This CVPR 2020 workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

# Segmentation and detection from organised 3D point clouds: a case study in broccoli head detection

Justin Le Louedec, Hector A. Montes, Tom Duckett, Grzegorz Cielniak
University of Lincoln
{jlelouedec,hMontes,tduckett,GCielniak}@lincoln.ac.uk

## Abstract

*Autonomous harvesting is becoming an important challenge and necessity in agriculture, because of the lack of labour and the growth of population needing to be fed. Perception is a key aspect of autonomous harvesting and is very challenging due to difficult lighting conditions, limited sensing technologies, occlusions, plant growth, etc. 3D vision approaches can bring several benefits addressing the aforementioned challenges such as localisation, size estimation, occlusion handling and shape analysis. In this paper, we propose a novel approach using 3D information for detecting broccoli heads based on Convolutional Neural Networks (CNNs), exploiting the organised nature of the point clouds originating from the RGBD sensors. The proposed algorithm, tested on real-world datasets, achieves better performances than the state-of-the-art, with better accuracy and generalisation in unseen scenarios, whilst significantly reducing inference time, making it better suited for real-time in-field applications.*

## 1. Introduction

Due to population growth and various social and economical factors, the interest in automation of agriculture has grown worldwide. Labour for harvesting is one of the biggest challenges facing this industry. Building autonomous system able to detect, analyse and pick crops is rapidly becoming a necessity, both economically and socially.

Perception for harvesting applications presents numerous challenges characteristic to outdoor scenes, such as difficult lighting conditions and occlusions of the target crop caused by the plant growth. In this paper, we focus on a perception system for the detection and segmentation of broccoli crops. Such a system must be precise both in terms of localisation and segmentation, and fast to allow rapid analysis and real-time operation. We propose a 3D vision approach, which is perfectly suited for tasks involving object geometry. We present a novel approach using 3D information for detecting broccoli heads based on Convolutional Neural Networks (CNNs), and evaluate its performance on publicly available datasets and compare the results with the recent state-of-the-art methods.

Typical 3D vision systems have been successfully applied to numerous indoor scenarios featuring relatively large man-made objects with distinguishable shapes such as furniture, rooms or offices [15, 12, 7]. When applied to an agricultural context in outdoor scenarios, however, these methods struggle to achieve satisfactory results [14], due to the noisy and sparse character of the 3D data originating from popular off-the-shelf RGBD sensors. In this work, we propose to overcome this limitation by exploiting the organised nature of the RGBD point clouds, and employ a CNN-based architecture to learn segmentation in a supervised manner. The neural network learns shape and manifold information in the point cloud by using the grid as a medium for grouping and sampling.

In particular, the contributions of this paper are as follows: 1) a novel technique and application of a CNN architecture to organised 3D point clouds for the task of object detection and segmentation; 2) improvement of the generalisation capabilities through unique data augmentation techniques, including spatial translation and rotation; 3) experimental validation and comparison to state-of-the-art engineered solutions on agricultural data collected from real fields of broccoli. Our system achieves on par to better performance in terms of accuracy, segmentation and localisation, with a better generalisation for the most difficult datasets. We also achieve very high inference speeds (50∼60fps), making our approach more suitable for real-world application. Section 2 presents related work whilst the system overview and methodology is presented in Section 3. We showcase quantitative and qualitative results, together with a comparison to the state of the art in Section 4, before concluding the paper in Section 6.

## 2. Related Work

Autonomous robotic harvesting presents numerous challenges, such as identification of crops, localisation, segmentation and analysis, which require fast operating speed. 2D vision has been the main focus in the literature so far [3, 20]. 3D sensors, however, can provide better localisation, size estimation and other analysis related to shape, while increasing computational requirements and algorithmic challenges. Recently, deep learning techniques have been successfully used in various applications in agriculture [8]. Bender *et al*. [1] used Convolutional Neural Networks (CNNs) for the detection of broccoli and cauliflower, achieving good performance (0.95 Mean average Precision or MaP), but for detecting the entire plant (leaves included) from the ground, rather than just the head, as required for harvesting applications as in this paper. Several studies can be found on the detection of broccoli. Ramirez *et al*. [16], developed a contrast-based algorithm on a set of 13 colours images and Blok *et al*. [2] proposed a new method based on colour and texture filters. In more recent work, Kusumam *et al*. [10] proposed the detection of broccoli using RGBD sensors, making use of 3D information by combining Euclidean clustering and a Viewpoint Feature Histogram (VFH) descriptor as input to a Support Vector Machine (SVM). They report an average precision of 0.952 and 0.845 for two broccoli varieties, but with a processing time of around 6 s per frame.

The most popular approaches for 3D object segmentation and detection were designed around processing unordered point clouds, such as in PointNet++ [15] or PointCNN [12]. The methods rely on sampling and grouping of points to learn surfaces, manifolds and shape. Such an approach, however, struggles with noisy, cluttered and complex 3D information characteristic to agricultural applications [14]. The alternative is to make use of the grid structure offered by the organisation of the data resulting from RGBD sensors. This can be achieved either by segmenting directly in 2D and projecting the values into 3D space as in [19], or by directly processing 3D information in a grid. [11] chose to represent the information from the point cloud in a grid and encoded occupancy with a simple Boolean value. Using a standard CNN architecture, they achieved good performance in objection detection with the KITTI dataset (a standard benchmark for 3D vision in autonomous driving).

