Received: 22 December 2016 | Revised: 19 March 2017 | Accepted: 18 April 2017

DOI: 10.1002/rob.21726

FIELD REPORT

WILEY

3D-vision based detection, localization, and sizing of broccoli heads in the field

Keerthy Kusumam[1] | **Tomáš Krajník**[2] | **Simon Pearson**[3] | **Tom Duckett**[4] | **Grzegorz Cielniak**[4]

[1]Computer Vision Laboratory, University of Nottingham, UK

[2]Artificial Intelligence Center, FEE, Czech Technical University, Czechia

[3]Lincoln Institute for Agri-food Technology, University of Lincoln, UK

[4]Lincoln Centre for Autonomous Systems, University of Lincoln, UK

Correspondence
Grzegorz Cielniak, Lincoln Centre for Autonomous Systems, University of Lincoln, UK
Brayford Pool
LN6 7TS Lincoln
Email: gcielniak@lincoln.ac.uk

Abstract

This paper describes a 3D vision system for robotic harvesting of broccoli using low-cost RGB-D sensors, which was developed and evaluated using sensory data collected under real-world field conditions in both the UK and Spain. The presented method addresses the tasks of detecting mature broccoli heads in the field and providing their 3D locations relative to the vehicle. The paper evaluates different 3D features, machine learning, and temporal filtering methods for detection of broccoli heads. Our experiments show that a combination of Viewpoint Feature Histograms, Support Vector Machine classifier, and a temporal filter to track the detected heads results in a system that detects broccoli heads with high precision. We also show that the temporal filtering can be used to generate a 3D map of the broccoli head positions in the field. Additionally, we present methods for automatically estimating the size of the broccoli heads, to determine when a head is ready for harvest. All of the methods were evaluated using ground-truth data from both the UK and Spain, which we also make available to the research community for subsequent algorithm development and result comparison. Cross-validation of the system trained on the UK dataset on the Spanish dataset, and vice versa, indicated good generalization capabilities of the system, confirming the strong potential of low-cost 3D imaging for commercial broccoli harvesting.

KEYWORDS

3D vision, agricultural robotics, semantic mapping

1 | INTRODUCTION

Sustainable intensification of agriculture can be achieved through various technological innovations such as automated harvesting. Automated harvesting approaches bring benefits of reduced labor costs, economic sustainability, increased productivity, less waste, and better use of natural resources. Selective harvesting methods choose only mature crops for harvesting, as compared to "slaughter harvesting" where an entire field is harvested in a single pass. Broccoli is an instance of the crops that require selective harvesting since the flowers exhibit a high variation in maturity levels, even when grown in the same field. To address these challenges, an automated selective harvesting robot would need an intelligent vision system that can detect and locate the harvestable broccoli heads, as well as measuring the head size to check conformance with the required standard, which is typically determined by the food retailers. However, such systems encounter a number of difficulties arising from the natural variations, partial views of the heads and occlusions due to leaves and weeds.

The main objective of the work presented was to investigate the feasibility of using low-cost consumer 3D cameras mounted on a moving tractor to identify mature broccoli heads in real, outdoor field conditions, providing both the locations and size estimates of the detected heads. The approach presented applies state-of-the-art methods for 3D feature extraction, classification, temporal filtering to remove false positives and track the detected heads, and estimation of the head size. To evaluate the approach, sensory datasets were recorded in two different countries with complementary growing seasons; the UK, where broccoli is a summer crop, and Spain, where broccoli is a winter crop. Considerable effort was placed into annotating these dataset, including hand annotation of the ground-truth positions for broccoli heads on both the UK and Spain datasets, and independent (destructive) measurements of the head sizes for a selected set of broccoli plants in the Spain dataset. We also are making these datasets publicly available to enable result comparison and further development of vision algorithms for automated harvesting by the research community.

The experiments conducted showed that a combination of Viewpoint Feature Histogram (VFH) features and a Support Vector Machine (SVM) classifier enables detection of broccoli heads with high precision. Moreover, we demonstrated that the integration of detection results across multiple frames using temporal filtering enables pruning of false positive detections, further improving the precision to around 95% for the UK dataset and 85% for the Spain dataset. The results indicate that the Spain dataset is more difficult than the UK one, due especially to occlusions of the broccoli heads by leaves. However, cross-validation of the system trained on the UK dataset to the Spain dataset, and vice versa, indicate excellent generalization capabilities of the system to work under different field conditions. We also demonstrated that the temporal filtering can be used to generate a 3D map of the broccoli head positions in the field, which would enable the future development of a robotic solution for broccoli harvesting.

In summary, this article documents the following research contributions:

- a 3D point-cloud-based classification pipeline for detection of mature broccoli plants under real-world field conditions;
- algorithms for localization and mapping of the detected heads in 3D image coordinates, without requiring GPS coverage or other external localization system;
- algorithms for automatically measuring the size of the detected heads, to determine when a head is ready for harvest;
- a collection of real-world datasets comprising annotated 3D point clouds and color images recorded in both UK and Spain, with ground-truth data for the broccoli head locations and sizes, which we also make publicly available for use by other researchers;
- a comprehensive evaluation of the system performance, including detection, tracking and head size estimation, based on the above datasets.

The rest of this article is structured as follows. Section 2 gives an analysis of related work. Section 3 describes the hardware platform for data acquisition, while Section 4 describes the algorithms developed as part of the vision pipeline. Section 5 describes the acquired datasets, followed by experimental results in Section 6 and conclusions in Section 7.

2 | RELATED WORK

Several approaches can be identified in precision agriculture for detection, recognition, localization and harvesting of different crop varieties. Analysis of 2D images acquired from high resolution, industrial grade cameras such as CCD is one of the most prominent approaches as shown in McCarthy et al.[1] and Jimenez et al.[2] Several methods use color analysis such as the strawberry picking robot in Hayashi et al.[3] and apple harvesting using color and shape features in Ji et al.[4] Okamoto and Li[5] developed a citrus harvesting robot by using template matching to find circular objects on an edge image. Circular Hough transform voting was used to detect apples in Wachs et al.[6] Haug et al.[7] classify carrot crops versus weeds using geometric features combined with a random forest classifier. Other sensors were also investigated such as NIR spectral imaging for harvesting capsicum in a cluttered environment using texture features.[8]

Using only image analysis may be insufficient for reliable estimation of the crop locations for machine vision systems in outdoor field conditions. Hence there has been an increase in the use of 3D sensors for depth perception.[9] The advantages of 3D include encoding the object geometry and providing different viewpoints under clutter. Barnea et al.[10] used combined information from RGB and depth for detecting sweet peppers, by detecting highlights in the image planes on registered RGB-D images to identify fruit regions and classifying peppers based on surface normal distribution and 3D object symmetry. Sa et al.[11] use an RGB-D sensor to detect sweet pepper peduncles to facilitate robotic harvesting. The cucumber harvesting robot by van Henten et al.[12] employs high resolution CCD cameras for detection of crops in greenhouses and 3D data for localization. Weiss and Biber[13] use 3D LIDAR for maize row detection and mapping using 3D geometry features. A similar sensing approach was used by Nakarmi and Tang[14] for in-row maize plant spacing measurements. Gai et al.[15] proposed to discriminate crops from weeds using a Kinect 2 sensor and a combination of 2D and 3D morphological features. Li[16] considered 3D vision for weed discrimination and plant phenotyping including broccoli and soybean plants. Nguyen et al.[17] employ an RGB-D sensor to detect apples, by using 3D data processing to segment out clusters in a point cloud and applying a circular Hough transform to the 2D transformed image to detect apples. The method uses color filtering which limits the kind of apples that can be detected.

Several approaches use RGB images to identify broccoli heads. Ramirez[18] explored a number of standard image processing techniques including texture to identify broccoli heads and showed that the approach has promise for in field selection, but the study was limited to a very small sample size (13 images). Tu et al.[19] showed that image analysis techniques and neural networks could be applied to identify broccoli quality parameters, but the approach was limited to heads imaged on a white light stable background, isolated from the leaves.

Our approach extends the state of the art by detecting and localizing broccoli heads with low-cost 3D sensors in real, unstructured outdoor field conditions (see Figure 1 for the overview of the entire system).

