DOI 10.31256/UKRAS19.7

Model based 3D point cloud segmentation for automated selective broccoli harvesting*

Hector A. Montes, Grzegorz Cielniak, and Tom Duckett**

Abstract— **In this paper we address the topic of feature matching in 3D point cloud data for accurate object segmentation. We present a matching method based on local features that operates on 3D point clouds to separate crops of broccoli heads from their background. We have implemented our approach and present experiments on datasets collected in cultivated broccoli fields, in which we analyse performance and matching capabilities and evaluate the usefulness of the system as a point feature-based segmentation method.**

I. INTRODUCTION

Segmentation of 3D objects in noisy and cluttered scenes is a highly relevant problem. Given a 3D point cloud produced by a depth sensor observing a 3D scene, the goal is to separate objects of interest in the foreground from other elements in the background. This has been extensively investigated in various research fields, such as computer vision, robotics, and pattern matching [14]. In this paper, we focus on 3D point clouds obtained with a structured light 3D camera and favourably compare our results to previously published experiments where sets of points where extracted based on the local proximity of the points. Our approach to this problem uses 3D classification based on point-to-point matching of estimated local 3D features. These features capture information of the local geometry of each point and are compared to the features of its surrounding points. The objective of the work reported in this paper is to research 3D imaging methods to accurately segment and identify broccoli plants in the field. The ability to separate parts into different sets of sensor readings is an important task towards this goal. This research is focused on the broccoli head segmentation problem as a first step towards size estimation of each broccoli crop in order to establish whether or not it is suitable for cutting.

The paper starts by a brief contextual introduction of automated solutions for broccoli harvesting as well as a concise review of related work in Section II. Section III describes the methodology and the data acquisition, while Section IV describes the experimental results along with the evaluation metrics used to assess the overall performance. Section V concludes the paper.

II. HARVESTING BROCCOLI CROPS

Broccoli is a vegetable in the cabbage family that belongs to the Brassica Oleracea plant species. The interest in its cultivation has grown in recent years due to genetic improvement programmes developed in several countries, and to the healthy compounds contained in the crop that have increased its consumption [1]. A consequence of the methods used to breed broccoli is that the heads grow at different rates. This makes them difficult to harvest [2]. Moreover, almost all broccoli is currently harvested by hand, relying on visual grading of size to estimate whether a head can be cut [2]. As a result, only around 50% of broccoli heads can be harvested economically. Two approaches can be readily compared when harvesting crops, namely, slaughter harvesting, i.e. cutting everything in one pass, and selective harvesting, i.e. cutting individually each crop [4]. Slaughter harvesting is not a productive option as it potentially produces large quantities of unmarketable broccoli heads, whereas selective harvesting presents its own challenges as it relies on a subjective assessment by each person cutting the broccoli as to which head is ready. Additionally, labour has become increasingly scarce and more expensive due to a variety of factors ranging from political pressures to migration dynamics [4]. The goal of growing fresh fruit and vegetables is to keep the quality high while minimising costs. It is therefore desirable to find a method to harvest more frequently, more quickly, more accurately, with less waste, and that reduces labour and overall operation costs [3]. Thus, developing an automated method for selective harvesting capable of accurately identify and separate broccoli crops from the background would help to increase productivity and to better control production costs.

A. *Related work*

Automated harvesting systems usually consist of three independent systems: a recognition system to identify and locate the product, a picking system to perform grasping and cutting operations, and a navigation system to allow the robot to move around the cultivated crop plants [4]. One major challenge in autonomous harvesting is the recognition and segmentation of the crop from the rest of the plant. One of the first and common approaches has been to detect crops using 2D images. This can be promptly perceived in the wealth of techniques based on computer vision available in the literature [4,5,6]. For the particular case of broccoli, some approaches have used colour images to separate the broccoli head from the soil and other plant parts. We address the most relevant work below.

Ramirez [7] developed an algorithm to locate the broccoli head within an image of an entire broccoli plant. To locate the head, first the method finds the leaf stems using a threshold, a Canny edge detector, and a Hough transform to extract geometric features that approximate lines that can be fit to the stems. Then the broccoli head can be located based on contrast texture analysis at the intersection of the stems. The method also determined the maturity of the crop using statistical texture analysis. Tu *et al.* [10] published results of a method to grade broccoli heads. The goal was to assess the quality decay of the

* This research has been supported by the UK Agriculture and Horticulture Development Board (AHDB), https://ahdb.org.uk.