In contrast to the prior work, we propose a new approach to organised point cloud processing, using a CNN directly applied on the points and normals, placing a bigger emphasis on localisation, shape, and object structures. This approach addresses the problem encountered with unorganised approaches [15], with a faster inference time and better feature extraction with facilitated grouping of points. The proposed approach improves on the current state of the art using classical methods for broccoli head detection [10] in terms of speed, generalisation and segmentation accuracy.

## 3. Method

Our approach for the segmentation and detection of broccoli heads uses organised point clouds originating from RGBD sensors. We train a CNN autoencoder for the task of semantic segmentation using 3D information. To avoid over-fitting and improve generalisation between different varieties and field conditions, we use several data augmentation techniques. The resulting segmentation results are transformed into instances using the connected components algorithm. Figure 1 provides a general overview of the system and indicates the core components described in detail in the following sections.

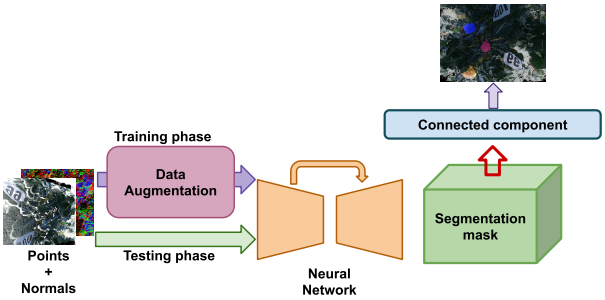


Figure 1. An overview of the system for detecting broccoli heads from the organised point clouds.

### 3.1. Processing organised point clouds

Typical methods for un-ordered point clouds such as PointNet++ [15] make use of 2D convolutions to learn local features from sampled and aggregated points. The spatial information and correlation is highly dependent on the sampling and grouping algorithms, which is an active area of research [12, 7]. For RGBD sensors, however, the data is captured from a single point of view and results in a spatial organisation and grouping of points through the image grid. Based on these preliminary assumptions, we can directly make use of 2D convolutions and traditional CNN architectures directly applied to a point cloud through its grid. We argue that surface and manifold can be retrieved successfully from the points in the grid with the use of convolutions and pooling functions.

Surface and manifold properties are often derived from normal information. Processing points using convolutions should lead to filters dedicated to computing these normals and extracting relevant information from them. Such information is easy to compute in a grid, however, and using it as an input on top of spatial information should improve the learning of local and shape features by the neural network.

Figure 2. Data collection from real fields of broccoli, taken from [10] (row 1) and resulting RGBD data together with segmented annotations collected in Spain (row 2) and UK (row 3 and 4).

As our work is focused on detection and segmentation using 3D information, we use only the points and normals as input, organised in the image grid. Figure 3 depicts an example organised point cloud and the associated normal map. Note that colour is only used for visualisation and all of the presented methods work with 3D point data only. For normal calculation, we use integral images, making use of the grid organisation of the points as presented in Hozier *et al*. [5]. We restrict ourselves to a low number of neighbours, making the computation time overhead insignificant (44ms per frame on average).

### 3.2. Neural Network

For the semantic segmentation task, we chose a classic auto-encoder architecture inspired by U-net [17], with the

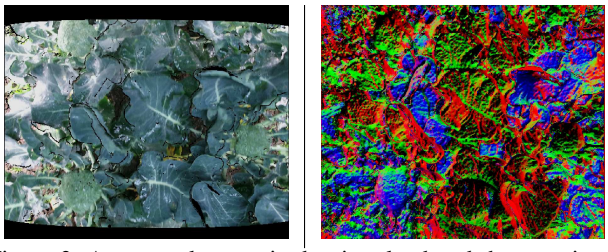

Figure 3. An example organised point cloud and the associated normal map.

encoder part responsible for extracting relevant features and the decoder part for transforming them into the correct class prediction. As presented in Figure 4, we use skip connections between the encoding and decoding part of the architecture to make use of multi-scale features in the segmentation process. The input is composed of 6 features (XYZ position of each point and their 3-component normals), which are compressed into a $512 * W * H$ feature map in the latent space of the network, before being decoded into the segmentation mask. We use a standard VGG16 architecture, saving the feature maps and pooling indices at the end of four different convolution blocks. We use these indices in the decoder part of the network to up-sample the feature maps. The saved feature maps are added to decoded feature maps in the decoder, to introduce multi-scale features and improve the extraction of point clusters corresponding to broccoli heads. Due to the compact nature of the CNNs, the inference time exceeds real time.

Since the neural architecture employed for this work was developed for the task of semantic segmentation, we use a connected component algorithm over the points in the image grid, processed as a binary mask. We remove clusters too small to be broccoli heads (lower than 200 points) and drop segmented values with a probability $< 0.5$.

