A Field Tested Robotic Harvesting System for Iceberg Lettuce

Simon Birrell
Department of Engineering
University of Cambridge
Cambridge, CB2 1PZ
sab233@cam.ac.uk

Josie Hughes
Department of Engineering
University of Cambridge
Cambridge, CB2 1PZ
jaeh2@cam.ac.uk

Julia Yunlu Cai
Department of Engineering
University of Cambridge
Cambridge, CB2 1PZ

Fumiya Iida
Department of Engineering
University of Cambridge
Cambridge, CB2 1PZ
fi224@cam.ac.uk

Abstract

Agriculture provides an unique opportunity for the development of robotic systems; robots must be developed which can operate in harsh conditions and in highly uncertain and unknown environments. One particular challenge is performing manipulation for autonomous robotic harvesting. This paper describes recent and current work to automate the harvesting of the iceberg lettuce. Unlike many other produce, iceberg is challenging to harvest as the crop is easily damaged by handling and is very hard to detect visually. A platform called Vegebot has been developed to enable the iterative development and field testing of the solution, which comprises of a vision system, custom end effector and software. To address the harvesting challenges posed by iceberg lettuce a bespoke vision and learning system has been developed which uses two integrated networks to achieve classification and localisation. A custom end effector has been developed to allow damage free harvesting. To allow this end effector to achieve repeatable and consistent harvesting, a control method using force feedback allows detection of the ground. The system has been tested in the field, with experimental evidence gained which demonstrates the success of the vision system to localise and classify the lettuce, and the full integrated system to harvest lettuce. This work demonstrates how existing state-of-the art vision approaches can be applied to agricultural robotics, and mechanical systems can be developed which leverage the environmental constraints imposed in such environments.

1 Introduction

The story of agriculture is one of increasing automation. Crops are planted, weeded and harvested with ever decreasing direct human involvement, reducing labour costs and improving yield. However, every fruit or vegetable is different, and solutions for a single crop can vary from country to country and even company

to company. While some crops such as wheat or potatoes have long been harvested mechanically at scale, many others such as sugar beet (Nieuwenhuizen et al., 2010), kiwi (Scarfe et al., 2009), cucumbers (Van Henten et al., 2002), citrus fruit (Harrell et al., 1990), strawberries (Hayashi et al., 2010), broccoli (Kusumam et al., 2016), grapes (Luo et al., 2016; Monta et al., 1995) and many others (Bac et al., 2014) have resisted commercial automation. Agricultural robotics presents unique challenges compared to robotics in the more common factory environments (Oetomo et al., 2009). Agricultural environments are unstructured, intrinsically uncertain, harsh on mechanical equipment (Reddy et al., 2016) and have high variability over weather conditions, locations and time. Autonomous agricultural systems must be flexible and adaptive (Hajjaj and Sahari, 2016; Edan et al., 2009) to cope. Harvesting and other crop manipulation tasks (Kemp et al., 2007; Hughes et al., 2018), are particularly challenging (Bac et al., 2014) along all these dimensions.

The iceberg lettuce is still harvested by hand using a hand-held knife, and presents two main challenges to automation. First, visually identifying the vegetable's location and suitability for harvesting in what appears to be a sea of green leaves is hard even for humans (Figure 1a). Any solution must be robust to the variation in individual lettuces, with their appearance varying greatly over weather conditions, maturity and surrounding vegetation. Second, in a terrain with an uneven ground the lettuce stem must be cut cleanly at a specified height to meet commercial standards, while the lettuce head can easily be damaged by unpractised handling. A lettuce harvesting solution should therefore incorporate a high-precision, high force cutting mechanism while being capable of handling the vegetable delicately. There is a growing need for automated, robotic iceberg lettuce harvesting due to increasing uncertainty in the reliability of labour and to allow for more flexible, 'on-demand' harvesting of lettuce.

This work investigates automating the harvesting of iceberg lettuce with three key research goals. Firstly, how vision systems can be developed using off-the-shelf convolutional neural networks as opposed to hand-tailored computer vision pipelines, with pragmatic architectural adjustments made to allow for the datasets available. Secondly, how mechanical systems can be developed to work within the operational constraints imposed by the agricultural environment. Finally, how field robots can be developed to allow rapid integration and hence testing in the field.

This paper describes the results to date of the Vegebot project, where a lettuce harvesting robot has been developed using an approach of rapid iterative design, prototyping and field testing. Two key methods are described for automating the harvesting of the iceberg lettuce under challenging and uncertain field conditions. First, the lettuces are localised and classified using a data-driven approach. This is implemented using two convolutional neural networks, the architecture being shaped by the datasets available. Using this method in field tests, a localisation success of 91% in field tests was achieved, and the crop accurately classified. Second, the lettuces are harvested with a custom designed end effector that incorporates a camera, pneumatics, a belt drive and a soft gripper. The end effector cuts the lettuce stems efficiently while grasping the lettuce head in a way that avoids damage. As the ground is uneven and its depth hard to detect under the foliage, a force feedback control system is used to detect when the end effector has reached the correct position to make the cut and achieve a consistent cutting height.

Following a review of the state of the art in crop harvesting, Section 3 defines the problem posed by iceberg lettuce harvesting and outlines the overall system that was developed. Section 4 focuses on the details of the two harvesting methods developed: the vision system and end effector. The field tests and experimental results are detailed in Section 5 and the paper concludes with a discussion and conclusion that suggests the application of the techniques and approaches in this work to other agricultural challenges.

2 State of the Art

There is prior work on vision techniques for agriculture. Many of the examples in the literature are from before the use of convolutional neural networks (CNNs) in the late 2000s, and so use a wide variety of hand-crafted features. The detection of volunteer potato plants was performed using adaptive Bayesian classification of Canny Edge Detectors among other features (Nieuwenhuizen et al., 2010). Broad-leaved dock detection (a weeding task) was performed using a texture-based approach, where image tiles were subjected to a Fourier Analysis (Evert et al., 2011). (Weeding is a similar task to harvesting, just with less concern for the fate of the extracted plant). An alternative approach to weed detection used wavelet features of Near Infrared (NIR) imagery (Scarfe et al., 2009), subsequently passed to a PCA component and a k-means classifier (Kiani et al., 2010). Grapes have also been detected with Canny Edge filters, using Decision Trees as the classification mechanism (Berenstein et al., 2010). Foliage detection on the same project required a separate algorithm. Grapes were classified on another project using the AdaBoost framework, which combined the results of four weak classifiers into one strong one (Luo et al., 2016). Radicchios have been detected by thresholding Hue Saturation Luminance (HSL) images and applying particle filters (Foglia and Reina, 2006). Cucumbers were detected using NIR photography at two positions 5cm apart, to give stereoscopic depth information (Van Henten et al., 2006) and classified for maturity by estimating their weight from the perceived volume (Van Henten et al., 2002). A more recent experiment detected Broccoli heads using an RGB-D sensor had the disadvantage that the robot had to move a tent across the field to prevent interference from outdoor light. Point clouds were clustered from the depth information, outliers were removed and Viewpoint Feature Histograms constructed. A Support Vector Machine performed the actual classification of the broccoli heads (Kusumam et al., 2016). The use of vision to provide control through methods including visual servoing has also been shown to increase positional accuracy when harvesting citrus fruit (Mehta and Burks, 2014; Mehta et al., 2016).