** Hector A. Montes, Grzegorz Cielniak, and Tom Duckertt are with the School of Computer Science, University of Lincoln, Brayford Pool, Lincoln, LN6 7TS, UK (corresponding author e-mail: hmontes@lincoln.ac.uk; tduckett@lincoln.ac.uk; GCielniak@lincoln.ac.uk).

Figure 1. Top: Data acquisition with the 3D sensor mounted at the rear of the tractor. The sensor is fixed inside a purpose-built "black box" enclosure to block direct sunlight and other external incidences. Bottom: 3D point cloud images of broccoli plants (far left) are analyzed offline based on local angular features (middle frames) to segment broccoli heads (far right).

harvested crop based on a set of colour and shape parameters. The system determined the area and roundness as the shape parameters and extracted the colour features using standard vision techniques. The resultant quality of the broccoli head was then decided by a neural network classifier. More recently, Blok *et al.* [8] presented a method for detecting and sizing broccoli heads based on computer vision techniques. The method segmented an image based on texture and colour of the broccoli head buds. Firstly, the contrast of the image was enhanced to emphasise high frequency areas, followed by a series of filters and several morphological operations to fine-tune the image. Then, pixel connectivity was used to generate connected green-coloured components. Lastly, a shape-based feature selection on the connected area was conducted to separate small non-connecting components from the foreground. The segmented heads were sized using circle templates, and the mean image processing time took a little less than 300 ms. The system was part of a prototype harvesting device attached to a modified tractor and was tested in cultivated broccoli fields reaching an accuracy of 94%. Kusumam *et al.* [9] documented a system for detecting and locating mature broccoli heads in cluttered outdoor field conditions based on depth images acquired by a low-cost RGBD sensor. The paper evaluates a combination of Viewpoint Feature Histograms (VFH), a Support Vector Machines (SVM) classifier, and a temporal filter to track the detected heads. Their results showed a precision rate of 95.2% and 84.5% on datasets collected from fields in the UK and Spain, respectively.

Although 2D imagery is clearly important, this paper focuses on effective 3D depth features. However, 2D features is an interesting addition to 3D data worth studying in future work.

III. METHODOLOGY

A. Point cloud data acquisition

The 3D point cloud data for our experiments was captured in outdoor fields under different weather conditions using the well-known Kinect 2 sensor (1920×1080 RGB, and 512×424 depth resolution). The sensor was fixed inside a specially constructed enclosure to block direct sunlight and to protect during rainy conditions. The point cloud data was collected with the camera enclosure mounted on the rear of a tractor, as shown in 0

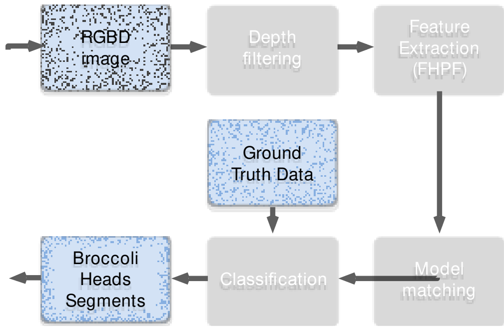

Figure 2. 3D point cloud segmentation pipeline. The frames of 3D point cloud data are first filtered by depth. Then features are extracted from each point and matched to the reference models. The points are then classified using a decision function. The points with the same target class are grouped to form the final segments.

B. Model based 3D point cloud segmentation

Two approaches are common in 3D object segmentation methods. In the first approach, the scene is segmented into smaller regions and *global* features are computed for each segment. These features are then matched to the descriptors of a model. In contrast, *local* methods commonly locate a list of appropriate points, often referred to as *interest points* or *key points*, and extract a set of features in the vicinity of those points. The points are then matched to a model and the correspondences are grouped according to the geometry of the model. This paper applies a local recognition method for segmenting broccoli heads in sets of 3D point cloud images collected in planted broccoli fields. The method processes the depth data in a pipeline of four stages: point cloud depth filtering, feature extraction, model matching, and classification, as shown in 0

The point cloud data captured by the sensor is first filtered to remove visible parts of the soil and other noisy points that are too distant from the surface of the scene. Feature descriptors are then computed in the remaining points and matched with the model references to finally determine if the points belong to a broccoli head. We use the algorithms available as part of the PCL C++ library [12] for processing point clouds.