3 | HARDWARE PLATFORM

One of the main requirements of the data collection is reliable RGB-D data capture in outdoor field conditions under different weather conditions such as sunny or overcast. In earlier work we found that an Asus Xtion Pro sensor performed poorly under ambient sunlight due to interference with its infra-red sensing for measuring depth. So in this work we evaluated the Kinect 2, a state-of-the-art low-cost RGB-D sensor based on time-of-flight technology.[20] The Kinect 2 provides high resolution RGB images at 1920×1080 pixels along with a depth resolution of 512×424 and a field of view of $71° \times 60°$ resulting in an average of 7×7 pixels per degree. The experimentally verified depth accuracy error of Kinect 2 reported by Yang et al.[21] is

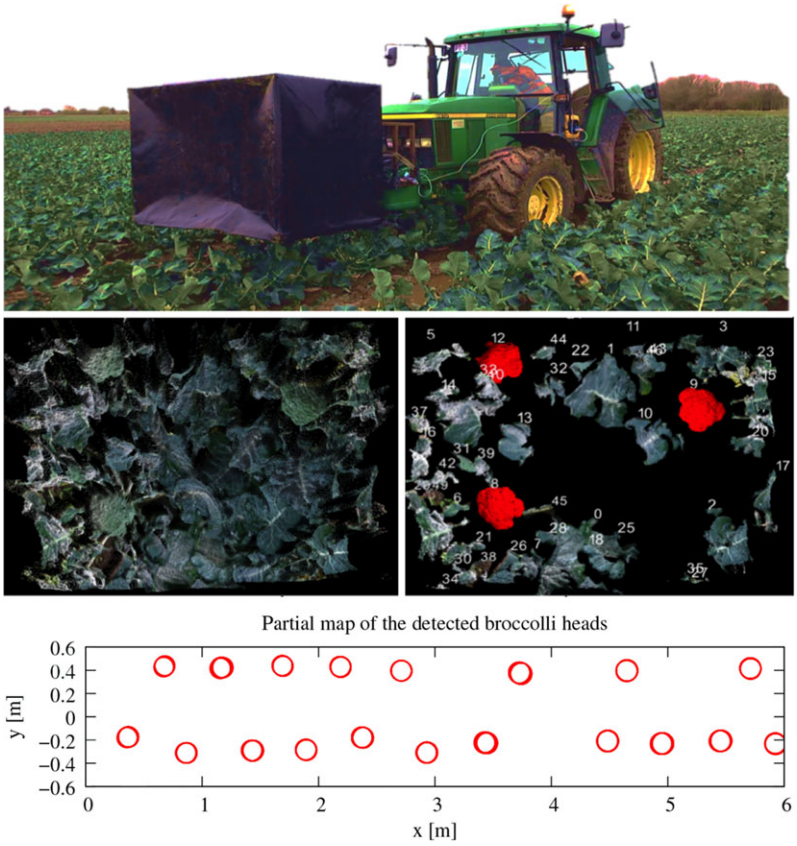


FIGURE 1 System overview. Top: Tractor equipped with 3D sensors used for field data collection. Middle: RGB-D images of broccoli plants (left) are analyzed for identifying head locations using 3D recognition algorithms (right). Bottom: Temporal filtering then combines detections from multiple frames to localize individual broccoli heads

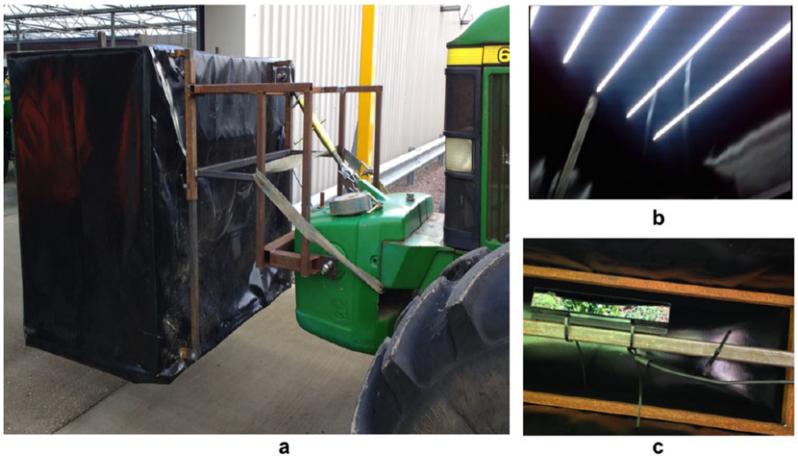


FIGURE 2 Enclosure and lighting set-up. (a) Manually adjustable shroud attached to the front of the tractor (b) artificial lighting using LED strip lights mounted inside the shroud to the top of the canopy (c) inside view of the enclosure with Kinect 2 mounted perpendicular to the field

< 2 mm in a central region of the sensor. The sensor was fixed inside a specially constructed enclosure, which acts as a "shroud" to block direct sunlight incident at the sensor to alleviate noise and also as an "umbrella" during rainy conditions. The enclosure was equipped with an artificial lighting source, comprising strip LED lighting, to help regularize the color images from the sensor, and to enable data capture during both day-time and night-time conditions. The sensor was mounted upright and at different heights from the ground (125–140 cm in our experiments), see Figure 2. We used the auto-calibration of color and depth images using the factory defaults provided by the Kinect 2, and the sensor data were recorded with a standard laptop.

For the UK-based experiments, the data was collected with the camera enclosure mounted on the front of a tractor, as shown in Figure 2 . To facilitate easy re-assembly of the hardware set-up on different tractor types, it was further decided to modify the set-up so that the camera enclosure could be mounted at the rear of a tractor using a standard three-point linkage. Thus, the set-up shown in Figure 3 was used for data capture in Spain.

FIGURE 3 Set-up for data collection in Spain with the hardware platform mounted at the rear of the tractor

4 | VISION SYSTEM

In 3D object recognition methods, there are two common approaches: local and global object recognition pipelines. The former involves locating a number of keypoints in the point cloud and extracting a set of features in the neighborhood of these points. These points are matched to a model and the correspondences are grouped according to the geometry of the model. In a global recognition pipeline, the scene is segmented into segments or regions and features are extracted for each segment. These features are matched to the global descriptors of the model.

This paper applies a global recognition method for detection of broccoli heads in the scene. The vision system processes the depth data and the detection pipeline comprises the following main steps: (i) 3D point cloud pre-processing, (ii) feature extraction, (iii) classification, and (iv) temporal filtering, as shown in Figure 4. Additionally, we evaluate two alternative algorithms for size estimation of the tracked broccoli heads.

4.1 | 3D point cloud pre-processing

We pre-process the raw point cloud data captured by the sensor to remove outliers, segment out the ground plane and group the remaining point cloud segments into clusters. We use the algorithms available as part of the PCL C++ library[22] for processing point clouds.

4.1.1 | Outlier removal

The input point cloud data may contain outliers resulting from sensor measurement inaccuracies which can be considered as noise. It is important to remove these noisy data as they may lead to errors in subsequent processing. We use a statistical outlier removal algorithm that analyzes the distribution of the distances between neighboring points. For each point in the input cloud the algorithm computes the distances to its k neighbors and finds the mean and standard deviation of these distances ($k = 100$ in our experiments to minimize noise in segmentation results). It removes all those points that fall outside a certain distance threshold defined by the sum of the global mean and standard deviation. The value of this parameter affects the computational performance of the method and therefore generally lower values are preferable.

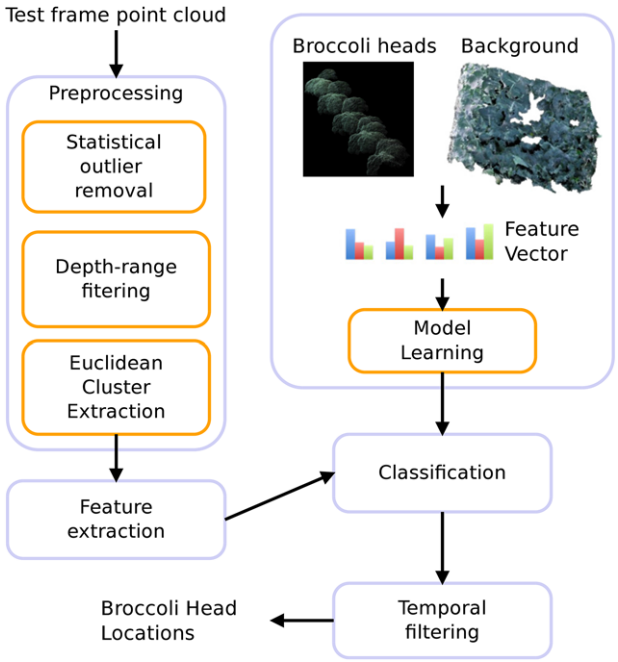


FIGURE 4 3D vision system pipeline. The frames of 3D point cloud data are first processed by pre-processing routines for outlier removal, depth filtering and cluster extraction. Then features are extracted and analyzed using a learned model to predict the target class. The returned detections are used by temporal filtering to remove false positives

4.1.2 | Ground filtering

The segmentation of the ground is achieved through thresholding the depth range, i.e., the z dimension of the input point cloud. The depth data are filtered at a user defined range of 0.5–1m and the points that lie outside the range are discarded. We defined these parameters based on the distance of the sensor to the ground measured during data collection.