These solutions are not appropriate for iceberg lettuce. Colour cues as used in (Foglia and Reina, 2006; Berenstein et al., 2010; Cubero et al., 2015) are less useful because the lettuces appear to be a 'sea of green'. Depth cues, as used in (Kusumam et al., 2016; Rajendra et al., 2008) also provide limited information because the plants and their leaves overlap and the heads are often hidden.

Similarly, there are a number of existing autonomous harvesting systems. Harvesting is a challenging task to automate and a recent review came to the gloomy conclusion that almost no progress had been made in the past thirty years (Bac et al., 2014). Many research projects have been performed, but little has filtered through into the commercial world. The more successful projects include a harvester for apples (Silwal et al., 2017) using a suction method, rice harvesting using custom harvesting systems (Kurita et al., 2017) and a sweet pepper harvesting system (Bac et al., 2017). There has also been significant work in the development of autonomous weeding or grading systems including a sugar beet classifying system (Lottes et al., 2017) and a grape pruning system (Botterill et al., 2017). There are a number of patents specifically relating to the harvesting of iceberg lettuce (Ottaway, 1996; Shepardson and Pollock, 1974; Ottaway, 2009), however, these have not been demonstrated under field conditions and do not clearly demonstrate how selective plant harvesting is possible. These previous approaches include using a belt driven band-saw type mechanisms or water jet cutting. These approaches have limitations, most notably that the outer-leaves of the lettuce can be easily damaged when harvesting and there is a lack of reliability in stem cutting height and quality.

(a) (b)

Figure 1: a) The challenging localisation and classification problem posed by the lettuce field. b) The existing harvesting method.

3 Problem Definition & System Architecture

3.1 Problem

The lettuces to be harvested must be both localised (their position detected) and classified according to their suitability for picking. For a mature lettuce, using the custom end effector, the lettuce head centre must be localised to within approximately 2cm of the ground truth position. The identified classes should include at a minimum (1) harvest-ready lettuces (which may be picked immediately) (2) immature lettuces (which can be returned to later) and (3) infected lettuces (which should not be touched with the end effector so as to avoid spreading the infection). The vision system should operate under varying weather and lighting conditions.

Once a harvest-ready lettuce has been identified it must be cut to supermarket standards. This is currently performed by a human worker with a knife. The worker tilts the head of the lettuce and then uses a high impulse manoeuvre to cut the stem of the lettuce. The lettuce is then bagged and placed on a harvesting rig (see Figure 1b.) There is a high degree of dexterity and accuracy required to achieve a supermarket-quality cut. The lettuce must have a stem of the correct length (1-2mm protruding), it must be clean, with minimal browning and have no damage to outer leaves. Additionally, if outer-leaves remain after harvesting, these should be removed, which has proved to be a challenging manipulation problem in itself (Hughes et al., 2018). If the lettuce falls outside these requirements it is not accepted by supermarkets. A lettuce worker can harvest a lettuce in under 10 seconds, which sets the benchmark for a robotic harvesting system.

There are also a number of constraints arising for the agricultural environment, which dictate the form factor and design decisions, these are summarised in Table 1.

3.2 System Architecture

The system developed for autonomous iceberg lettuce harvesting (Vegebot) is shown in Figure 2. Vegebot comprises a laptop computer running control software, a standard 6 degree of freedom (DOF) UR10 robot arm, two cameras and a custom end effector, all housed on a mobile platform for field testing. A block diagram showing the integration of the system is shown in Figure 3.

Table 1: Conditions for the design and development of a lettuce harvesting system determined by the agricultural environment.

	Parameter	Specification	Influence on design
Environment	Width of lettuce lanes	2	Determines width of platform
	Spacing between lettuce	30cm	Determines max size of end effector
	Height of lettuce plants	30cm	Determines of height of platform
	Diameter of lettuce	20cm	Determines size of end effector
	Diameter of lettuce stem	approx 30mm	Determined blade specification
Robot	Generator Power	240V, 2kW	Sufficient to power all systems
	Compressor Air Pressure	8 bar	Sufficient for pneumatics
	Vegebot Dimensions	2m x 0.6m x 0.5m	Fits within Lettuce lanes

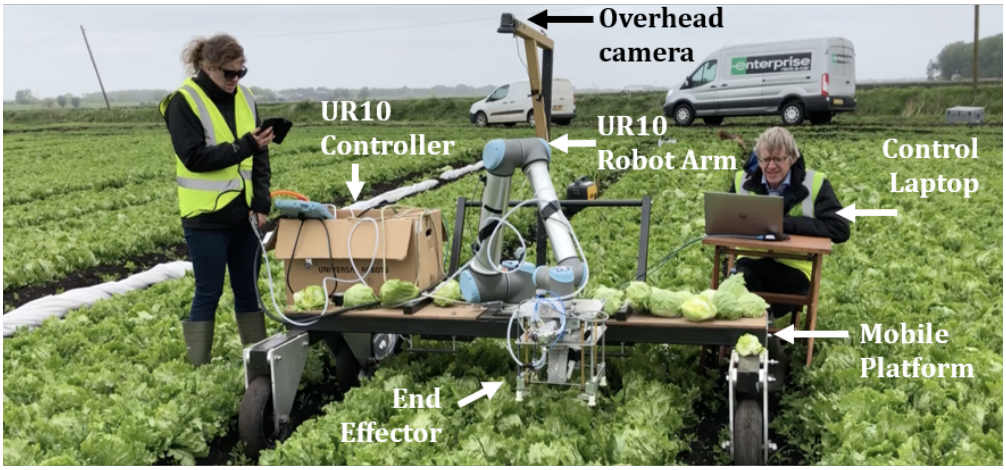


Figure 2: The Vegebot harvesting system, shown undergoing field experiments.

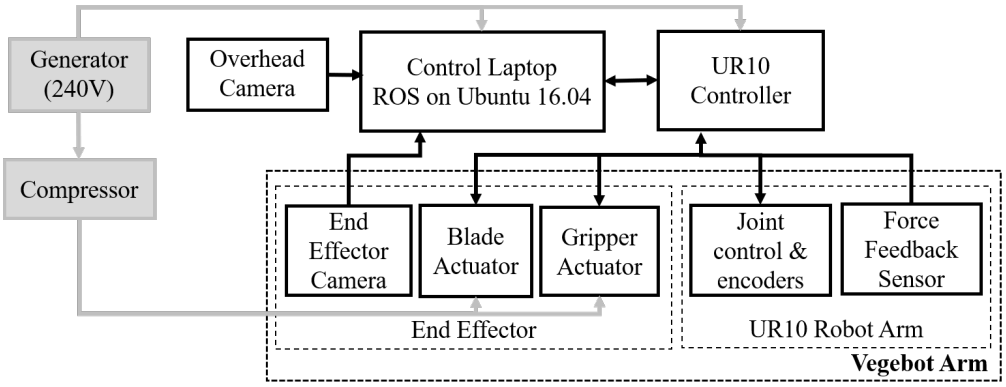


Figure 3: Block diagram of the robotic lettuce harvester system developed.

Vegebot contains two cameras: an *overhead camera* positioned approximately 2 meters above the ground and another *end effector camera* mounted inside the end effector. Both are ordinary, low-cost USB webcams and stream video to the *control laptop*. Together, these allow Vegebot to detect (localise and classify) lettuces, and to move the end effector into position. There are additional sensors built into the *robot arm*: the standard *joint encoders* and a *force feedback sensor* that records the force and torque being applied to the end effector.