B.1 Depth filtering

Depth filtering of the soil and other distant points is achieved through a simple depth range thresholding of the input point cloud. The points that lie outside the desired range are simply discarded. The depth threshold is defined to be of *1 m* and is based on the distance of the sensor to the ground measured during data collection.

B.2 Feature extraction

We use a set of local 3D feature descriptors that are extracted for individual input points. Local descriptors are often used for object recognition and image registration. Even though descriptors have no notion of what object the points belong to, they do describe how the local geometry is around

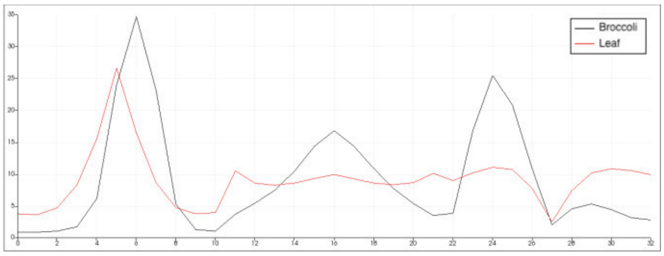

Figure 3. Histograms of the reference models used in our algorithm. A FPFH descriptor is computed for each data point. The descriptor is then matched to both reference models and the difference provides the final classification score.

that point. Every feature descriptor should be discriminative with respect to the two given reference models, i.e., broccoli and non-broccoli points. To this end, we use the *Fast Point Feature Histogram* (FPFH) descriptor introduced by Rusu *et al.* [13].

B.2.1 Fast Point Feature Histogram

We briefly summarize here the structure of the FPFH descriptor. The interested reader is referred to [13] for extended details on the descriptor and a discussion on its relevant properties. The FPFH captures information of the local geometry of the point by analysing the difference between the directions of the normals in its surrounding area. The distribution of the surface normal directions should encode the underlying geometry of the broccoli heads and be discriminative compared to that of other elements in the scene. The FPFH derives from a more computationally expensive descriptor called Point Feature Histogram (PFH) [13]. To calculate the FPFH, the algorithm pairs the current point to all the points in the vicinity, and for each pair, a fixed coordinate frame is computed from their normals. The direction differences between the normals can then be encoded into three angular variables between the normal and the three axes of the fixed frame. These variables are then binned into a histogram when all pairs have been computed. The histograms of the neighbours are merged with the current point histogram, weighted according to their distances. The final FPFH descriptor is the concatenation of the histograms of each angular variable.

B.3 Classification

To compare two FPFH descriptors, we use a fast but effective measure between the descriptors of two points: the *histogram intersection*. Given a query point and its corresponding histogram H (its descriptor), and a reference histogram h calculated from the descriptors of sets of known models, the histogram intersection is defined as:

$$d(H,h) = \sum_{i=1}^{n} min(H_i, h_i) \tag{1}$$

where n is the number of histogram bins. For the histogram intersection, higher values are better. After the intersections are calculated between the query and the reference descriptors, the difference between both intersections is used to make a choice as to which model is the best match and provides the final classification score. This score is computed with point-to-point correspondences obtained by matching local descriptors of feature points to a set of known reference models. Consecutive cloud points with similar high score are part of the same point cloud segment.

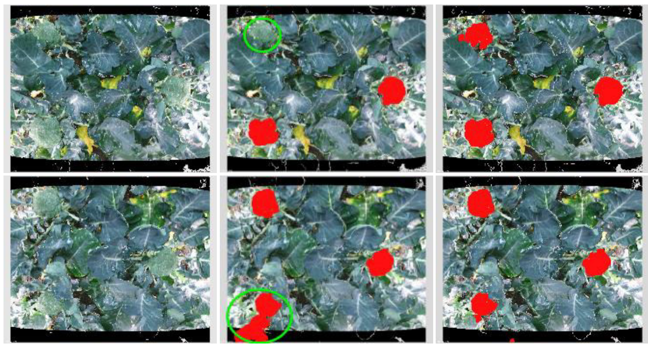

Figure 4. Two segmentation samples on different frames. First column: The original frame. Center column: circled in green, under segmented (missed one head) and over segmented objects (not only the head) using the Euclidean Clustering method from [9]. Right column: Samples of our model-based segmentation method.

