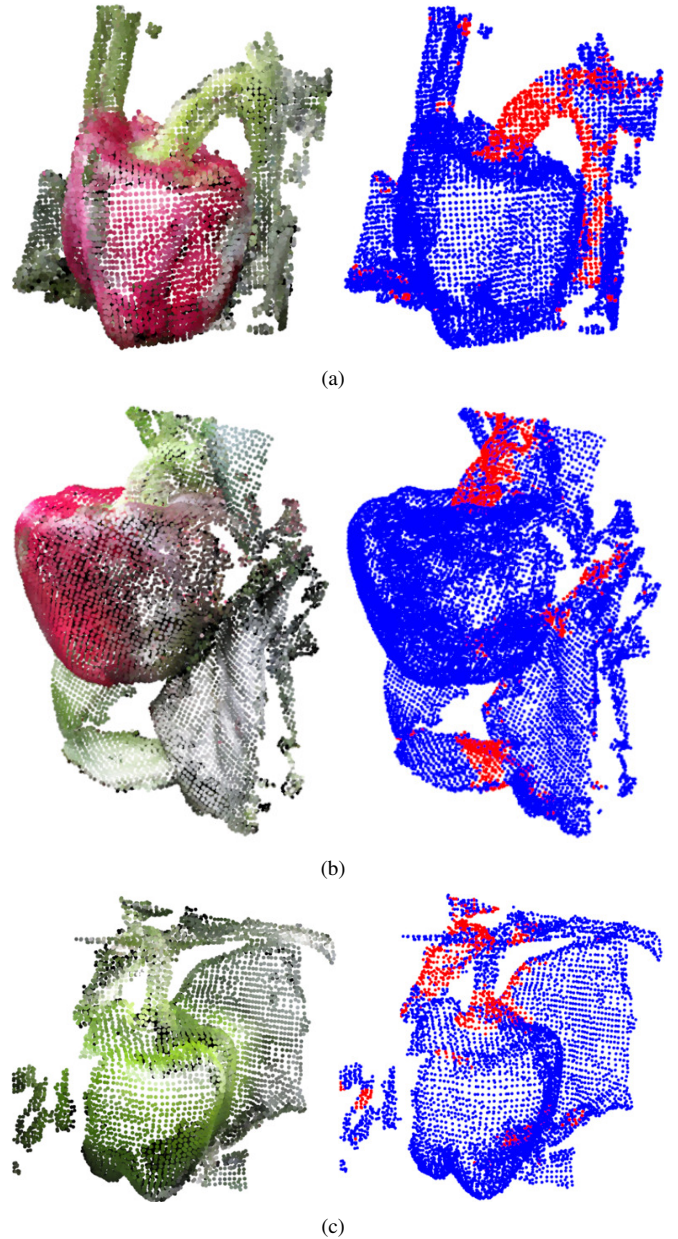


Fig. 14. Qualitative detection results with the background presence for (a) red, (b) mixed, and (c) green sweet peppers.

power and demonstrates feasibility in using depth information for agricultural applications. We present promising qualitative results and quantitatively achieve an AUC of 0.71 for point-level segmentation.

Future work will consider ways of utilising GPU parallel processing for detection speed improvement. Other classifiers such as Random Forest or Gaussian Processes will also be compared regarding their prediction speed and detection rate. With more training data, detection performance could be improved by incorporating different weather conditions in future field trips during harvesting season. Also, to avoid intensive labour work associated with manual annotation, we plan to conduct further data collection field trips in near future. We are interested in applying this system to other sweet pepper varieties or different peduncle types.



## ACKNOWLEDGEMENT

The authors would like to thank Raymond Russell and Jason Kulk of QUT Strategic Investment in Farm Robotics Program for their key contributions to the design and fabrication of the harvesting platform. We also would like to acknowledge Elio Jovicich and Heidi Wiggenhauser from Queensland Department of Agriculture and Fisheries for their advice and support during the field trips. We thank Marija Popović for comments and proofreading that greatly improved the quality of manuscript.