### 3.3. Data augmentation

Training the network only on point coordinates leads to fast learning with a quick convergence toward dataset-specific over-fitting. Training only on one dataset leads to excellent results ($\sim 0.99$ Mean average Precision) but offers little to no generalisation when applied to unseen scenarios. As the position of broccoli heads in each dataset is changing linearly, the network tends to learn their position and change in position. To counteract this, we first add the normal information on top of the points coordinates, and chose to apply various data augmentation techniques during the training phase. Since the data is contained in a 2D grid, we can easily apply rotations in this grid to avoid localisation over-fitting. Similar rotations also need to be applied to the points and normals to avoid discrepancies between the grid and the spatial coordinates and orientation. We can also add translations over the points, adding more diversity

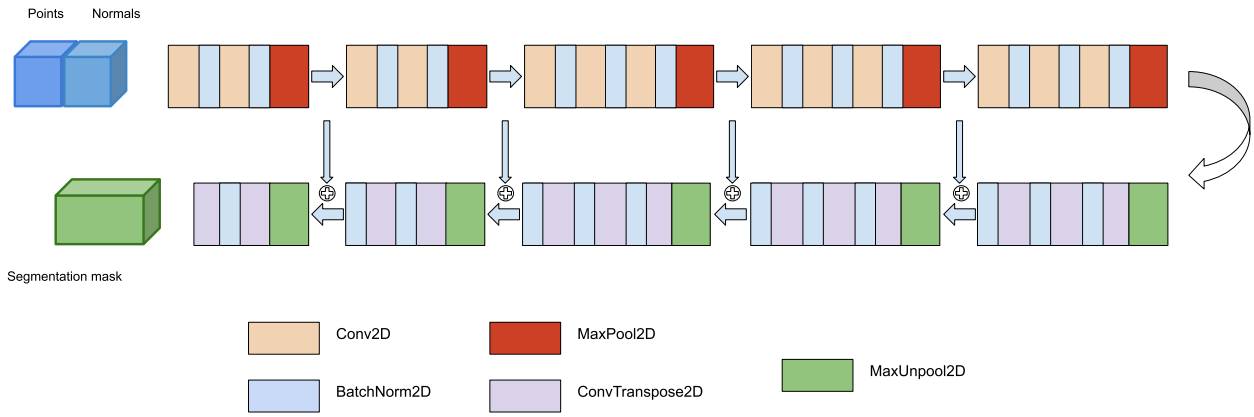


Figure 4. The neural architecture (NN) employed for segmentation in organised point clouds.

| Dataset Name | Spain | UK1 | UK2 |
|---|---|---|---|
| #Point clouds | 300 | 300 | 300 |
| Overlapping | 94% | 95% | 90 % |
| Average Width | 1.10m | 0.60m | 0.60m |
| Average Height | 0.86m | 0.80m | 0.80m |
| Average Distance | 0.81 | 0.75 | 0.75 |
| Broccoli Species | Titanium | Ironman | Ironman |

Table 1. Summary of the dataset characteristics. All datasets share the same resolution ($512x424$) and are annotated following an instance segmentation format.

to the object positions. These three augmentations allows us to add more diversity to the object locations. For every frame in the training set, we rotate the points in space and in the grid by a random angle between $-180$ and $180$ degrees. We also translate the point cloud by a random value between its minimum and maximum on every axis.

### 3.4. Baseline Method

We use the method published by Kusumam *et al*. [10] as a baseline method to compare and contrast our results. To this end, we have implemented a more efficient version of this method called here Fast Euclidean Clustering (FEC). The detection pipeline from [10] includes: 1) statistical outlier removal, 2) depth-range filtering, 3) Euclidean cluster extraction, 3) normal estimation, 4) feature extraction, 5) classification, and 6) a temporal filter to further improve the classification results. In the FEC pipeline, the statistical outlier removal and the temporal filter steps were discarded. Removing outliers is computationally expensive and the filtered points have a negligible effect, while the temporal filter improved classifier predictions by only $0.5\%$. FEC works by first filtering out points which are too close or too far away from the sensor. Clusters are then formed by iteratively grouping points together within a predefined radius, *i.e*., for every point added to a cluster, its neighbours are also checked for the distance restriction until no more new points can be added. Each cluster is then added to the list of clusters if they are within a valid size. Normal vectors are then computed for each point using the much faster integral images normal estimation method [6]. The algorithm uses the inherent grid structure of the point clouds as collected by the RGB-D sensors we used. This allows to quickly create rectangular areas over which the normals are computed without the need for costly euclidean space searches. For each extracted cluster, a Viewpoint Feature Histogram (VFH) descriptor [18] is computed. The VFH is a global 3D feature descriptor that uses the surface normal directions to encode the underlying geometry of an object. Finally, these descriptors become the samples to be predicted as either positive (broccoli) or negative (leaf, soil, etc.) using a Support Vector Machine (SVM) classifier.

## 4. Evaluation setup and data collection

Our project aims at providing reliable computer vision algorithms for the detection, segmentation and analysis of broccoli heads in 3D. Capturing and annotating 3D data is a very challenging task. This leads to a lack of available data in agriculture and in general. Often with RGBD sensors, the RGB picture is annotated and the masks generated transposed to the aligned depth image. However in case of wrongly aligned frames, spatial information in the point cloud is wrongly annotated and miss-leading.