C. Reference models

We construct models for the 3D point cloud objects that we are interested in. The reference models are two FPFH descriptors calculated from the histograms of sets of known models. These models are sets of selected 3D points that are already labelled to be part of either a *broccoli* head or a *leaf* (although labelled as *leaf*, the points also include other elements in the scene that are not part of a broccoli head). 0shows a plot of the FPFH reference descriptors selected for our experiments. These two descriptors suffice for our segmentation purposes as the angular distributions of a broccoli head is more relevant for the classification and segmentation tasks.

IV. EXPERIMENTAL RESULTS

The experimental evaluation aims to determine the overall performance and the accuracy of the method. To evaluate our results, we use a set of the same point cloud dataset used in the experiments reported by Kusumam *et al.* [9]. We particularly focus on frames where those experiments produced either *under segmentation*, i.e., some of the target broccoli heads were missed, or *over segmentation*, i.e., the extracted segments were larger that the broccoli heads seen in the scene frame. A sample of these two cases is shown in 0

The frames of a 3D point cloud are first filtered by depth and the FPFH descriptors are computed for each point and then matched to the reference models. These initial steps already show the areas of the point cloud that are more likely to contain broccoli heads. This allows to classify every point using a simple decision function that takes the current point matching score and determines the best label class. A function that examines nearby points of the same class forms the final segments and helps to eliminate false positives. An illustration of the broccoli segmentation steps is shown in 0

A. Classifier evaluation

We evaluated the classifier of the 3D system pipeline for segmenting broccoli heads using individual FPFH descriptors. For each point in the current frame a classification score is produced. If the point is part to a broccoli head, according to

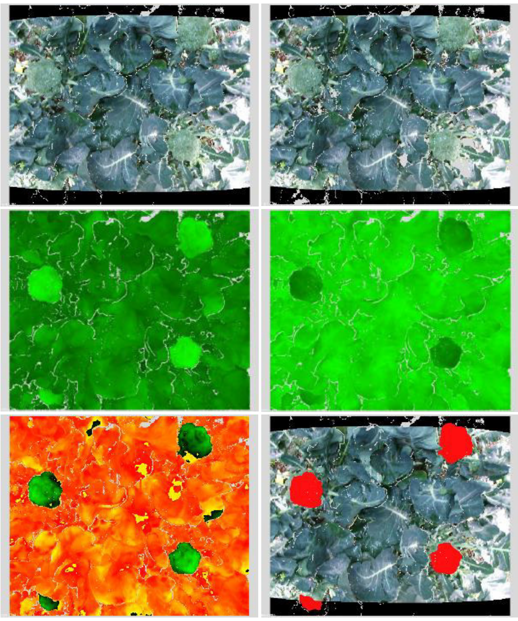

Figure 5. Broccoli segmentation steps. Top row: The original frame on the left, and after been depth filtered on the right. Mid row: matched points of broccoli heads shown in bright green on the left, and other points shown on the right. Bottom row: The difference of the two reference models shown in contrasting colors on the left and, on the right, the extracted segments in red.

ground truth data, it is labelled as a positive sample; otherwise it is labelled negative. The resultant sample sets are highly unbalanced, i.e. there is a significant difference in the number of positive and negative samples. In this case, the negatives notably outnumber the positives as large portions of each point cloud frame are from leafs, soil or other elements. Classification results were evaluated using precision-recall curves (PRC), as they provide a more accurate interpretation of a classifier performance on unbalanced samples [11]. *Precision* represents a ratio of true positive detections to the total number of positive detections (true and false), whereas *recall* is a ratio of true positive detections to the total number of both true positive and false negative detections. The precision and recall values are computed over a range of discrimination threshold values across the classification scores. 0shows the performance evaluation on a PRC plot on the set of classified points. The plot shows the average precision results on the scores computed for every point. The results show a precision rate of 92.22% on the datasets examined by our model-based method, versus a precision rate of 73.20% of the results published in [9]. Originally the EC method was applied on clusters of points and global features were extracted. The PRC plot shows the performance of the same method reflected on all the points that form the same clusters.