## REFERENCES

- [1] N. Kondo, M. Monta, and N. Noguchi, *Agricultural Robots: Mechanisms and Practice*. Apollo Books, 2011.
- [2] A. English, P. Ross, D. Ball, and P. Corke, "Vision based guidance for robot navigation in agriculture," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 1693–1698, IEEE, 2014.
- [3] D. Ball, B. Upcroft, G. Wyeth, P. Corke, A. English, P. Ross, T. Patten, R. Fitch, S. Sukkariéh, and A. Bate, "Vision-based obstacle detection and navigation for an agricultural robot," *Journal of Field Robotics*, pp. n/a–n/a, 2016.
- [4] I. Sa, C. McCool, C. Lehnert, and T. Perez, "On visual detection of highly-occluded objects for harvesting automation in horticulture," *ICRA*, 2015.
- [5] C. McCool, I. Sa, F. Dayoub, C. Lehnert, T. Perez, and B. Upcroft, "Visual detection of occluded crop: For automated harvesting," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2506–2512, IEEE, 2016.
- [6] C. Lehnert, I. Sa, C. McCool, B. Upcroft, and T. Perez, "Sweet Pepper Pose Detection and Grasping for Automated Crop Harvesting," in *IEEE International Conference on Robotics and Automation*, 2016.
- [7] C. W. Bac, E. J. Henten, J. Hemming, and Y. Edan, "Harvesting Robots for High-value Crops: State-of-the-art Review and Challenges Ahead," *Journal of Field Robotics*, vol. 31, no. 6, pp. 888–911, 2014.
- [8] S. Kitamura, K. Oka, and F. Takeda, "Development of picking robot in greenhouse horticulture," in *Proceedings of the SICE Annual Conference, Okayama, Aug*, pp. 8–10, Citeseer, 2005.
- [9] J. Blasco, N. Aleixos, and E. Moltó, "Machine vision system for automatic quality grading of fruit," *Biosystems Engineering*, vol. 85, no. 4, pp. 415–423, 2003.
- [10] L. A. Ruiz, E. Moltó, F. Juste, F. Plá, and R. Valiente, "Location and characterization of the stem–calyx area on oranges by computer vision," *Journal of Agricultural Engineering Research*, vol. 64, no. 3, pp. 165–172, 1996.
- [11] C. Bac, J. Hemming, and E. van Henten, "Robust pixel-based classification of obstacles for robotic harvesting of sweet-pepper," *Computers and Electronics in Agriculture*, vol. 96, pp. 148–162, 2013.
- [12] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "Deepfruits: A fruit detection system using deep neural networks," *Sensors*, vol. 16, no. 8, p. 1222, 2016.
- [13] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*, pp. 127–136, IEEE, 2011.
- [14] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pp. 3212–3217, IEEE, 2009.
- [15] S. Cubero, M. P. Diago, J. Blasco, J. Tardáguila, B. Millán, and N. Aleixos, "A new method for pedicel/peduncle detection and size assessment of grapevine berries and other fruits by image analysis," *Biosystems Engineering*, vol. 117, pp. 62 – 72, 2014. Image Analysis in Agriculture.
- [16] D. Dey, L. Mummert, and R. Sukthankar, "Classification of plant structures from uncalibrated image sequences," in *Applications of Computer Vision (WACV), 2012 IEEE Workshop on*, pp. 329–336, IEEE, 2012.
- [17] J. Hemming, C. Bac, B. van Tuijl, R. Barth, J. Bontsema, E. Pekkeriet, and E. Van Henten, "A robot for harvesting sweet-pepper in greenhouses," in *Proceedings of the International Conference of Agricultural Engineering, 6-10 July 2014, Zürich, Switzerland*, 2014.
- [18] S. Paulus, J. Dupuis, A.-K. Mahlein, and H. Kuhlmann, "Surface feature based classification of plant organs from 3D laserscanned point clouds for plant phenotyping," *BMC bioinformatics*, vol. 14, p. 238, January 2013.
- [19] M. Wahabzada, S. Paulus, K. Kersting, and A.-K. Mahlein, "Automated interpretation of 3D laserscanned point clouds for plant organ segmentation," *BMC Bioinformatics*, vol. 16, no. 1, p. 248, 2015.
- [20] S. Hayashi, K. Shigematsu, S. Yamamoto, K. Kobayashi, Y. Kohno, J. Kamata, and M. Kurita, "Evaluation of a strawberry-harvesting robot in a field test," *Biosystems Engineering*, vol. 105, no. 2, pp. 160–171, 2010.
- [21] R. B. Rusu, A. Holzbach, M. Beetz, and G. Bradski, "Detecting and segmenting objects for mobile manipulation," in *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pp. 47–54, IEEE, 2009.
- [22] M. Tkalcic, J. F. Tasic, *et al.*, "Colour spaces: perceptual, historical and applicational background," in *Eurocon*, pp. 304–308, 2003.
- [23] A. Aldoma, Z.-C. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R. B. Rusu, S. Gedikli, and M. Vincze, "Point cloud library," *IEEE Robotics & Automation Magazine*, vol. 1070, no. 9932/12, 2012.
- [24] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.
- [25] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 1–4, IEEE, 2011.
- [26] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, p. 5, 2009.
- [27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [28] I. Guyon, "A scaling law for the validation-set training-set size ratio," *AT&T Bell Laboratories*, pp. 1–11, 1997.
- [29] N. Cornelis and L. V. Gool, "Fast scale invariant feature detection and matching on programmable graphics hardware," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*, pp. 1–8, IEEE, 2008.