The UR10 arm provides a wide range of movements, and provides force and torque information allowing force feedback to be implemented. A commercial implementation would likely have simpler arms each with an end effector, all operating in parallel (for an example of such a system, see (Scarfe et al., 2009). The control laptop controls the *end effector* using two digital I/O lines routed through the UR10 arm. These switch the two pneumatic actuators on and off, the *blade actuator* causing the blade to slice through the lettuce stalk and retract, while the *gripper actuator* causes the soft gripper to grasp and release the target lettuce.

The mobile platform supports the above hardware items and is moved manually around the field. The system is powered by a generator, which provides sufficient power to meet the peak demands of the system. An air compressor is used to enable actuation of the pneumatic systems. The generator and compressor can sit on the Vegebot to allow the system to be completely mobile.

The software architecture is shown in Figure 17a and detailed in Appendix A. The web-based user interface is shown in Figure 17b.

3.2.1 Control & Processes

The processes for training and operating Vegebot can be analyzed at three levels (see Figure 4). At the highest level, the *Learning Cycle*, datasets are gathered for the initial training of the vision system, harvesting is performed and additional data is gathered. As soon as enough new data is gathered to merit it, the system can be retrained. In this way, the accuracy and generalization abilities of the Vegebot can in principle be improved as images are obtained from new fields and under different weather conditions. The testing of these improvements is the subject of a future paper.

The *Harvesting Session* outlines the structure of the work in the field. First the Vegebot is moved along the lettuce lanes (seen in Figure 2) to bring approximately 10 lettuces within the robot's workspace and field of view. The current iteration of Vegebot is simply pushed into position. Next, the Vegebot is optionally

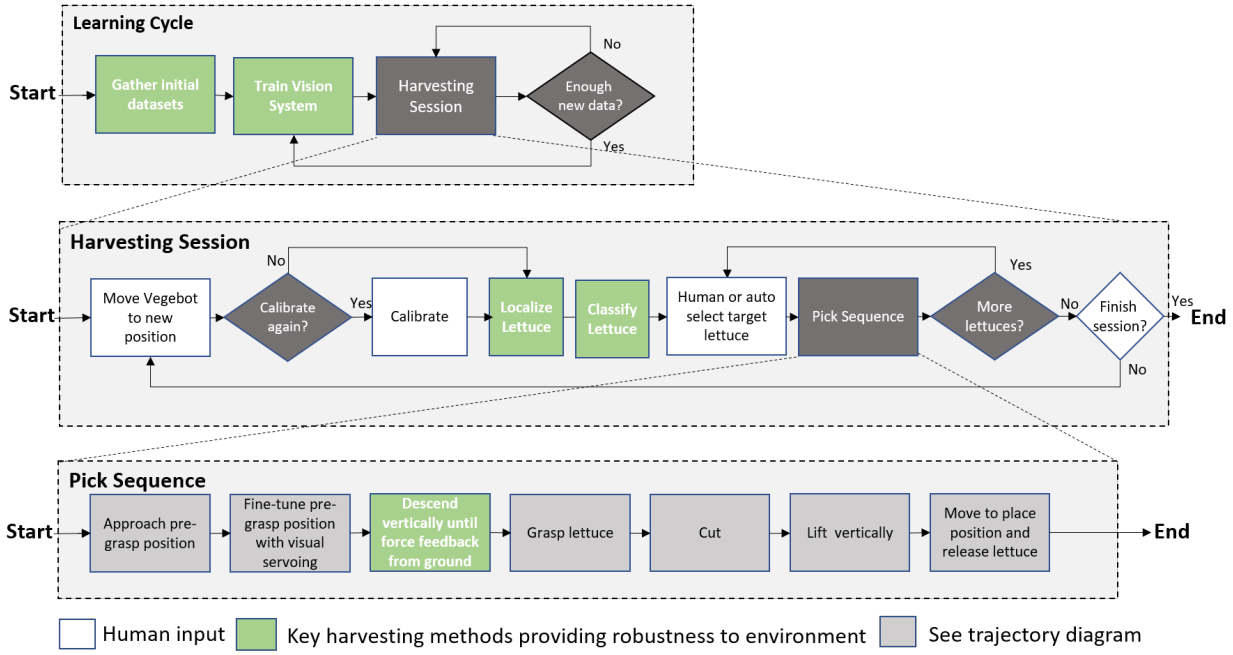


Figure 4: Processes for training and operation of the Vegebot, showing the key processes in green. The trajectory diagram for the lowest level pick sequence is shown in Figure 14

calibrated, using the method described in Section 4.1.3. *Calibration* is always performed at the start of a session and then on an as-needed basis as discrepancy between the lettuce position inferred by the overhead camera and that detected by the end effector camera increases.

Next, the vision system *detects lettuces* in the video feed from the overhead camera. A human then *selects a lettuce* by clicking on the user interface. This was a manual process during the experiments for the sake of safety. Selection could be automated with a trivial modification. The Pick Sequence then begins, with the lettuce being picked and placed onto the platform. Once the reachable lettuces have been picked, the Vegebot can either be moved to a new position or the session finished.

The *Pick Sequence* is fully automated and comprises seven stages. First, the end effector *approaches the pre-grasp position*, a point centred approximately 10cm over the inferred top of the lettuce, based on the localisation predictions from the overhead camera. Because of the rugged nature of the environment and the impacts received by the Vegebot, this prediction is inevitably inaccurate to a greater or lesser degree. At this point, the camera in the end effector takes over to *fine tune* the end effector position to be directly over the centre of the lettuce. The end effector then *descends vertically* down over the lettuce until the force feedback sensor registers the upward force of the ground resisting the downward trajectory. The soft gripper is then activated and *grasps the lettuce*. Next, the blade actuator is activated and the blade moves horizontally and *cuts* through the lettuce stalk. Still grasping the lettuce, the end effector then *lifts vertically* to the same height as the pre-grasp position, clearing it from contact with the surrounding lettuces. The arm then moves the end effector to a convenient *place position* where the soft gripper is deactivated and the lettuce is released.

The following section addresses key the harvesting methods which have been implemented to allow robust and reliable harvesting in the agriculture environment (and are shown in green boxes in Figure 4).

4 Harvesting Methods

4.1 Lettuce Localisation & Classification

The lettuce *detection* process comprises both *localisation* (discovering where the lettuce is relative to the robot) and *classification* (determining whether the lettuce is a suitable candidate for being harvested). Lettuces heads are variable in appearance and are typically partially or wholly occluded by their own leaves and by leaves of neighbouring lettuces. The outdoor lighting conditions also vary drastically with different weather, including very different levels of brightness and contrast. The lettuces need to be classified as "Harvest Ready" (for immediate picking), "Immature" (for picking at a later date) or "Infected" (to be avoided and reported). Additionally, the localisation system must transform the viewport coordinates of the lettuce into robot-centric coordinates for picking in the face of very rugged physical conditions. All these operations must be performed in close to real time given that Vegebot uses localisation information dynamically to fine-tune the trajectory of its end effector.