V. CONCLUSION

In this paper, we discussed a method for 3D point cloud segmentation based on 3D feature descriptors matching. Comparative experimental results show that our method performed favourably against an existing 3D broccoli detection algorithm based on the Euclidean proximity of 3D points when tested on the same dataset. The results showed a promising precision score. Moreover, our results also showed that the segmentation method can be used to detect broccoli heads, as a first step in the development of a fully autonomous selective harvester. Interesting future research directions include a more principled selection of key points to be examined by, for instance, performing a point cloud compression that retains the original perceived distribution of 3D points. Another research direction is to adopt strategies to find or even synthesize descriptors for the reference models, so they better encode the properties of the broccoli heads we are interested to segment. Future work will also address the issues of developing a real-time implementation of the presented approach for deployment on open field conditions.

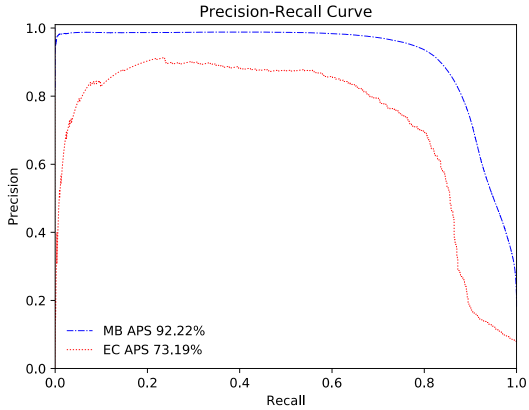

Figure 6. Precision-Recall curve showing the classification performance of the segmented points. The value shown is the average precision score (APS) at various discrimination threshold settings. The plot shows the performance of Euclidean Clustering (EC) from [9] and our Model Based (MB) approach.

REFERENCES

[1] L. Maggioni, R. von Bothmer, G. Poulsen, and F. Branca. Origin and domestication of cole crops (Brassica oleracea L.): linguistic and literary considerations. Economic botany, 64(2):109–123, 2010.

[2] M. D. Orzolek, W. J. Lamont, L. F. Kime Jr., and J. K. Harper. Broccoli production. In Agricultural Alternatives series, Agricultural Alternatives series. Penn State Cooperative Extension, 2012.

[3] C. Wouter Bac, Eldert J. van Henten, Jochen Hemming, and Yael Edan. Harvesting robots for high-value crops: State-of-the-art review and challenges ahead. Journal of Field Robotics, 31(6):888–911, 2014.

[4] S. Bachche. Deliberation on design strategies of automatic harvesting systems: A survey. Robotics, 4(2):194–222, 2015.

[5] A. R. Jimenez, R. Ceres, and J. L. Pons. A survey of computer vision methods for locating fruit on trees. Trans. ASAE, 43(6):1911, 2000.

[6] Y. Zhao, L. Gong, Y. Huang, and C. Liu. A review of key techniques of vision-based control for harvesting robot. Computers and Electronics in Agriculture, 127:311–323, 2016.

[7] R. A. Ramirez. Computer vision based analysis of broccoli for application in a selective autonomous harvester. Master thesis, Virginia Polytechnic Institute and State University, July 2006.

[8] P. M. Blok, R. Barth, and W. van den Berg. Machine vision for a selective broccoli harvesting robot. IFAC-PapersOnLine, 49(16):66–71, 2016. 5th IFAC Conference on Sensing, Control and Automation Technologies for Agriculture AGRICONTROL 2016.

[9] K. Kusumam, T. Krajník, S. Pearson, T. Duckett, and G. Cielniak. 3D-vision based detection, localization, and sizing of broccoli heads in the field. Journal of Field Robotics, 34(8):1505–1518, 2017.

[10] K. Tu, K. Ren, L. Pan, and H. Li. A study of broccoli grading system based on machine vision and neural networks. In Intl Conference on Mechatronics and Automation, pp 2332–2336. IEEE, 2007.

[11] T. Saito and M. Rehmsmeier. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PLoS ONE, 10(3), 2015.

[12] R. B. Rusu and S. Cousins. 3D is here: Point Cloud Library (PCL). In Proc. ICRA. Shanghai, China: 2011.

[13] R. B. Rusu, N. Blodow, and M. Beetz. Fast Point Feature Histograms (FPFH) for 3D Registration. In Proc. IEEE ICRA, Kobe, Japan. 2009.

[14] U. Castellani, M. Cristani, S. Fantoni, and V. Murino. Sparse points matching by combining 3D mesh saliency with statistical descriptors. In Computer Graphics Forum, 27:643–652.