4.1.3 | Cluster segmentation

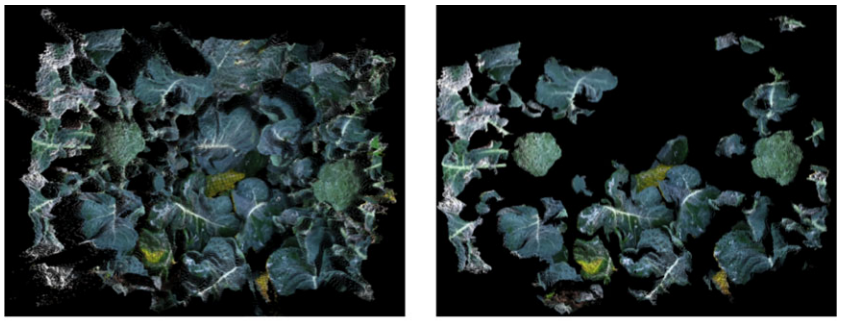
The next step is to group the remaining point cloud segments into different clusters for segmentation. We use a distance-based clustering algorithm to cluster points based on the Euclidean distance between point pairs. The algorithm chooses each point and considers its neighbors defined by a certain radius. It greedily adds new points to the current cluster if the distance of the neighboring points are within a user-defined cluster tolerance.

An important parameter to this algorithm is the cluster tolerance, which is set to 5 mm. Smaller values would result in over-segmentation and higher values would result in merged clusters. In this case, we search for prominent foreground objects, and set the minimum and maximum size of the returned clusters as 500 and 10,000, respectively. Clusters outside that range are discarded.

Figure 5 shows example results of the pre-processing steps.

4.2 | Feature extraction

We use a set of global 3D feature descriptors that characterize the geometry of an object as a whole. The features are extracted for



(a) Point cloud after depth filtering. (b) Point cloud after cluster extraction.

FIGURE 5 Pre-processing steps performed on the 3D data

each clustered segment $\mathcal{P}$ in the input point cloud derived after pre-processing. A good feature descriptor should be discriminative with respect to the two given classes, i.e., broccoli heads and non-broccoli segments representing leaves, ground or weeds. We describe the different 3D feature extraction algorithms used in the pipeline as follows.

4.2.1 | Viewpoint angle histogram

The distribution of the surface normal directions should encode the underlying geometry of the broccoli heads and hence be discriminative compared to that of the leaves or other background clusters. To calculate a Viewpoint Angle (VA) feature, first for each point $\mathbf{p}_i \in \mathcal{P}$ a corresponding normal $\hat{\mathbf{n}}_i$ is calculated. Then the centroid $\mathbf{p}_c$ of the input cluster $\mathcal{P}$ is calculated together with a normalized vector $\hat{\mathbf{v}}_c$ between the viewpoint (i.e., the sensor position) and the centroid. The viewpoint vector is translated to each point location before calculating the angle to achieve scale invariance. The angle measuring the difference between each normal $\hat{\mathbf{n}}_i$ and a viewpoint $\hat{\mathbf{v}}_c$ can be derived as $\gamma_i = \arccos(\hat{\mathbf{n}}_i \cdot \hat{\mathbf{v}}_c)$. Finally, these angles are then binned into a histogram of the range $0 - \pi$ radians and normalized. We use 12 orientation bins in our experiments.

4.2.2 | Viewpoint feature histogram

The Viewpoint Feature Histogram (VFH) descriptor extends the VA features by incorporating also the Fast Point Feature Histogram (FPFH) descriptor introduced by Rusu et al.[23] The extended FPFH component is computed by calculating the roll α, pitch ϕ and yaw θ angles between each point $\mathbf{p}_i$ and each $\mathbf{p}_j$ point from its k-nearest neighbors as

$$\alpha = \mathbf{v} \cdot \hat{\mathbf{n}}_j, \tag{1}$$

$$\phi = \mathbf{u} \cdot \frac{\mathbf{p}_i - \mathbf{p}_j}{d}, \tag{2}$$

$$\theta = \arctan\left(\mathbf{w} \cdot \hat{\mathbf{n}}_j, \mathbf{u} \cdot \hat{\mathbf{n}}_j\right), \tag{3}$$

where $d = \|\mathbf{p}_i - \mathbf{p}_j\|$ and $\mathbf{u}, \mathbf{v}, \mathbf{w}$ represent so-called Darboux coordinates defined as $\mathbf{u} = \hat{\mathbf{n}}_i, \mathbf{v} = (\mathbf{p}_j - \mathbf{p}_i) \times \mathbf{u}, \mathbf{w} = \mathbf{u} \times \mathbf{v}$. Finally, the tuple $\langle \alpha, \phi, \theta, d \rangle$ is discretized into 4 histograms. We use 128 bins for the VAF part and 45 bins for each FPFH component resulting in a feature vector of 308 values in total.

4.2.3 | Clustered viewpoint feature histogram

Clustered Viewpoint Feature Histogram features[24] are an extension to the VFH features for robustness against occlusions and partial views. The features are computed by first dividing the cluster into multiple smooth and stable regions using a region-growing segmentation algorithm, which clusters points of similar normals. The algorithm also removes points with high curvature, which are typically associated with noise, object edges, etc. The VFH features are then calculated for each of the smooth regions forming the final descriptor. This property allows the recognition of the clusters given that only partial views are available. We use the same histogram bin numbers as for the VFH features in our experiments and therefore the resulting feature vector length is also 308.

4.2.4 | Geometric features

The geometric features consist of a set of measures that define different geometric properties of objects. We use a subset of measures defined in Ref. 25 which characterize the underlying geometry of a clustered segment $\mathcal{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$ by using the morphological attributes defined as follows:

i *Area*: defined by the total number of points in a cluster $f_a = n$.

ii *Compactness*: estimates the compactness of the cluster by computing the ratio of the total surface area of the cluster to the surface area of the smallest binding sphere with a radius r. We approximate this measure by using the following formula $f_{cp} = n/r^2$. The more spherical the cluster looks, the higher would be its compactness value. We expect broccoli clusters to have higher compactness values than clusters originating from leaves and background.

iii *Smoothness*: measures if the neighborhood around a point $\mathbf{p}_i$ is spread uniformly by projecting the neighborhood points to the tangent plane defined by the surface normal at the point $\hat{\mathbf{n}}_i$. The calculated angles of the projected points in the local 2D coordinate system are then binned into a histogram from which an entropy metric is calculated. Finally, the mean entropy for all points is used as an estimate of the smoothness value f_{sm} where high entropy implies high smoothness. Lower values for this metric should correspond to flat and thin clusters such as broccoli leaves.

iv *Local convexity*: measures if a cluster consists of locally convex regions, i.e., those that satisfy the following condition:

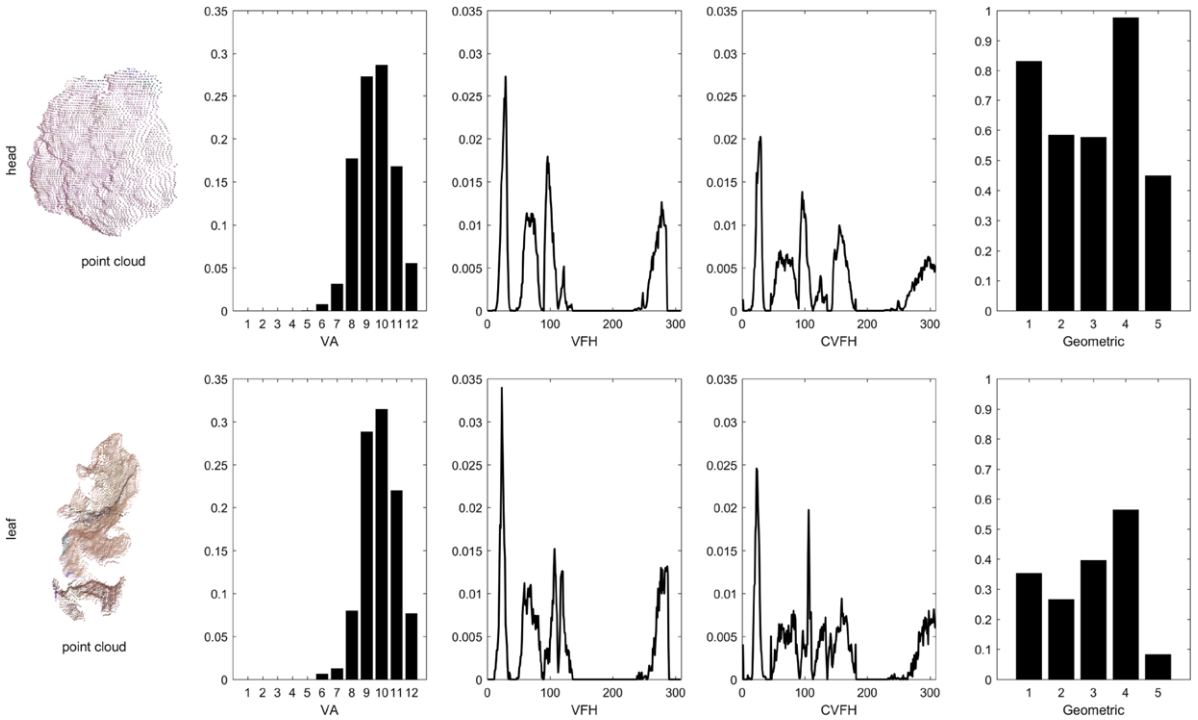


FIGURE 6 Example feature set extracted for a broccoli head (top) and a leaf (bottom)

$(\mathbf{p}_j - \mathbf{p}_i) \cdot \hat{\mathbf{n}}_j > 0$. Each cluster is ranked by the percentage of detected convex edges which we denote as f_{cn}. Lower values should correspond to clusters with abrupt changes which should not be present in broccoli clusters.