In principle, any of the latest crop of deep-learning based object detectors could fulfill this function. Candidates such as YOLOv3 and Faster R-CNN (Redmon and Farhadi, 2018; Ren et al., 2015) can both provide object bounding boxes and class labels in real time (Hui and Hui, 2018). In this case, YOLOv3 was chosen as it gave the fastest detection times and its principal disadvantage (poor performance on very small close-together objects) was irrelevant in this use case. Fast detection times on a laptop implied the possibility of later re-implementing the algorithm on more modest, embedded hardware.

With a large enough detection dataset, rich in examples of all lettuce categories, there would be little more to do. In the present project there were only two datasets available. The first was a detection dataset gathered by one of the authors (see Figure 6), with images captured by a webcam and bounding boxes and class labels added manually. This dataset (detailed in Table 2) was rich in positional data but the less common classes such as "Infected" were under-represented. The second dataset originated from a previous student project (Nagrani, 2016) in lettuce classification and was rich in examples of all classes, but had no useful positional information, all lettuces being in the centre of each image.

Ideally, a more extensive detection database would have been gathered from multiple fields and stages of the crop cycle, to fully represent the position and location of exemplars of all classes. Alternatively, the existing classification images could have been inserted over other backgrounds to produce an artificial training set for detection. This latter strategy runs the risk of the network learning to detect artefacts in the synthetic images, rather than genuinely localising the vegetables based on natural visual cues.

Instead, the solution chosen was to divide the pipeline into two networks (see Figure 5), each trained by one of the existing datasets. The first network, a YOLOv3 object detector would be used simply to discover the presence and location of lettuces (the number of classes being reduced to a single 'lettuce' class) and output their bounding boxes. Narrow bounding boxes, likely caused by lettuces at the edge of the viewport and out of reach of the arm, are rejected as candidates. Each of the remaining bounded boxes is then cropped (adding a small margin round the outside of the bounding box to provide more visual information to the next stage) and then a second Darknet Object Classification Network was applied to each. Finally, bounding boxes predicted by the first stage and the class labels predicted by the second stage are merged. Although requiring a two-stage network, this approach offers greater performance of both localisation and classification. The architecture has been chosen to achieve the best performance with the datasets available and given the information content of those datasets.

There is an additional advantage to using a two-stage network. Images input to YOLO are re-sized from 1920x1080 to a resolution of 320 by 320. This is still enough visual information to distinguish, say, a man

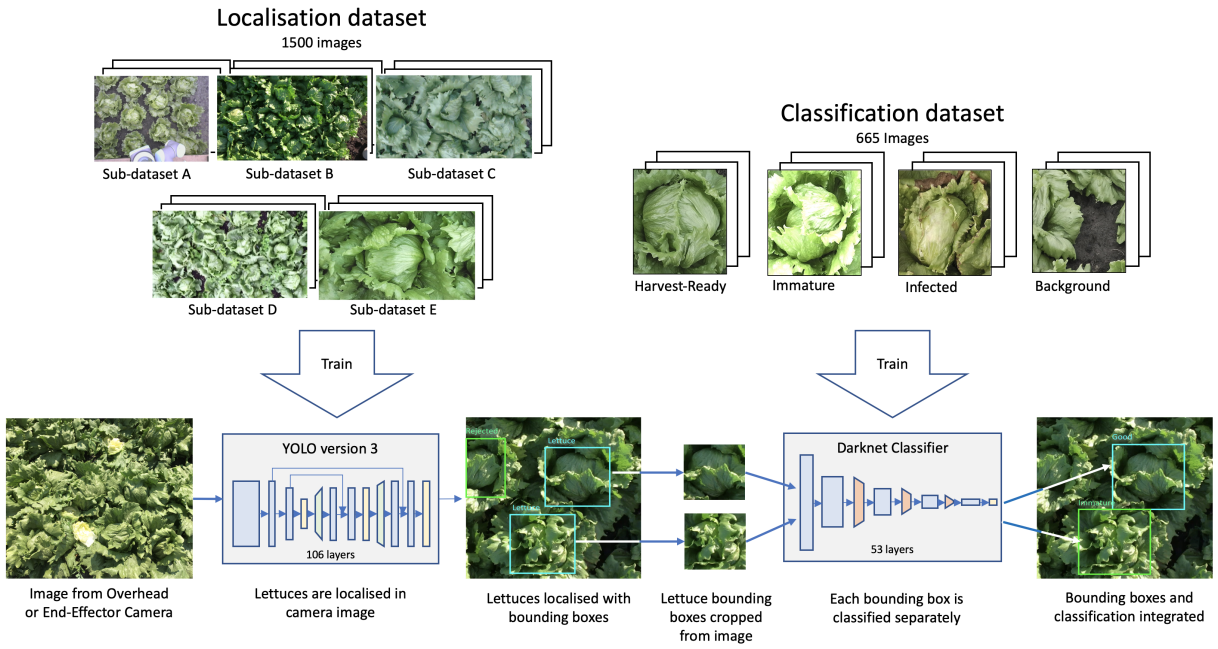


Figure 5: The vision system pipeline showing the two stages of neural network. First, the lettuces are localised using one network. A second network using both the lettuces localised from the first network and pre-segmented lettuce images from a classification dataset is used.

from a dog, but may not be enough to determine whether one of the ten lettuces visible in the overhead camera is infected or not. By first detecting the bounding boxes and then cropping each lettuce from the original 1920x1080 image before resizing to 224 by 224, much more visual information on each lettuce is available for the classification network. This improves the likelihood of a correct classification on images from the overhead camera.

Predictions on the network took 0.082s for localisation in the first stage and 0.013s classification time for each detected lettuce passed to the second stage. Assuming 10 candidate lettuces per image the total time for localisation and classification on the current hardware is approximately 0.212s, slower than a single YOLO object detection network would be, but still sufficiently fast for real-time adjustments. The end effector camera typically has only one lettuce in view during fine tuning, reducing the detection time to 0.095s. The pipeline processes images from both overhead and end effector cameras. The overhead camera provides candidates for picking and the end effector camera is used to fine tune the approach of the end effector to the desired lettuce.

The two-stage network uses the existing datasets to maximum advantage and provides better classification by maintaining a higher resolution on the images of individual lettuces.

4.1.1 Localisation Dataset

Training a deep CNN object detector requires a large amount of data. The dataset also needed to be a good representation of the real scenarios the Vegebot would encounter. Since there was no existing dataset suitable for the propose of this project, a new lettuce localisation dataset was collected, labelled and assembled. Images were collected from three different sources: images taken by the overhead camera on the Vegebot

Figure 6: Obtaining data for the data-set showing the user holding a webcamera to capture data sets at different heights.

Table 2: Details of the different sub datasets used to create the localisation dataset including the number of lettuce and conditions in which the images were taken.

Sub Dataset	Number of Images	Number of Lettuce per Image	Camera Height	Weather Conditions	Image Quality
A	157	7-10	≈1.8m	cloudy/sunny	medium
B	209	8-14	≈ 2m	sunny	high
C	117	3-6	≈1m	cloudy	medium
D	131	4-11	≈1.2	cloudy/rainy	low
E	891	1	≈0.3m	cloudy/sunny/rainy	high

platform, images taken directly with a camera and extracted images from videos taken by mobile phones and webcams. Figure 6 shows the process of obtaining images from the field using a webcam.