Captured on different crops from various countries, the initial datasets were first reported in the previous work of Kusumam *et al*. [10] (see Figure 2). The datasets were captured using the Kinect 2 sensor with two different resolutions : $1920 \times 1080$ for RGB and $512 \times 424$ for depth. When de-projecting the depth data into a point cloud, the

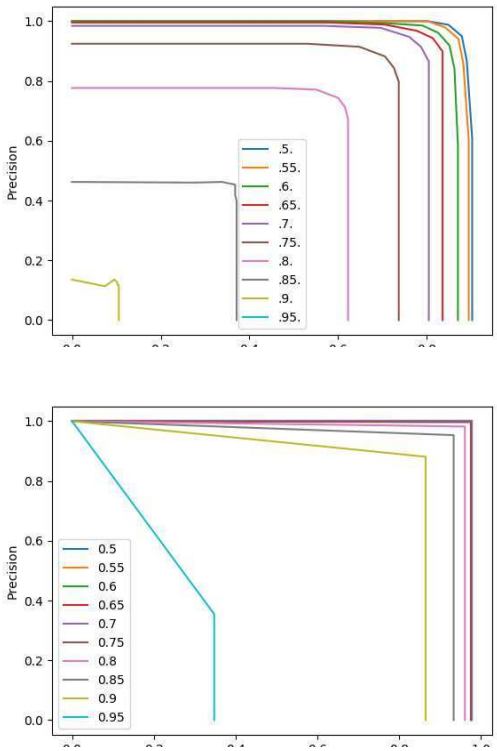


Figure 5. Precision-recall curves for the FEC (top) and the neural network (bottom) trained and tested on the Spain dataset and the range of IoU threshold $[0.5 : 0.95]$.

RGB information is downsampled and aligned to the depth map resolution. Enclosed in a box mounted on a farm tractor with some additional LED lighting, the sensor pointed downward toward the broccoli crops. The box helped to create a more controlled environment as RGBD sensors use the infrared spectrum, and suffer from exposure to sunny environments. The datasets were first captured in Spain on a variety called "Titanium" for about 300 frames with around 94% overlap. The second part of the dataset was captured in the UK, consisting of 600 frames of the broccoli variety "Ironman", with around 95% overlap for half of them and 90% for the rest. We present the dataset characteristics in Table 1 and show some examples in Figure 2. The broccoli varieties in the two datasets are noticeably different in terms of shape, size and localisation, but share some common features. The two black bands at the top and bottom of the captures are due to the alignment of the RGB frame to the depth frame, and the RGB edge distortion is also visible. Even if RGB information is missing on the edges, depth information and their de-projected points are still present. Here the usage of 3D vision over 2D approaches is clear. Due to miss-alignment of the RGB to the depth information as seen in Figure 2, processing it directly will lead to wrong localisation, shape and extraction of the

| | FEC | NN | FEC | NN | FEC | NN |
|---|---|---|---|---|---|---|
| Trained / Tested | Spain | | UK1 | | UK2 | |
| Spain | 0.65 | **0.87** | 0.56 | **0.79** | 0.61 | **0.81** |
| UK1 | **0.94** | 0.76 | **0.96** | 0.95 | 0.92 | **0.93** |
| UK2 | **0.93** | 0.76 | **0.92** | 0.91 | 0.92 | **0.93** |
| Mean | **0.84** | 0.80 | 0.81 | **0.88** | 0.82 | **0.89** |

Table 2. Comparison of the *MaP* for the instance detection masks. We compare FEC and the neural network, with the dataset used for training at the top, and the one used for testing on the left side. We also show the average performances for each training set.

objects. There is also a slight variation in orientation of the camera between the Spain and UK sets and the main differences between Spain and UK dataset lies in the Broccoli's heads sizes. The Spain dataset even though smaller offers a greater challenge with smaller occluded crops. Also, borders of the point cloud present distortion due to the light and sensor, which affect crops found in such areas. For all datasets we have an instance segmentation annotation, where each point has a class associated to it (background and broccoli) and each object a different number identifying it in the point cloud. The points are directly annotated in the point clouds, making the annotation, a true 3D annotation.

### 4.1. Training and evaluation

We separate each of the above datasets into two different sets for training and testing our various algorithms. We take 75% of each for training and 25% for testing. We also test each algorithm on the 25% test set from the other datasets, to analyse the generalisation performance of our algorithms and their shortcomings. We use the Adam optimiser [9], with a starting learning rate 0f 0.0001. We reduce the learning rate by 0.7 every 200 epochs, and stop the training when the loss stop decreasing significantly. We use an NVIDIA 1080Ti GPU (Graphics Processing Unit), for training and testing the neural network and an Intel i7 4790 for the CPU code which handles I/O operations and pre-processing stage.