- *Symmetry*: computes the score for reflective symmetry of a cluster $\mathcal{P}$ through three principal axes:

$$f_{sy} = \sum_{d \in \{x,y,z\}} \frac{\lambda_d}{\lambda_x + \lambda_y + \lambda_z} \left(\mathcal{O}\left(\mathcal{P}, \mathcal{P}_{-d}, r_d\right) + \mathcal{O}\left(\mathcal{P}_{-d}, \mathcal{P}, r_d\right) \right), \quad (4)$$

where λ is an eigenvalue, $\mathcal{P}_{-d}$ is a reflection of a cluster along the d axis and $\mathcal{O}$ is an overlap measure. We expect broccoli clusters to be more symmetrical than leaves/background.

Finally, we concatenate all geometrical features into one feature vector $\mathbf{f}_g = \{f_a, f_{cp}, f_{sm}, f_{cn}, f_{sy}\}$. Figure 6 depicts the described features for an example of broccoli head and a leaf.

4.3 | Model learning and classification

We employ machine learning to learn the appearance of the broccoli heads using one or more of the described features. We use models learnt by the learning algorithms (i.e., classifiers) to distinguish between broccoli heads and background leaves or ground. For training the classifiers, a set of training data is used consisting of N input feature vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$, where $\mathbf{x}_i \in \Re^n$, along with the corresponding class labels $t_n \in \{-1, +1\}$. The aim of the recognition algorithm is to classify each feature vector $\mathbf{x}$ as one of the target classes t_1 (i.e., broccoli) or t_2 (i.e., background/leaves).

K-Nearest Neighbors (KNN) is a popular instance-based classification algorithm where the training phase involves storing all the training feature vectors along with the class labels. Given a new feature instance $\mathbf{x}_i$, the nearest neighbor algorithm searches for the k nearest neighbors to the query point in the training set. A distance metric is used to rank the closest neighbors—we use Euclidean distance (L2 measure) in our implementation. The class that represents the majority of the k neighbors is assigned as the predicted class of the query instance. We use this basic classifier as a reference point to the state-of-the-art method based on support vectors.

The Support Vector Machine (SVM) is a binary classification algorithm. The SVM is shown to be efficient even in cases where the data is not linearly separable. It can also be used to classify data in higher dimensions using kernels. The linear discriminant function that separates the two classes is given by $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$, where $\mathbf{w}$ is the weight vector, $\mathbf{x}$ is the feature vector and b is the bias. If the training data are linearly separable, then the sign of the function determines the target class assigned to the data points, i.e., $t_n y(\mathbf{x}_n) > 0$ holds true for all correctly classified instances. The parameter C controls the trade-off between training errors and generalization or complexity of the classifier and this parameter can be tuned for the given data using cross-validation.

The output of the classification step is a set of clusters representing the broccoli heads as shown in Figure 7, along with the x, y, z positions of the centroid locations of each of the clusters.

4.4 | Head size estimation

The final task of the proposed vision system is to estimate the size of the detected broccoli heads in order to determine where they are ready for harvest with respect to the market specifications. We propose two different methods to estimate the head size based on (i) bounding box, (ii) convex hull. The first method estimates the

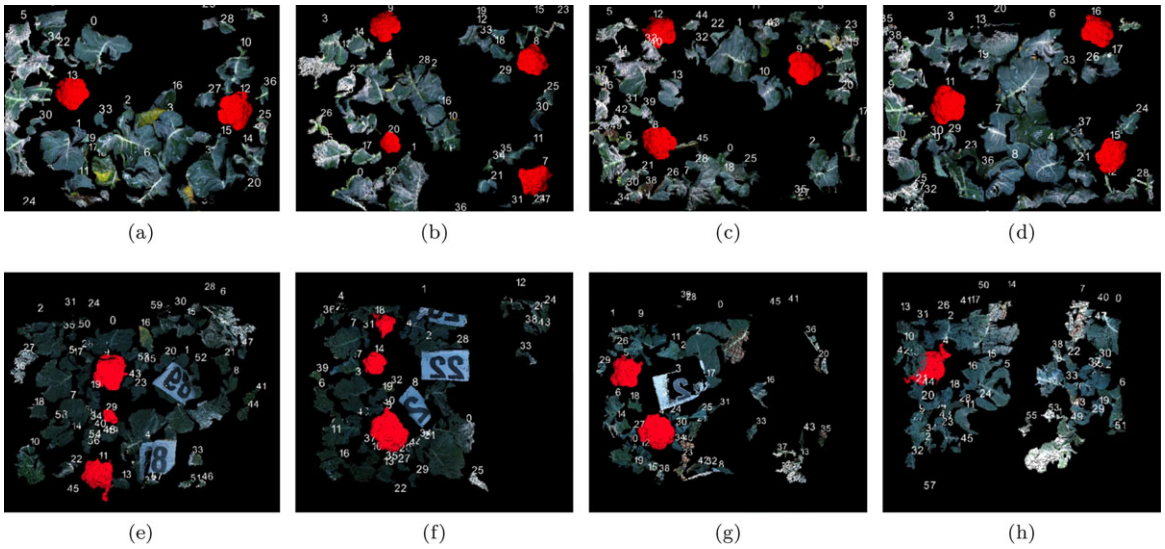


FIGURE 7 The output of the detection algorithm on selected frames from (a)–(d) the UK dataset, (e)–(h) the Spanish dataset. Numbers indicate the point cloud segments after cluster extraction, ordered by the size. The segments marked in red are the detections retrieved from the vision system using VFH features and SVM classifier

diameter of the head by measuring the limits of the point cloud. In our implementation, we only consider the x axis thus measuring the width of the head. This has proven to be more robust than relying on all three dimensions. The proposed method is computationally attractive as it only requires a single iteration through all cluster points. The latter method first computes the bounding convex hull (we use the 3D convex hull method included in the PCL library[26]) and its centroid. Then we measure distances between all points and the calculated centroid. The mean value of these distances represents the radius of the broccoli head. We evaluate the accuracy and precision of both methods in the evaluation section.

4.5 | Temporal filtering and 3D mapping

Since the RGB-D sensor provides 15 frames per second and the harvester speed is approximately 0.3 m/s, the x, y, z positions of the individual broccoli heads in consecutive frames differ only by a few centimeters. This allows to track the locations of the broccoli heads over several frames as they pass through the sensed area.

The tracking algorithm maintains the position x_c and velocity v_c of the RGB-D camera and three sets of broccoli heads: a set of tracked broccoli heads $\mathcal{T}$, a set of currently detected heads $\mathcal{D}$ and a set of mapped broccoli $\mathcal{M}$. Each time a picture is processed, the set $\mathcal{D}$ is populated with the detected broccoli, where each $\mathbf{d_i} \in \mathcal{D}$ contains the broccoli centroid coordinates x, y, z in the camera coordinate system, its tracking score n and head size s. First, we transform the coordinates of each broccoli $\mathbf{d_i}$ in the set $\mathcal{D}$ to the global coordinate system simply by translating them by the camera coordinates x_c – rotation is unnecessary, as the tractor moves along a straight row. Then, for each broccoli $\mathbf{d_i}$ in the set $\mathcal{D}$, we find the closest (in terms of Euclidean distance) broccoli $\mathbf{t_j}$ in the set $\mathcal{T}$ and if their distance is lower than a given threshold (we used 0.05 m in our experiments) we consider the $\mathbf{d_i}$ to be a detection of the already-tracked broccoli $\mathbf{t_j}$. Next, we calculate the mean position difference of the $\mathbf{d_i}$-$\mathbf{t_j}$ pairs and we use this value to update the position x_c and velocity v_c of the camera. Then we increment the tracking score n for all associated broccoli in the tracking set $\mathcal{T}$ and update (by means of a running average) their positions and head sizes using the values of the corresponding broccoli in the set $\mathcal{D}$. The broccoli in the set $\mathcal{D}$, which were not associated with any of the tracked ones in the set $\mathcal{T}$ are simply added to the set $\mathcal{T}$. Unassociated broccoli in the set $\mathcal{T}$ are flagged as undetected and those undetected in the last 3 frames are moved to the set $\mathcal{M}$.