Images were divided into 5 sub-datasets (A, B, C, D and E) according to the characteristics of the images and corresponding to the different field experiments in which they were obtained. This allowed better tracking of the dataset to make sure the assembled dataset was well balanced. Figure 5 shows some sample images from each of the five datasets. The images cover different weather conditions, camera heights, lettuce fields, lettuce layouts, lettuce maturity and image qualities, since these are factors that can vary during lettuce harvesting. Table 3 gives a detailed overview for each subset including the number of images, number of lettuces per image, camera heights, weather conditions and image quality. Image quality refers to the subjectively evaluated blurriness of the images.

The images were labelled manually in square bounding boxes using the VoTT Visual Object Tagging Tool (VOTT, 2018). The lettuce images were labelled such that centre of the bounding box is the geometrical centre of the corresponding lettuce and the dimensions of the bounding box are 10% larger than the lettuce head. Only the lettuces whose heads are fully included in the image were labelled. The dataset was randomly separated into training (70%), validation (20%), and test (10%) sets, where the validation set is used for parameter tuning and the test set is only used for benchmarking the final performance.

Even though only lettuces that were fully visible within the image were labelled, the YOLO algorithm was robust enough to detect lettuces at the edges as well. Classifying these partial lettuces would have increased the complexity of the problem unnecessarily. Practically, these lettuces were likely to be out of the reach of the Vegebot robot arm and therefore they were rejected from the detected candidates. There were also cases where lettuces were blocked by weeds, the Vegebot itself or other obstacles, which led to narrow bounding boxes instead of square ones. Lettuce rejection algorithms were implemented to reject such candidates. A

Table 3: Classification dataset, showing the number of each type of lettuce in the dataset.

Lettuce Class	Harvest Ready	Immature	Infected	Background	Total
Number of Images	181	149	121	214	665

candidate was rejected if it met either of the following criteria:

- Rejection of non-square bounding boxes which are on the edges of the images
  $\frac{l}{w} > 1.15$ and $d < margin$ where $margin = \frac{L+W}{75}$
- Rejection of narrow bounding boxes
  $\frac{l}{w} > 1.4$

where w and l are the lengths of the bounding box edges, with w being the longer of the two. L and W are the width and height of the overall image, and d is the distance between the bounding box and the edge of the image.

The localisation network was based on the YOLOv3 architecture and was trained with a batch size of 64, subdivision of 8 and 10,000 iterations. The network was trained on a PC with a 4.5Ghz Intel i7-7700k CPU and an nVidia 1080Ti GeForce GTX GPU. Training took around 12 hours. Pre-trained weights based on ImageNet were used. No data augmentation was applied: this could improve localisation performance and remains for future work.

4.1.2 Classification Dataset

The goal of the classification network is to pick out the harvest-ready (i.e. mature and healthy) lettuces among all the lettuces recognised from the previous localisation step. Immature and infected lettuces should be left in the field. False negative localisation results can be hazardous: reaching for a non-lettuce object can damage the robot (if the object is a rock) as well as the object itself (if the object is a human hand or robot part). Adding a negative 'background' class acted as an additional filter for false positives detection.

The images were labelled by one of the authors with assistance provided by cultivation experts to allow labelling and classification of the dataset. Figure 5 shows sample images from each of the four classes. Table 3 is an overview of the size of the dataset. The 665 images were randomly separated into training (87.5%) and test (12.5%) sets. A higher portion of images were allocated to the training set deliberately due to the limitation of the images available.

The classification network used was the standard Object Classifier supplied with Darknet, with no transfer learning (the use of pre-trained weights would likely increase performance further). The batch size was 64, the subdivision was 4 and the network was trained to 260 iterations. The training was on the same hardware as the localisation network and took 2 hours.

4.1.3 Calibration & End Effector Positioning

The first approach tried on the positioning problem was the classic one of modelling the robot and its coordinate systems, calibrating the camera parameters and then transforming the target centre pixel of the lettuce (the centre of the bounding box) to a position in 3D space and finally using inverse kinematics to

Figure 7: Development of lettuce harvesting end effectors. A - two handed approach with one hand to hold the lettuce, one hand with knife, B - rotary DC motor cutting mechanisms, C linear actuator knife powered mechanism, D - pneumatic cutter chosen as the best mechanism.

move the arm as required. The problem encountered was that the system worked well in the lab, but would fail once subjected to knocks and bumps in the field. Even small deviations in the position of the overhead camera would mean that the robot might incorrectly locate its target by up to 10cm.

A different approach was therefore attempted, where the robot could self-calibrate the transformation from viewport pixels to arm position, using Aruco markers positioned on the top of the end effector. An occasional self-calibration would be sufficient to reset the transformation, for example after moving the platform. Calibration also reset the target location of the lettuce centre within the viewport of the end effector camera. We assume the platform is kept approximately level with reference to the field due to the tracks in which them Vegebot moves. Further details of the final calibration procedure can be found in the Appendix.

4.2 Force Feedback Driven Harvesting

The lettuce harvester has been designed to achieve reliable, efficient harvesting of lettuce with minimal damage to the lettuce. To meet supermarket specifications the lettuce stem should be cut with a single consistent straight cut such that there is approximately 2mm of stem. The outer leaves of the lettuce should also be removed where possible. A UR10 6 degree-of-freedom arm is used to provide movement of a custom end effector which has been specifically designed for lettuce harvesting. The UR10 arm is mounted on a mobile base which can be moved along the rows of lettuce.

The picking sequence (Figure 4 'Pick Sequence') demonstrates how there are two stages to the physical cutting aspect of the harvesting procedure. To minimize the damage to the lettuce and also achieve a clean cut a method where the end effector is made of two mechanisms has been used. Firstly, a soft clamping method is used to hold the lettuce throughout cutting and when lifting. Secondly, a cutting mechanisms is required to cut the stem of the lettuce at a given height. The cutting mechanism requires force (≈20N) to cut through the stem and outer leaves, height adjustability and also straight linear cut.

4.2.1 End Effector Design

To achieve sufficient cutting force to cut the stem, a high impact, straight cut is required at the base of the lettuce. A number of different mechanisms were tested to determine which could achieve sufficient force and quality of cut: soft gripper and knife hand, pneumatic actuation, belt drive and rotary chopping. Figure 7 shows the different mechanisms considered.