For the application of perception algorithms to harvesting, we are mainly interested in the accuracy of our algorithms, their precision, the generalisation between different locations and crop species, and the inference speed. We decide to evaluate both semantic segmentation and instance segmentation for two main reasons. First we are aiming at detecting broccoli heads for harvesting using instance segmentation. We can reflect better on the quality of the detection and their accuracy in terms of missing detection and false detection. Secondly, the mask from detection and extraction of the broccoli head from the background is very important for harvesting. Evaluating the segmentation only

adds more nuance to this problem and our performances, and allows us to assess the quality of the masks extracted.

To evaluate our algorithms we use two different metrics: Mean average Precision (MaP) and Mean Intersection over Union (MIoU). For MaP, we use the definition from MS COCO [13], where it represents the average of the precision over all samples and classes for IoU thresholds in the range of $[0.5 : 0.95]$, with the definition of Precision being: $\frac{TP}{TP+FP}$ (TP = number of True Positives, FP = number of True Positives). MIoU is the average of the IoU defined as $\frac{Intersection}{Union}$ of the segmentation masks, for all samples.

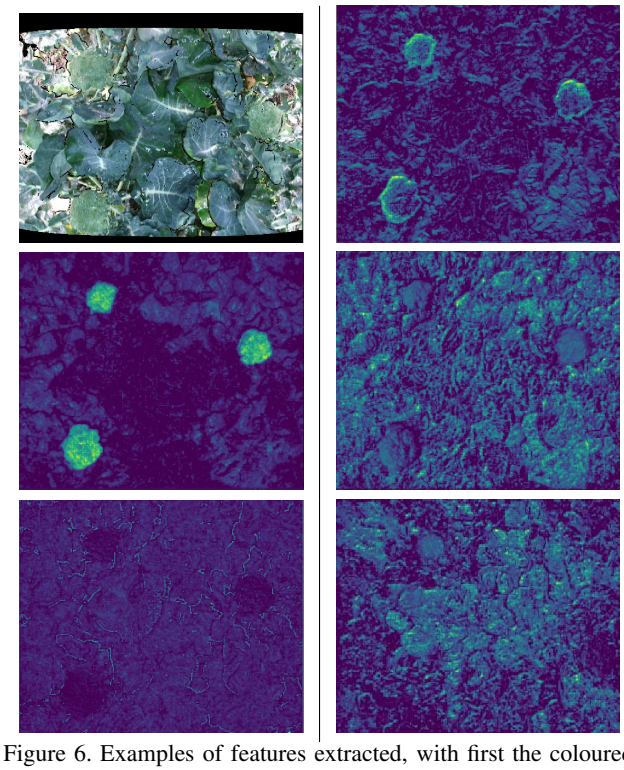

Figure 6. Examples of features extracted, with first the coloured version of the point cloud given as an input to the network (we use the colour of the point cloud for visualisation only)

## 5. Results

We compare our NN to a baseline solution (FEC) taking into account point and instance segmentation, and appraise the performance of both solutions on the collected data.

### 5.1. Instance segmentation

We present in Table 2 the different results obtained using the neural network and the improved FEC on the three datasets. Overall both algorithms perform very well for all training and testing scenarios. FEC achieves poor performances for the Spain dataset. On the other hand, the neural network, while achieving good performances across all datasets (MaP$>$ $0.80$), improves by a significant margin

| | FEC | NN | FEC | NN | FEC | NN |
|---|---|---|---|---|---|---|
| Trained / Tested | Spain | | UK1 | | UK2 | |
| Spain | 0.73 | **0.94** | 0.64 | **0.81** | 0.67 | **0.85** |
| UK1 | **0.90** | 0.85 | 0.94 | **0.95** | 0.92 | **0.94** |
| UK2 | **0.92** | 0.85 | 0.92 | **0.92** | 0.94 | **0.94** |
| Mean | 0.85 | **0.88** | 0.83 | **0.89** | 0.84 | **0.91** |

Table 3. Comparison of the Mean Intersection over Union (MIoU) of the segmentation masks

($\sim$ $22\%$ on average) the results on the Spain dataset and seems to achieve better generalisation. Lower results from the neural network overall are explained by missing or partially detected broccoli heads on the upper and lower side of the point cloud. With a lower IoU, they impact negatively on the precision-recall metrics, being classified as false positives and false negatives. When trained on the Spain set and tested on the UK set, the lower score obtained by the neural network can be attributed to the extraction of small areas on leaves similar in shape to the smaller Spain broccoli heads, as seen in Figure 8.

The FEC clustering algorithm extracts more segments that are then found in the detection predictions later on, as seen in Figure 10. However they rarely have a probability to be classified positive higher than 75%. The lower performances from FEC on the Spain dataset but high when tested on the UK sets, are due to the segment extraction strategies of the algorithm, which struggles more on smaller and more cluttered and occluded objects (Spain set), but performs very well on big and separated objects (UK sets). We illustrate this in Figure 8.

Also with a high precision and recall, the neural network tends to be very selective in the choices for detection. This results in good masks for the detected instances with high probabilities, but it also gets rid of small detection in the Spain set when trained on the UK one, as small objects are rarely seen in the UK sets. We show in Figure 5 the precision-recall curves for the neural network and FEC algorithms when trained on the Spain set and tested on the same set. As one can see, the neural network performs very well for thresholds up to $85\%$, but achieves lower results after $95\%$. FEC starts to struggle earlier around $70\%$ of IoU and fails after $85\%$.