Once the data collection is terminated by the operator, the broccoli in $\mathcal{T}$ are added to the set $\mathcal{M}$, which contains the global coordinates of the broccoli, their head sizes and their tracking score. Finally, elements of the set $\mathcal{M}$ with a low tracking score n are deleted, because their low score indicates that they were not detected consistently and thus, they are probably caused by false positive detections, rather than by real broccoli.

Thus, the tracking algorithm not only filters out false positive detections improving the performance of the system, as shown in Fig. 11, but it also produces a map of the detected broccoli heads in the field $\mathcal{M}$ as shown in Figures 17 and 18.

5 | DATA SETS

5.1 | Experimental data

We used the set-up described in Section 3 for collection of experimental data required for evaluation of the proposed system. The four data collection sessions were conducted at three different sites in Lincolnshire, UK and one in Murcia, Spain (see Fig. 8). The sessions were conducted at the beginning and towards the end of harvesting season in UK and at the end of the harvest in Spain. Broccoli plants mature around 80 days post planting and their average height is around 60 cm. The variety of broccoli plants grown in UK ("Iron Man")

(a) (b)

FIGURE 8 Location of the four sites featured in our dataset collection (a) in UK, (b) in Spain

FIGURE 9 An example image sequence from our datasets (300 images from Surfleet A, run 1)

differed significantly from the variety grown in Spain ("Titanium"), which grows with a larger canopy and occluded heads therefore posing bigger challenges for our vision system. The crops in Lincolnshire were grown on a marine silt soil at a density of 60 cm between rows and 38 cm within rows. The broccoli grown in Fuente Álamo di Murcia were grown on a haplic calcisol soil with large content of lime at a density of 90 cm between rows and 30 cm within rows. The weather during UK data capture included a mixture of different conditions including sunny, overcast and raining with broccoli varying in maturity levels from small to larger to already harvested (missing), while the conditions for data capture in Spain included strong sunlight and mature plants at the very end of the harvesting season. The tractor was driven through the broccoli field at a walking speed with two rows of broccoli plants being imaged by the RGB-D sensor (see Fig. 9). Further details on the collected datasets are presented in Table 1.

TABLE 1 Summary of the 3D RGB-D datasets provided. The sequence duration is specified in mm:ss

location/date	GPS	run	duration	frames	fps
Boston, UK/Apr 2015	-	1	01:08	898	13.0
		2	01:10	801	11.3
		3	01:20	979	12.1
		4	01:06	806	12.1
		5	01:00	743	12.2
Surfleet A, UK/Nov 2015	+	1	01:47	776	7.2
		2	04:54	2201	7.5
		3	09:32	4682	8.2
		4	07:53	1580	3.3
Surfleet B, UK/Nov 2015	+	1	08:10	4045	8.2
		2	01:39	823	8.2
		3	18:11	9468	8.7
Fuente Alamo, Spain/Apr 2016	+	1	03:56	1518	6.4
		2	02:52	989	5.7
		3	18:03	6048	5.6
		4	17:30	5865	5.6

5.2 | Ground truth

To evaluate our vision system, we collected ground truth information to test the quality of broccoli detection and localization, and broccoli head size estimation. For broccoli detection and localization, the 3D point clouds from selected data sets were pre-processed using our pipeline, which resulted in a number of extracted clusters for each frame. We then manually labeled these clusters as a broccoli head or not. We used subsets of Surfleet A, UK and Fuente Álamo di Murcia, Spain datasets which we refer to simply as the UK and SPAIN sets. The classifier training data set consisted of randomly selected instances of broccoli and non-broccoli clusters. In addition, we created a combined training dataset to test generality of the training process consisting of selected frames from both UK and SPAIN sets and refer to this set as UK + SPAIN. For the test set we annotated frame sequences from the UK and SPAIN datasets to be also able to evaluate our system with the multi-frame temporal filtering component. A summary of the annotated sets is presented in Table 2.

To evaluate our head size estimators, we collected a set of 100 manual measurements of broccoli head size using tools commonly used in the broccoli production industry, including calipers to measure head diameter (Fig. 10(a)) and weighing scales, for the SPAIN dataset only. The standard procedure requires collecting two diameter measurements along perpendicular axes which are then averaged to get a head size measurement. Figure 10 shows also the head size distribution for the Spanish variety of broccoli to illustrate the variation in the dataset.

TABLE 2 Overview of the Datasets. A summary of the training and testing datasets, and the parameters used for a clustering stage. The pre-processing parameters for the combined UK + SPAIN dataset were determined by the original dataset of a particular instance

	Training Set			Preprocessing Parameters	
	Heads	Background	Test Set	Cluster tolerance	Cluster size (min. - max.)
Dataset	-	-	frames	[mm]	[points]
UK	32	324	600	5	500–10,000
SPAIN	105	1,415	1,169	4	300–10,000
UK + SPAIN	77	752	-	-	-

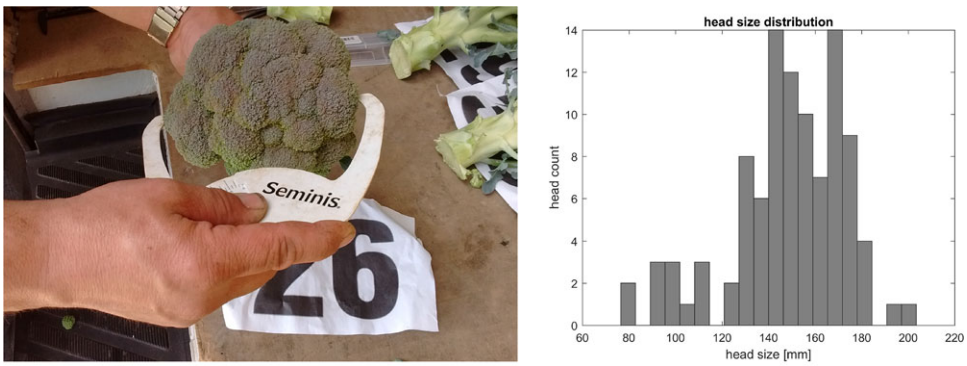


FIGURE 10 Manually collected ground truth data for head size estimation, Fuente Álamo di Murcia, Spain. (a) Measurement process, (b) head size distribution

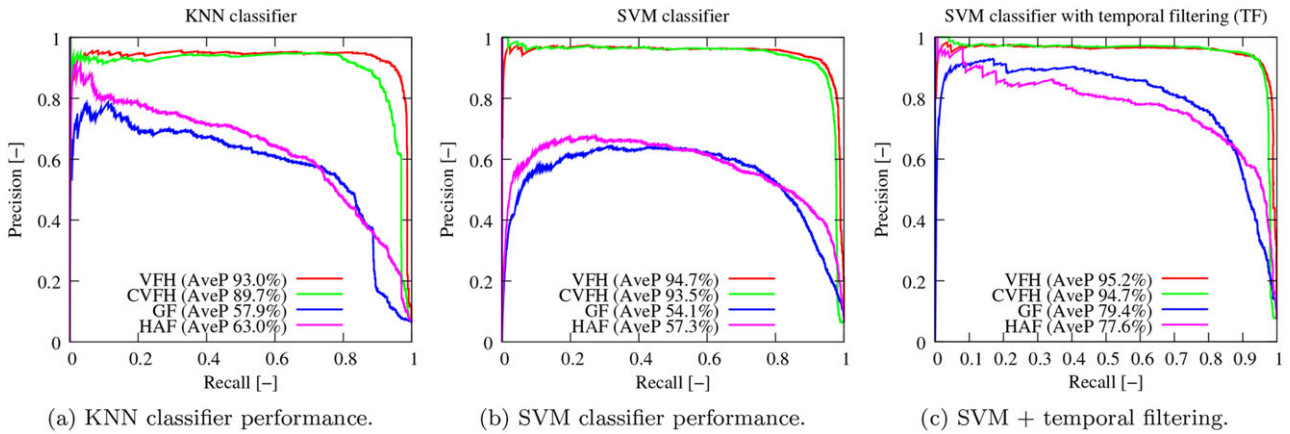


FIGURE 11 Performance evaluation of different 3D features and (a) KNN, (b) SVM, (c) SVM with Temporal Filtering on the UK dataset. VFH features with SVM gives the highest precision at 94.7%. Temporal filtering further improves this to 95.2%

We share all the collected datasets together with accompanying ground truth information at the following website `https://lcas.lincoln.ac.uk/owncloud/shared/agritech-datasets/broccoli/broccoli_datasets.html`.

6 | RESULTS

6.1 | Broccoli detection

6.1.1 | System performance

We evaluated the vision system for detecting broccoli heads using multiple feature descriptors as mentioned in Section 4 and using two classifiers KNN and SVM on the UK dataset. The parameters of the pre-processing algorithms, feature descriptors and classifiers were tuned according to a validation set, as shown in Table 2. We also tried to train the best performing classifier, SVM, using all the implemented features but this did not lead to better performance.