The two handed approach lacked sufficient cutting force and required a high level of co-ordination between the two arms. A rotary electric motor approach lacked the force to reliably cut the stem and led to the

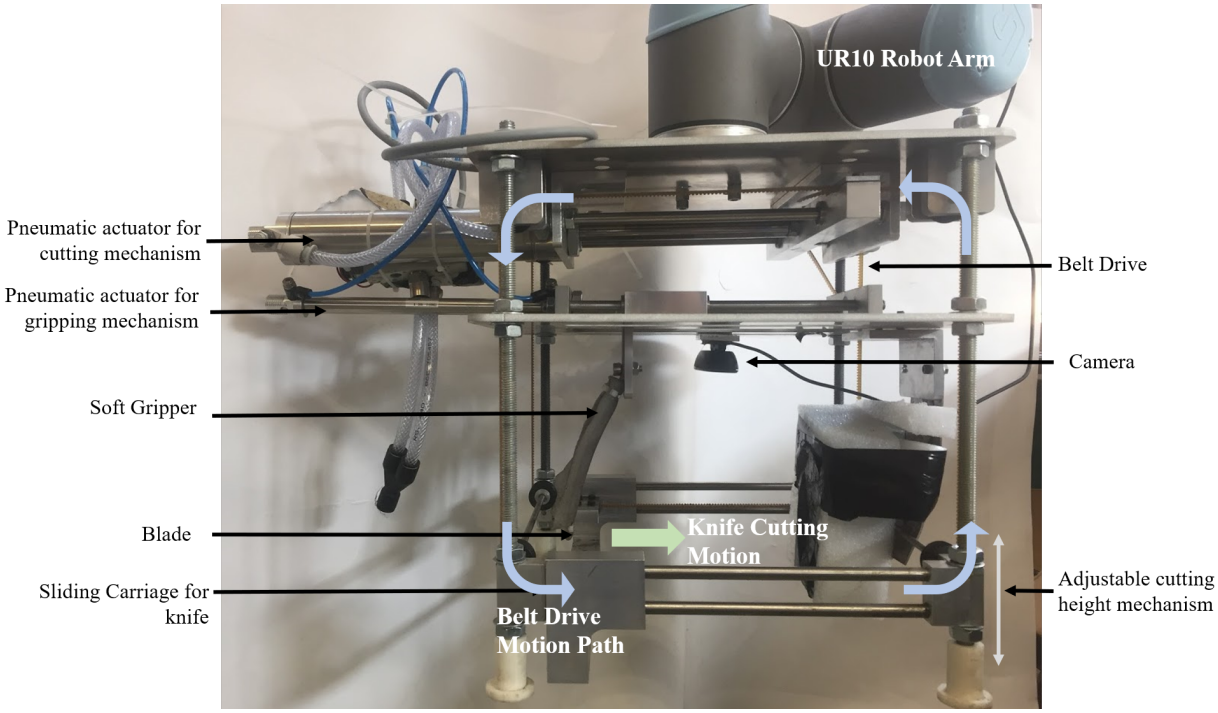


Figure 8: The final end effector developed, showing the belt drive mechanisms and dual pneumatic actuator system.

mechanism having to hack at the stem. The pneumatic cutting mechanics provides a high power-to-weight ratio, making it highly suited for this application where a fast clean cut is required. Although there is no position control, pneumatic actuation allows for easy to implement cut/open control.

The soft gripping mechanism has a single moving gripper and a fixed gripper lined with foam. Similar to other harvesting end effectors (De-An et al., 2011; Foglia and Reina, 2006), a pneumatic actuator is used to control the gripper as this can be used to provide controllable compliance by varying the air pressure such that the lettuce is held but not damaged with simple open/close control

The end effector developed is shown in Figure 8, with the design parameters given in Table 4. The end effector used only two actuators, one for grasping and one for cutting to enable simple control. A timing belt system was used to transfer the linear motion from a single actuator to both sides of the blade to allow smooth movement. This allows the actuator to be mounted above the height of the lettuce, such that when cutting it does not interfere. The belt drive system allows for the height of the cutting mechanism to be easily altered by changing the height of the cutting mechanism.

4.2.2 Force Feedback Control

A key challenge to successful harvesting was reliably cutting the lettuce stalk at the correct height in an environment which is highly varying, uncertain and unknown. To achieve this, the ground was used as a fixed reference point and the stem was assumed to be a fixed distance above the surface. Using force-feedback from the joints of the UR10 robot arm, the end effector is lowered towards the ground, enveloping the lettuce, until a given force was achieved and contact with the ground could be assumed. The cutting height relative to the ground can be adjusted by manually varying the height of the cutting mechanism. A force threshold,

Table 4: Specification of the end-effector developed

End Effector Parameters	**Specification**
Weight	8 kg
Height	45cm
Width	45 cm
Depth	30cm
Gripper Pneumatic Actuator Specification	1 MPa, Bore 10 mm, Stroke 15 cm
Cutter Pneumatic Actuator Specification	1.5MPa, Bore 15mm, Stroke 20cm
Timing Belt	5.08mm pitch, 203cm length, 20mm width
Length of Travel of Blade	200 mm
Cutting Knife Length	250 mm
Inner Area to encapsulate Lettuce	25 cm x 25 cm

T, was found by experimentally determining what force is required for the end effector to interact with the ground, i.e. when it overcomes the resistive force of the leaves and other ground reaction forces, F_R. The force threshold was experimentally determined to be 60N to ensure all leaves were pushed away from the lettuce head and the end effector was in contact and level with the ground. This approach is summarized in Figure 9.

This approach helped push out the outer leaves of the lettuce which interfered with the cutting mechanism. This also allows the end effector to self-level on the ground, and provided stability and consistency. Small 'feet' were added to the end effector to allow stability to be achieved and prevent it from pressing too low into the ground. This approach allows the system to adapt to different field conditions, for example different soil heights relative to the tractor track heights.

Once fully positioned, the lettuce is grasped and the cutting takes place. Each of the pneumatic actuators is controlled by a valve which has two position controls. Two digital outputs from the UR10 end effector are used to control the valves. After the correct height is achieved using force feedback, cutting is triggered by first actuating the grabbing mechanism so the lettuce is held in a fixed place. The cutter pneumatic system is then actuated so the blade cuts the stem of the lettuce. The arm can then be lifted, with the knife released and then the grabber retracted to release the lettuce.

Besides these two challenges, an additional one was that the weight of the end effector was at the limit of the payload ability of the UR10. This restricted the arm to moving more slowly than would otherwise be necessary. This will be discussed in the experimental results.

5 Field Experiment Results

Ten experimental sessions were carried out in the harvesting seasons in 2016-2018 in lettuce fields in Cambridgeshire, UK in varying weather conditions and across many different fields. In these field trips, the system was developed and tested[1]. Field experiments were undertaken to test the performance of the localisation and classification system in isolation from the harvester. The entire system was also integrated to test the full functioning of the system in conjunction with its physical harvesting abilities. In this section the localisation and classification is discussed for both individual and system level tests, after which the harvesting system results are presented.

[1]These were in collaboration with a major agricultural company, G's Growers

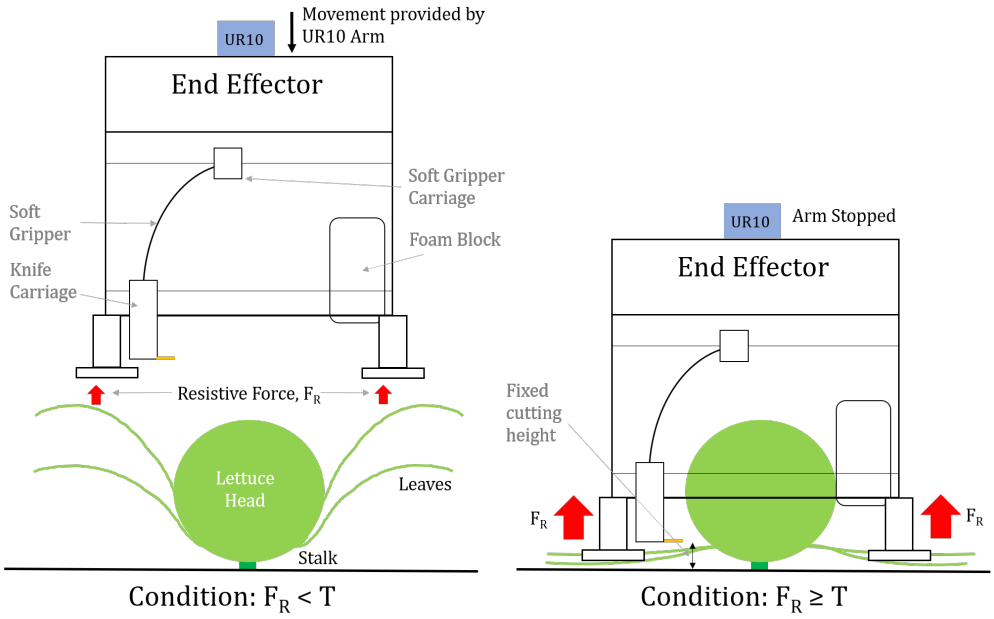


Figure 9: The force feedback method, allowing a repeatable height between the ground and the knife to be achieved.