### 5.2. Semantic segmentation

To study the effectiveness of the segmentation of each approach, we study them for the semantic segmentation task. We present in Table 3 the results obtained by both algorithms for the MIoU metric. The neural network for this task shows significant improvements or similar results compared to FEC. This offers several advantages such as more accurate localisation, better shape estimation for phe-

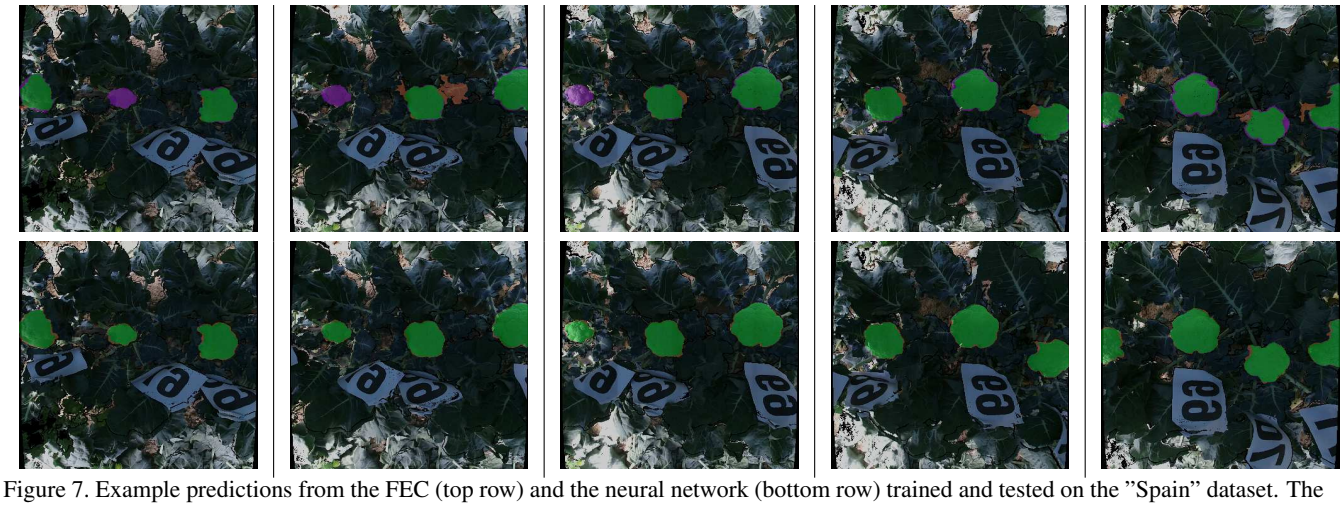

Figure 7. Example predictions from the FEC (top row) and the neural network (bottom row) trained and tested on the "Spain" dataset. The colour overlay labels correspond to true positives (green), false positive (orange) and false negatives (purple).

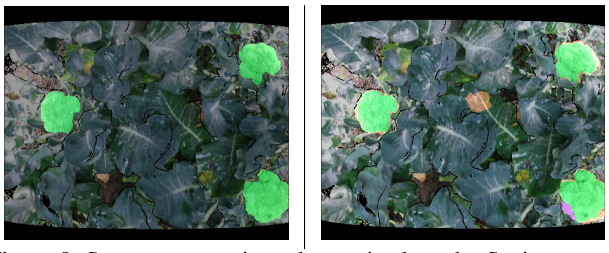

Figure 8. Segment extraction when trained on the Spain set and testing on UK set for FEC (left) and neural network (right).

notyping, and grasp prediction for harvesting. Furthermore the results are consistent with Table 2, but we see some increase for the methods tested on the Spain set. This is due to the false positive detection now being evaluated for their mask, and impacted less than as a whole object (for a point cloud with 4 broccoli heads, 10 false positives impact the score more than 400 points among 200k points). Figs. 7 and 10 illustrate the differences in terms of mask and point classification, and how the neural network performs better. In Table 2 and Table 3, NN show similar differences in performances with FEC. While performing better or similarly well on most training/testing scenarios, NN struggles to generalise when trained on Spain and tested on UK. This comes from the broccoli heads size, creating more False Positives on UK where the scale differs and Spain-like broccoli shapes can be found on leaves and foliage (Figure 8).