The best C value for the linear SVM was chosen as 0.002 for the best performing features, by using cross-validation on a grid search. The number of nearest neighbors was chosen as 11.

We use precision-recall curves to evaluate the performance of the classifier and report the average precision as in Ref. 27. Precision represents a ratio of true positive detections to the total number of detections and recall is a ratio of true positive detections to the total number of actual instances present. The precision and recall values are computed over a range of confidence score thresholds of the classifier. We report the results of the experiments in Figure 11 using average

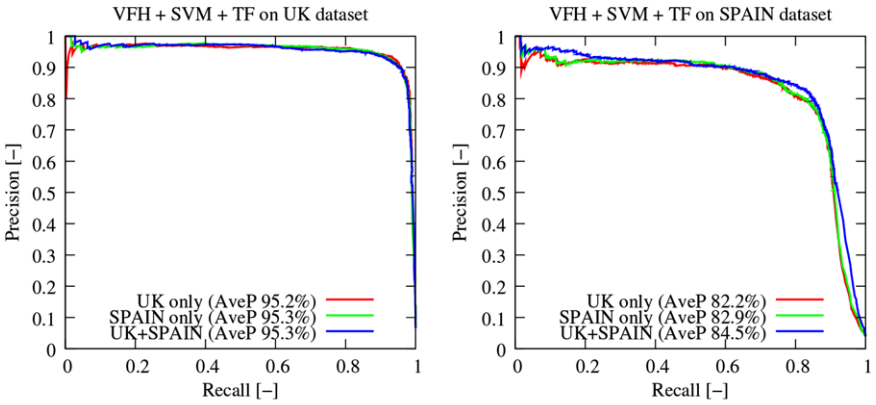


FIGURE 12 Performance evaluation of VFH features, SVM classifier and temporal filtering trained on the UK, SPAIN and UK+SPAIN datasets and tested on the UK dataset (a) and the SPAIN dataset (b)

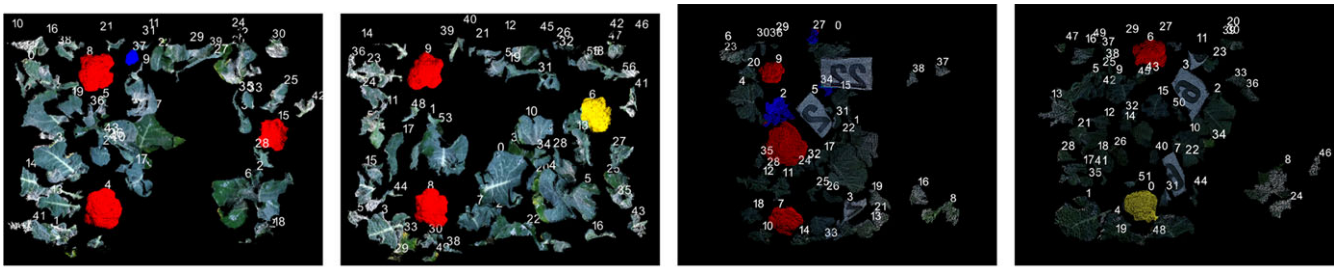


FIGURE 13 Selected examples of false positive (blue) and false negative (yellow) detections in the UK (a,b) and SPAIN (c,d) datasets

precision. The results show that the surface normal geometry based features, VFH along with SVM, gives the highest accuracy on the UK test data of 94.7%. We show that temporal filtering improves the average precision of all the feature combinations using SVM on the UK dataset.

The average run time of the entire pipeline per image is 5-6s on an Intel i7 CPU, 3.4 GHz.

6.1.2 | Cross-validation between the UK and Spain datasets

We also conducted the experiments to verify the generalization capability of the models trained on the two alternative datasets and also their combination. Hence we show in Figures 12 a and 12 b the results of the evaluation of models trained on the UK, SPAIN and UK+SPAIN data on the two test datasets, UK and SPAIN. We can see that while using the SPAIN dataset for testing, the model trained only on the UK dataset gives similar performance to the model trained only on the SPAIN data. It is also evident that while testing on the UK dataset, the models trained on SPAIN and the UK+SPAIN datasets give similar results. This can be attributed to the generalization capability of the trained models across these two datasets. The relatively lower precision given on the Spanish data arises from the increased occlusions of the heads from the close assembly of leaves and thicker foliage found in the Spanish variety compared to the UK counterpart. This can affect the segmentation step, which leads to missing the heads or merging of clusters containing heads and leaves. For both datasets, the segmentation step is the most sensitive to head occlusions, but some convex clusters corresponding to leaves are sometimes mistakenly identified as broccoli. Selected examples of false positive and false negative detections are presented in Figure 13.

6.2 | Feature analysis

We analyse the models derived from training the point clouds representing broccoli heads and background on the UK, SPAIN and UK+SPAIN datasets, as shown in Figure 14. The 45 bins each represent the roll, pitch and yaw angles between the viewpoint vector and each of the surface normals. The next 45 bins corresponds to the histogram of distances of all the points to the centroid. The last 128 bins represent the viewpoint component which is a histogram of angles between the viewpoint vector and each of the normals. We can see that the profile of weights for the UK and SPAIN data are quite similar with major peaks concentrated around the same features. This suggests that the learnt models generalize well to different varieties of broccoli. The weights in the combined model (UK+SPAIN) are more evenly distributed. The distance component and first part of the viewpoint component were not captured in all three models, which indicates their low importance for this application.

6.3 | Head size estimation

We have compared the output of our two algorithms for broccoli head size estimation against 100 manually collected ground-truth measures on the SPAIN dataset (see Fig. 15). Typical mean-based

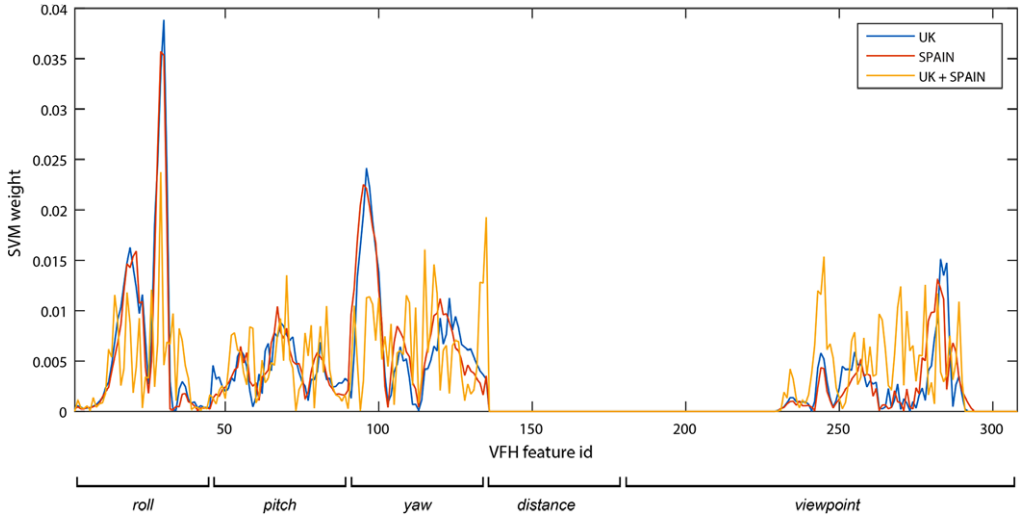


FIGURE 14 Feature analysis for models trained on VFH features, SVM classifier on UK, Spanish and UK + Spanish datasets

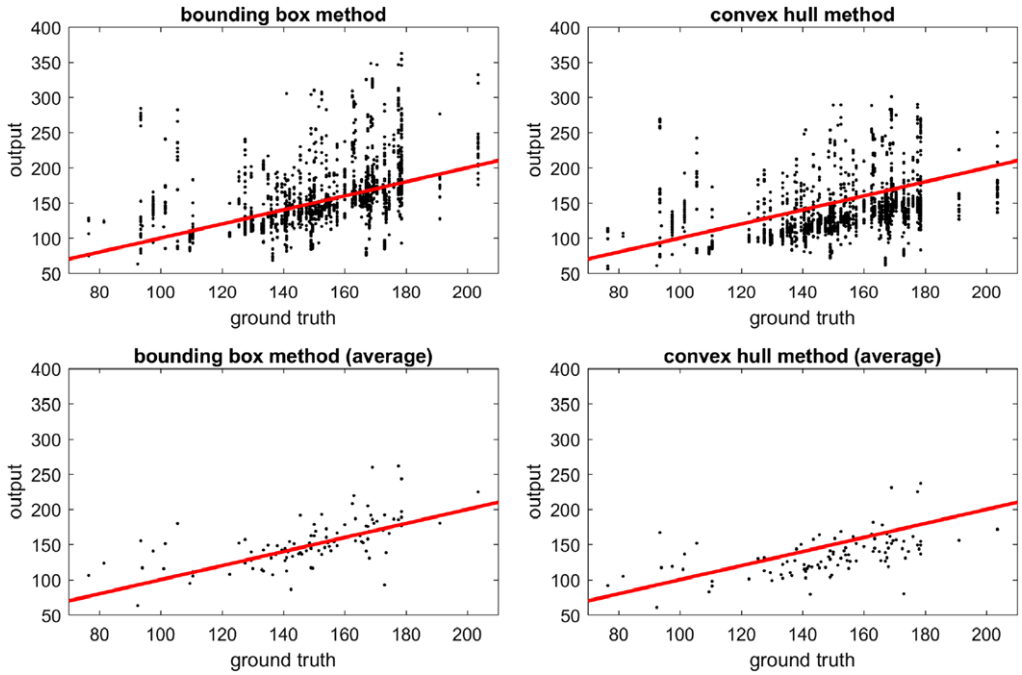


FIGURE 15 Two head size estimators compared to ground truth: (left) bounding box method, (right) convex hull method. The top row corresponds to direct output from each individual frame while the bottom row to the averaged estimates for each individual broccoli head