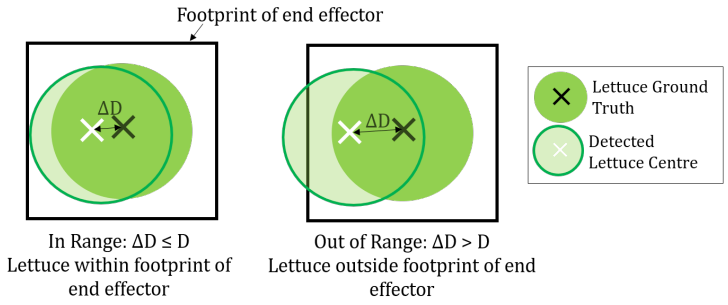
At the beginning of each experimental session, the Vegebot was assembled at the start of a lettuce lane. Typically, a three person crew participated, one operating the control laptop, one observer and one checking and resolving any physical issues and enabling the air compressor when required.

5.1 Localisation

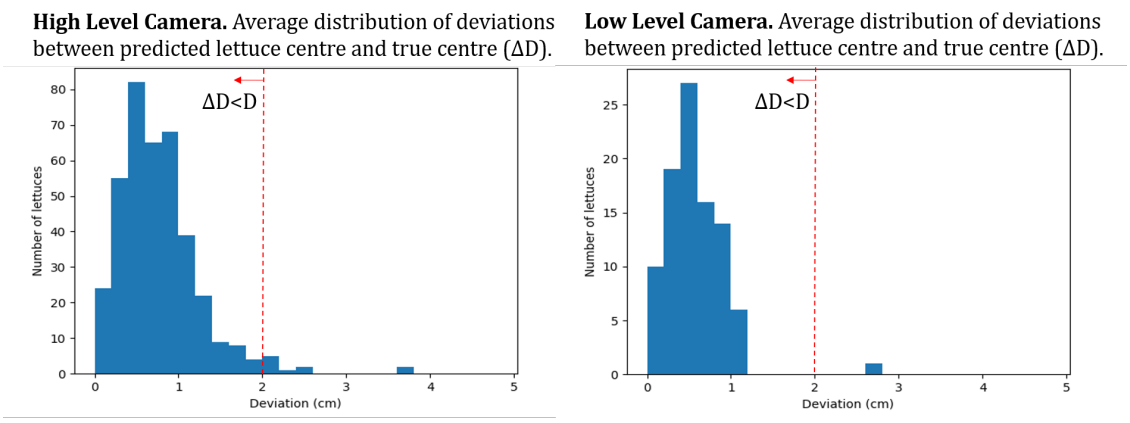
In order for a lettuce to be successfully picked, the centre of the end effector must be placed with a tolerance, D, of the true centre of the lettuce. The tolerance, D, which is determined by the mechanical design of the end effector is approximately 2cm for average sized lettuce. For successful harvesting, the localisation system must predict the centre of the lettuce, such that the absolute difference from the ground truth, ΔD is less than the tolerance ($\Delta D < D$). In practice, for a given camera height the threshold was specified in pixels, calculated taking into account the scale of the image. This threshold is illustrated by Figure 10a.

To test the ability of the system to localise lettuce heads with sufficient accuracy to allow success harvesting, images taken with both low level and high level cameras were used (approximately 20cm and 170cm above the crop respectively). The difference between the detected and ground truth of the lettuce centre was found. The distributions of the accuracy in the localisation performance of the two cameras is shown in Figure 10b.

In the field, the lighting and weather conditions may vary significantly. To test robustness to different lighting conditions, the test subsets of datasets A-E in Figure 5 were artificially modified with image processing (using ImageEnhance Brightness and ImageEnhance. Contrast functions in the Python Willow library) to different levels of brightness and contrast, producing 6 times (7200) the original number of test images (1200). The localisation system was then tested on this set of images (Figure 11). The precision and recall were then found. The system showed a high robustness to changes in image brightness (the most likely changing field conditions), with minimal changes in precision and recall. For the variation in image contrast, although the precision remained high, the recall dropped significantly for high changes in contrast. It is likely that using



(a)



(b)

Figure 10: a) The requirements for successfully lettuce harvesting determined by the physical end effector. The lettuce centre must be detected within a distance such that the lettuce is fully within the footprint of the end effector when cutting. b) The distribution of accuracy of the lettuce localisation system for the two different cameras used, with images from sub-datasets C and E respectively.