### 5.3. Qualitative analysis

We show in Figure 6 different feature maps decoded by our network. They are obtained by passing the points and normals through the network and visualising one of the dimensions of the feature map at the end of one of the decoder de-convolution blocks. One can distinguish the contour features characteristic of the broccoli in the first feature map, which can be related to Figure 9, allowing a better extrac-

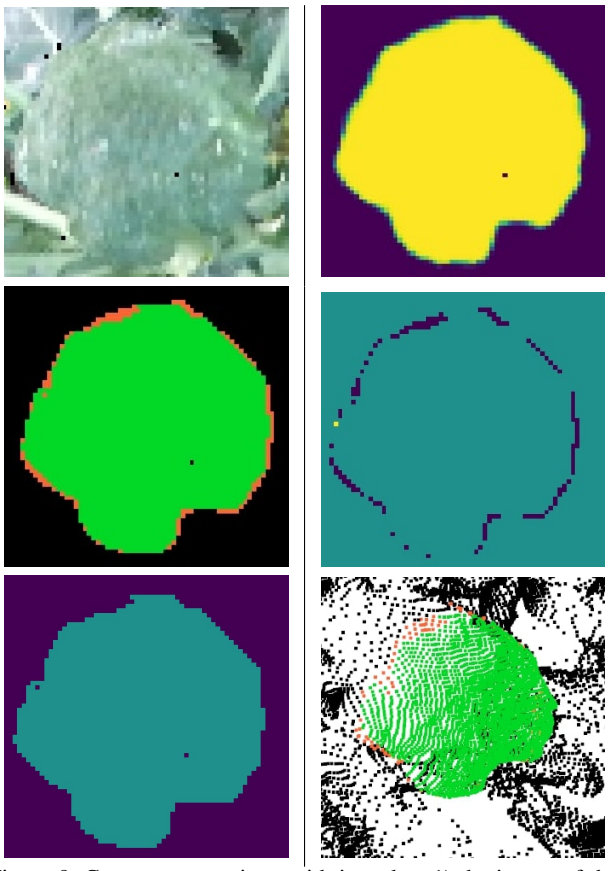

Figure 9. Contour uncertainty, with in order: 1) the image of the broccoli, 2) the predicted segmentation mask, 3) the true positive (green) and false positive (orange) map, 4) the difference with ground truth, 5) the ground truth annotation, and finally, 6) the visualisation in 3D space of the prediction.

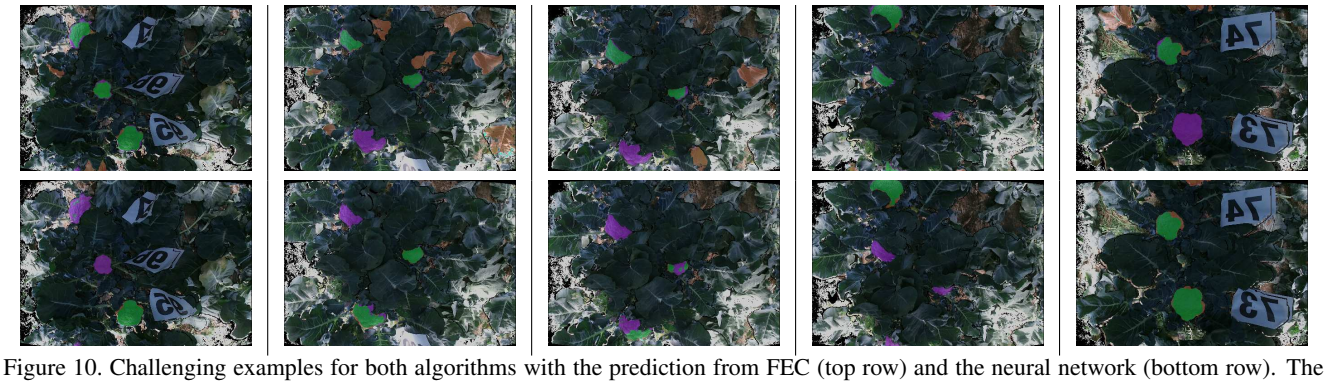

Figure 10. Challenging examples for both algorithms with the prediction from FEC (top row) and the neural network (bottom row). The colour overlay labels correspond to true positives (green), false positive (orange) and false negatives (purple).

tion of the object by the network. In the second and third feature maps, features related to the shape of the broccoli are extracted, with an emphasis on the texture for the second one and on the normal features for the third one. The fourth feature map let us see the leaf edges extracted from the point cloud. The last feature map shows features which seem to be more related to distances along the $z$-axis, with an emphasis on vegetation above the ground.

We decide to a qualitative analysis of some Spain examples, as they offer a bigger challenge than the UK set, and show the limitation of the baseline work and least performing results of our NN. Figure 7 presents some qualitative results from our network on the different datasets. Both algorithms were trained on the Spain set and tested on the same set. As one can see, the neural network offers more reliable results for the localisation and detection of the broccoli heads. On the other hand, FEC struggles to find small instances and tends to cluster the broccoli heads with surrounding leaves, reducing the precision of its segmentation. Also, for bigger objects FEC tends to generate more false negatives on the contours of the objects, while the neural network tends to add a few more points to the mask.

We present in Figure 9 what the uncertainty in our network represents for our detection. Using the softmax function, we can represent our prediction for each class (background and broccoli) using probabilities. With very low probability noisy segmentation removed using a 0.5 threshold, some uncertain areas remain surrounding the objects. In Figure 9 we see for a broccoli example the lower probability surrounding the shape. These low probability contours are most of the time true positive or false positive, but are easy to get rid of using a higher threshold. However, even though not present in the annotation, they give us some information about the surrounding of the broccoli and its width for further analysis. We also show their representation from a 3D perspective, where we can see that such contours can be used to separate the broccoli effectively from the more noisy ground points.