statistics have not produced informative summaries as the error distributions are heavily tailed on one side. Therefore, to estimate the accuracy and precision of both estimators we calculated median-based statistics: accuracy using median error $MEDE = median(e_i)$, and precision using median absolute deviation $MAD = median(|e_i - \tilde{e}|)$. The bias of the bounding box estimator was $MEDE = -0.9$ mm whilst its precision was $MAD = 12.8$ mm. The bias of the convex hull-based estimator was larger ($MEDE = -22.0$ mm) but the estimates were more accurate ($MAD = 9.4$ mm). If the bias can be accounted for, this would favor the convex-hull based method as having the better precision. The latter method is however more computationally demanding and the final choice would depend on the computational resources available. The Pearson's linear coefficient for the bounding box estimator was $r = 0.63$ and for the convex hull method $r = 0.55$ which confirms the overall better agreement with the ground truth of the former approach. The distribution of errors e_i for both cases has a similar shape regardless of the viewpoint distance (see Fig. 16). This is a desirable property as both size estimation methods can be applied at almost any sensor angle without significant change in the results.

6.4 | Temporal filtering

Finally, we provide the results of the SVM-based classification augmented by the temporal filtering method for tracking the 3D positions

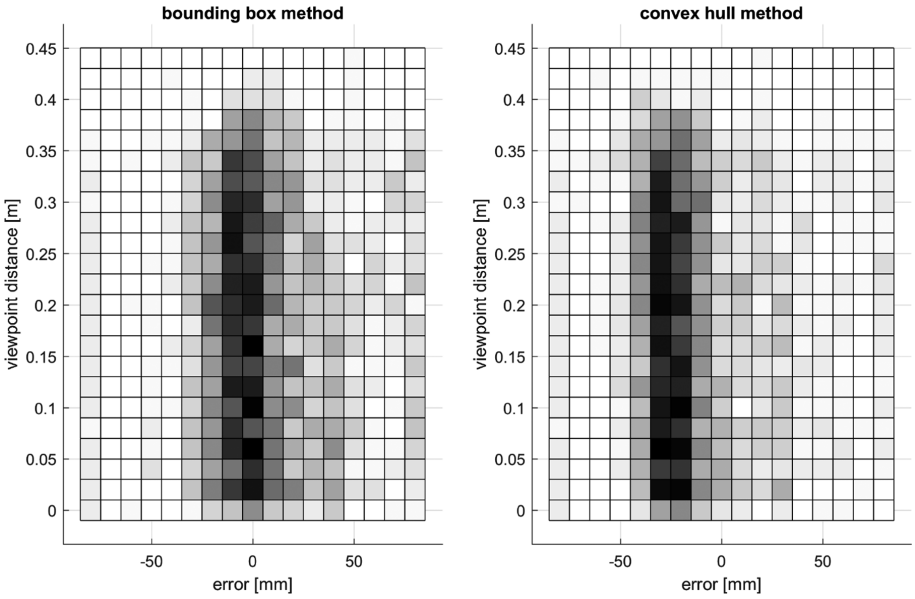


FIGURE 16 Distribution of errors with respect to viewpoint distance from the optical center of the sensor for two different head size estimation methods: (left) bounding box, (right) convex hull

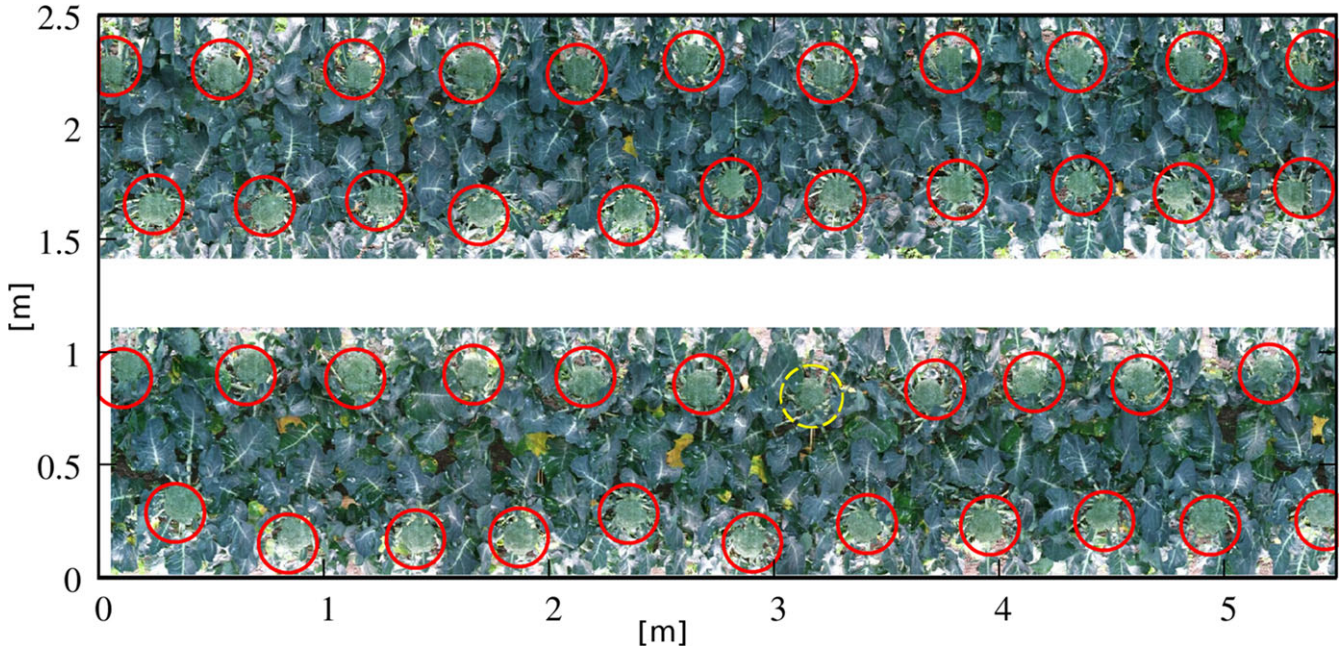


FIGURE 17 A partial map of broccoli locations for the UK dataset, generated from the testing data containing sequences from 2 tractor rows (4 rows of broccoli)

of the detected broccoli heads, as described in Section 4.5. Figure 11 c shows that rejecting detections which were not consistent over several frames improved the precision of the classification. The figure indicates that the combination of the Viewpoint Feature Histogram, Support Vector Machine and temporal filtering results in a system that detects broccoli heads with more than 95% precision.

The corresponding maps of the detected broccoli heads for both UK and Spain datasets are shown in Figures 17 and 18. The maps are shown in 2D (x, y) coordinates, while the z-dimension (not shown) comprises the corresponding depth estimates for the detected broccoli heads.