In Figure 10 we focus on challenging cases when training our algorithms on the first UK set and testing it on the challenging Spain set. In this case the neural network mostly fails to detect some of the broccoli heads with particular shapes and strong occlusions not present in the UK set. On the other hand, FEC detects more of these challenging broccoli heads, but is not very discriminating and also detects false positives on the leaves and other similar areas.

For the complete processing of a single point cloud consisting of 217k points, our method's inference time is $\sim 0.02$ s compared to $\sim 6$ s for FEC.

## 6. Conclusion

We presented a new method for processing 3D information acquired through RGB-D cameras in the context of robotic vision for agriculture. We chose the broccoli harvesting applications to evaluate our method and its implications, due to the availability of high quality data and a state-of-the-art algorithm for processing it. Our method achieves similar results on the baseline datasets and better results for the challenging sets, while providing better segmentation of the objects, competitive instance segmentation, better localisation, and faster inference by a factor of 300. All these new aspects make it better suited for real-time applications in selective harvesting, and open new possible applications in online phenotyping, crop analysis and yield prediction.
However our method faces some challenges intrinsic to the data. Difference in size between objects and datasets leads to missed detection, especially on the upper and lower boundaries where the distortion varies the most. Training on intrinsically different object size than the test set (Spain training to UK testing) also affect the results, yielding more False Positives (Figure 8).

Future work will involve solving these challenges through simple processes and experiments such as data normalisation, un-distortion, data augmentation, etc. We also plan to investigate the use of more advanced architecture such as ResNet[4], to improve robustness to scale and shape variation, occlusion and data diversity.

## References


[1] Asher Bender, Brett Whelan, and Salah Sukkarieh. A high-resolution, multimodal data set for agricultural robotics: A Ladybird's-eye view of Brassica. *Journal of Field Robotics*, (Early View), 2019.

[2] Pieter M. Blok, Ruud Barth, and Wim 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.

[3] Esmael Hamuda, Martin Glavin, and Edward Jones. A survey of image processing techniques for plant extraction and segmentation in the field. *Computers and Electronics in Agriculture*, 125:184–199, 2016.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] S. Holzer, R. B. Rusu, M. Dixon, S. Gedikli, and N. Navab. Adaptive neighborhood selection for real-time surface normal estimation from organized point cloud data using integral images. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2684–2689, 2012.

[6] Stefan Holzer, Radu Bogdan Rusu, Michael Dixon, Suat Gedikli, and Nassir Navab. Adaptive neighborhood selection for real-time surface normal estimation from organized point cloud data using integral images. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2684–2689. IEEE, 2012.

[7] Mingyang Jiang, Yiran Wu, Tianqi Zhao, Zelin Zhao, and Cewu Lu. PointSIFT: A SIFT-like Network Module for 3D Point Cloud Semantic Segmentation. *arXiv e-prints*, Jul 2018.

[8] Andreas Kamilaris and Francesc X. Prenafeta-Boldú. Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147:70–90, 2018.

[9] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv e-prints*, page arXiv:1412.6980, Dec. 2014.

[10] Keerthy Kusumam, Tomáš Krajník, Simon Pearson, Tom Duckett, and Grzegorz Cielniak. 3D-vision based detection, localization, and sizing of broccoli heads in the field. *Journal of Field Robotics*, 34(8):1505–1518, 2017.

[11] Bo Li. 3D Fully Convolutional Network for Vehicle Detection in Point Cloud. *arXiv e-prints*, page arXiv:1611.08069, Nov. 2016.

[12] Yangyan Li, Rui Bu, Mingchao Sun, and Baoquan Chen. PointCNN: Convolution On $X$-Transformed Points. *arXiv preprint arXiv:1801.07791*, 2018.

[13] T Lin, Michael Maire, Serge J Belongie, Lubomir D Bourdev, Ross B Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. arxiv 2014. *arXiv preprint arXiv:1405.0312*.

[14] Justin Le Louedec., Bo Li., and Grzegorz Cielniak. Evaluation of 3d vision systems for detection of small objects in agricultural environments. In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP,*, pages 682–689. INSTICC, SciTePress, 2020.

[15] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, pages 5099–5108, 2017.

[16] Rachael Angela Ramirez. Computer vision based analysis of broccoli for application in a selective autonomous harvester. mathesis, Virginia Polytechnic Institute and State University, 2006.

[17] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[18] Radu Bogdan Rusu, Gary Bradski, Romain Thibaux, and John Hsu. Fast 3d recognition and pose using the viewpoint feature histogram. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2155–2162. IEEE, 2010.

[19] Andy Zeng, Kuan-Ting Yu, Shuran Song, Daniel Suo, Jr. Walker, Ed, Alberto Rodriguez, and Jianxiong Xiao. Multi-view Self-supervised Deep Learning for 6D Pose Estimation in the Amazon Picking Challenge. *arXiv e-prints*, page arXiv:1609.09475, Sept. 2016.

[20] Yuanshen Zhao, Liang Gong, Yixiang Huang, and Chengliang Liu. A review of key techniques of vision-based control for harvesting robot. *Computers and Electronics in Agriculture*, 127:311–323, 2016.