7 | CONCLUSION

This paper demonstrated the development of a 3D-based approach for detecting mature broccoli heads that could be applied in an automatic broccoli harvester. We showed that the depth images provided by low-cost RGB-D sensors can be used for reliable detection and localization of broccoli heads in cluttered outdoor field conditions. We also showed that the information from a sequence of detections can be used to reliably track the individual broccoli and filter out false detections with a precision rate of 95.2% on the UK data and 84.5% on the Spain data. Cross-validation of the system trained on the UK dataset on the

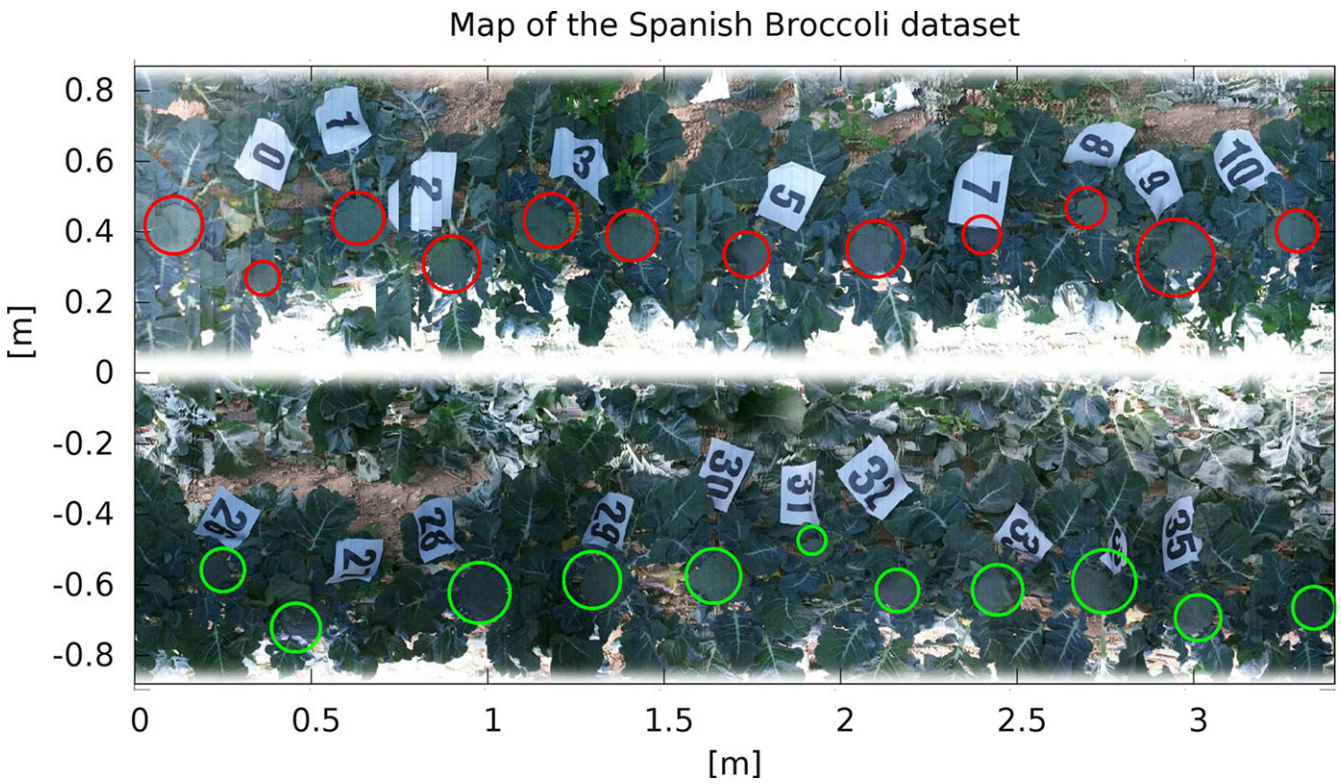


FIGURE 18 Map of locations of tracked detections of broccoli for the Spain data

Spanish dataset, and vice versa, indicated good generalization capabilities of the system, confirming the strong potential of low-cost 3D imaging for commercial broccoli harvesting. Additionally we evaluated two alternative methods for automatic estimation of the broccoli head size, which could be used to inform a cutting robot on when each individual head is ready for harvest.

Future work will include fusion of GPS and IMU data to geo-locate the broccoli, enabling further applications in field mapping and yield prediction. Texture features from the RGB images could also be added to further improve the results. The described methods could be further used to establish effective decision support on likely yields and cost benefits in terms of when to harvest a field, and underpinning information on yield variation within and between fields.

Feature selection methods such as adaptive boosting algorithms could also be used to reduce the dimensionality of the input vector to the classifiers, and thus reduce computational costs.

ACKNOWLEDGMENTS

The work was funded by BBSRC and Innovate UK, project No:BB/N004841/1 and by CSF project no. 17-27006Y. Many thanks to Adam Turner for all his help with ground truthing the datasets used in this paper.

REFERENCES

1. McCarthy CL, Hancock NH, Raine SR. Applied machine vision of plants: A review with implications for field deployment in automated farming operations. *Intell Serv Robotic*. 2010;3(4):209–217.
2. Jimenez A, Ceres R, Pons J. A survey of computer vision methods for locating fruit on trees. *T ASAE*. 2000;43(6):1911.
3. Hayashi S, Shigematsu K, Yamamoto S, et al. Evaluation of a strawberry-harvesting robot in a field test. *Biosyst Eng*. 2010;105(2):160–171.
4. Ji W, Zhao D, Cheng F, Xu B, Zhang Y, Wang J. Automatic recognition vision system guided for apple harvesting robot. *Comp Electric Eng*. 2012;38(5):1186–1195.
5. Okamoto H, Lee WS. Machine vision for green citrus detection in tree images. *Environ Control Biol*. 2010;48(2):93–99.
6. Wachs JP, Stern H, Burks T, Alchanatis V. Low and high-level visual feature-based apple detection from multi-modal images. *Precis Agric*. 2010;11(6):717–735.
7. Haug S, Michaels A, Biber P, Ostermann J. Plant classification system for crop/weed discrimination without segmentation. In *2014 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2014:1142–1149.
8. Sa I, McCool C, Lehnert C, Perez T. On visual detection of highly-occluded objects for harvesting automation in horticulture. In *Proc. ICRA. Seattle*, Washington: 2015.
9. Kapach K, Barnea E, Mairon R, Edan Y, Ben-Shahar O. Computer vision for fruit harvesting robots–state of the art and challenges ahead. *Int J Comput Vision Robotic*. 2012;3(1-2):4–34.
10. Barnea E, Mairon R, Ben-Shahar O. Colour-agnostic shape-based 3D fruit detection for crop harvesting robots. *Biosyst Eng*. 2016.
11. Sa I, Lehnert C, English A, et al. Peduncle detection of sweet pepper for autonomous crop harvesting - combined color and 3-D information. *IEEE Robot Autom Lett*. 2017;2(2):765–772.
12. van Henten EJ, Hemming J, VanTuijl B, et al. An autonomous robot for harvesting cucumbers in greenhouses. *Auton Robot*. 2002;13(3):241–258.

13. Weiss U, Biber P. Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. *Robot Auton Syst*. 2011;59(5):265–273.
14. Nakarmi AD, Tang L. Within-row spacing sensing of maize plants using 3d computer vision. *Biosyst Eng*. 2014;125:54–64.
15. Gai J, Tang L, Steward B. Plant recognition through the fusion of 2D and 3D images for robotic weeding. In *2015 ASABE Annual International Meeting*. American Society of Agricultural and Biological Engineers; 2015:1.
16. Li J. *3D machine vision system for robotic weeding and plant phenotyping*. PhD thesis, Iowa State University; 2014.
17. Nguyen TT, Vandevoorde K, Kayacan E, De Baerdemaeker J, Saeys W. Apple detection algorithm for robotic harvesting using a RGB-D camera. In *International Conference of Agricultural Engineering, Zurich, Switzerland*. 2014.
18. Ramirez RA. *Computer vision based analysis of broccoli for application in a selective autonomous harvester*. Master's thesis, Virginia Polytechnic Institute and State University; 2006.
19. Tu K, Ren K, Pan L, Li H. A study of broccoli grading system based on machine vision and neural networks. In *Proc. ICMA*. IEEE; 2007:2332–2336.
20. Fankhauser P, Bloesch M, Rodriguez D, Kaestner R, Hutter M, Siegwart R. Kinect v2 for mobile robot navigation: Evaluation and modeling. In *in Proc. ICAR*. 2015:388–394.
21. Yang L, Zhang L, Dong H, Alelaiwi A, Saddik AE. Evaluating and improving the depth accuracy of kinect for Windows v2. *IEEE Sensors J*. 2015;15(8):4275–4285.
22. Rusu RB, Cousins S. 3D is here: Point Cloud Library (PCL). In *Proc. ICRA*. Shanghai, China: 2011.
23. Rusu RB, Bradski G, Thibaux R, Hsu J. Fast 3d recognition and pose using the viewpoint feature histogram. In *Proc. IROS*. 2010:2155–2162.
24. Aldoma A, Vincze M, Blodow N, et al. CAD-model recognition and 6DOF pose estimation using 3D cues. In *IEEE International Conference on Computer Vision Workshops*. Barcelona, Spain: 2011:585–592.
25. Karpathy A, Miller S, Fei-Fei L. Object discovery in 3d scenes via shape analysis. In *Proc. ICRA*. 2013:2088–2095.
26. Aldoma A, Marton Z-C, Tombari F, et al. Point cloud library. *IEEE Robot Autom Mag*. 2012;1070(9932/12).
27. Everingham M, VanGool L, Williams CK, Winn J, Zisserman A. The PASCAL visual object classes (VOC) challenge. *Int J Comp Vision*. 2010;88(2):303–338.

How to cite this article: Kusumam K, Krajník T, Pearson S, Duckett T, Cielniak G. 3D-vision based detection, localization, and sizing of broccoli heads in the field. *J Field Robotics*. 2017;34:1505–1518. https://doi.org/10.1002/rob.21726