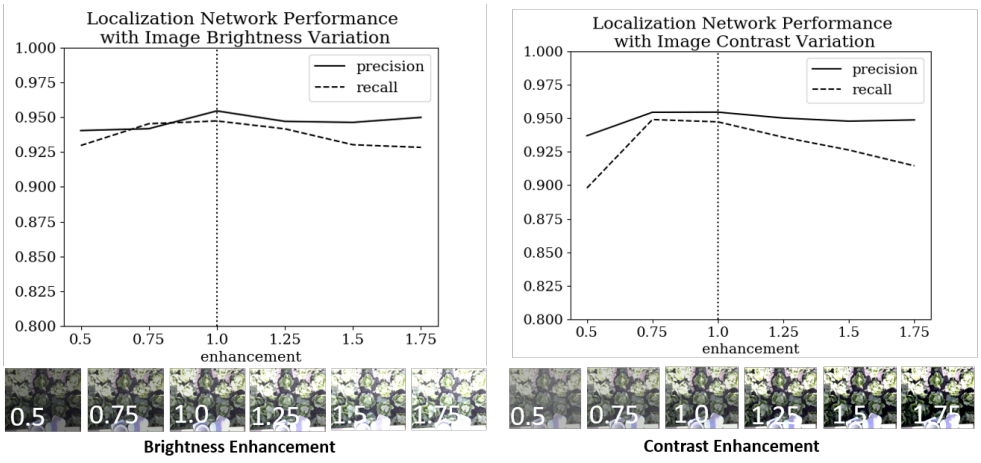
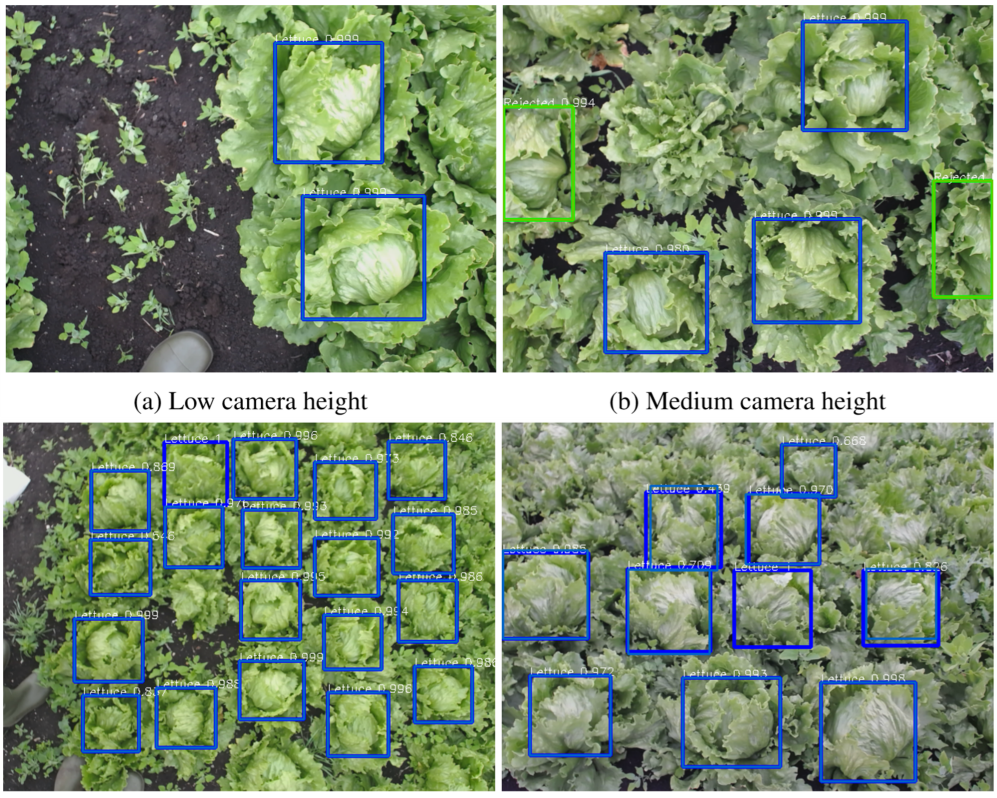


Figure 11: Localisation performance with varying brightness and image contrast. The precision and recall are given in both cases. The images below show the contrast and brightness enhancement added applied to a typical image in the test dataset.



(a) Low camera height

(b) Medium camera height

(c) High camera height

(d) Camera angled by 45 degree

Figure 12: Examples of the localisation system working on different lettuce and with camera setups with different heights and angles and showing usage on different crops and different fields demonstrating robustness. Blue bounding boxes indicate the entire head of lettuce could be seen, green indicate where only part of the head is visible.

data augmentation techniques on the original training dataset would have improved this.

Figure 12 shows some examples of the localisation results. Figure 12a, 12b and 12c show the robustness at different camera heights, different angles and different parts of the field (middle and edges). The system was able to avoid detecting weed (12a and 12c), human feet (12a and 12b) as well as lettuces that fail to form lettuce heads (12b). Figure 12b also shows that the lettuce rejection algorithm is able to effectively reject lettuces which are on the edge of the image. Localisation was also effective at different heights (ranging from 20cm to 170cm) and with the camera tilted by up to 45 degrees.

When integrated into the full system, the overall performance of the localisation system could be tested in harvesting trials. The success rate (number of correctly identified lettuce over total number of lettuce observed) and false positive detections were recorded. The results from this overall system results include over 60 individual lettuce harvesting experiments, where the localisation results of all lettuce that could be visible observed by the system were recorded. The results are shown in Table 5.

Table 5: Overall system harvesting tests showing the localisation performance.

Metric	Result	Definition
Lettuce Localisation Success	91.0%	$\frac{Number\ of\ detected\ qualified}{Number\ of\ real\ qualified}$
False Positive Detection	1.5%	$\frac{Number\ of\ false\ qualified}{Number\ of\ real\ qualified}$

5.2 Classification

Robustness and accuracy of the classification system is critical for avoiding infected or damaged crops which could infect the harvesting system. By skipping immature heads and avoiding unnecessary harvesting the efficiency of the harvester can be maximised. To test the robustness of the system, the same images from the localisation experiments (modified for brightness and contrast) were passed to the classification network and the accuracy recorded. The results are shown in Figure 13a. For classification, the network showed greatest robustness to contrast as opposed to brightness variations; this could be because the training data showed greater variation in contrast opposed to brightness.

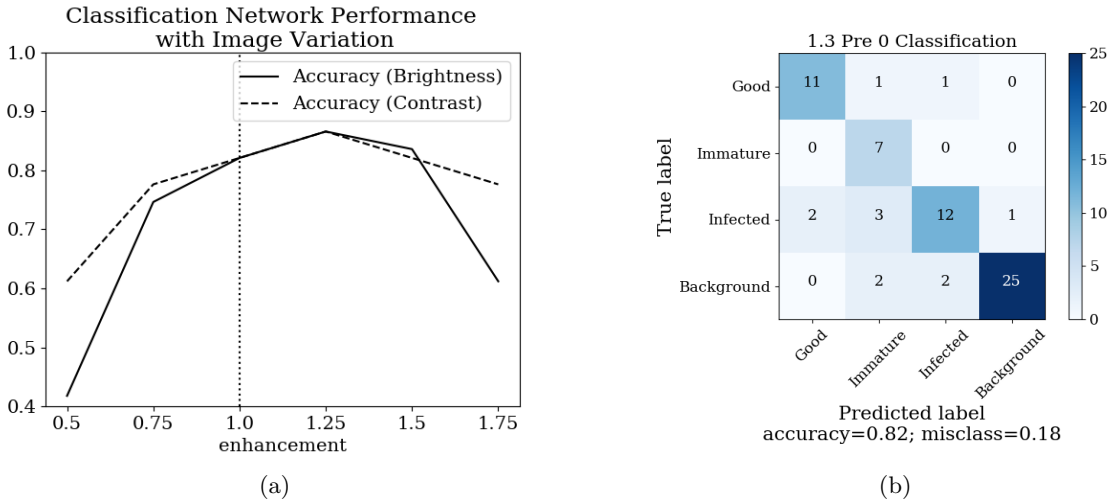
To understand the classification decisions made by the network a confusion matrix of the field tests has been generated and is shown in Figure 13b. The diagonal shows the correctly classified lettuce, showing that the classification performs adequately for identifying background, infected and harvest ready lettuce. Identifying infected lettuce is crucial for avoiding contamination and further work should be undertaken to further improve the classification.

The network struggles to separate harvest ready and immature lettuces. One of the reasons is that the boundary between harvest ready and immature lettuces is very vague and changes accordingly to current market requirements, and thus creating a meaningful dataset is challenging. The classification dataset was labelled under the rules that a 'harvest-ready' lettuce head is around 18cm in diameter, which for the majority of the time is the harvesting requirement. On the day of the field test, there was a change in harvesting specification: lettuces that would normally be treated as 'immature' and left in the field were also harvested, which explains why many of the 'immature' predictions got corrected to 'harvest-ready'.

When entire system tests were ran, the system provide 100% accuracy when classifying lettuce. Although a reasonable number of experiments were ran (69), the number of non-ideal lettuce in this experiment was low, so there was little variation in the classification of lettuce seen.

5.3 Harvesting Performance

The final field tests were performed in May 2018 at a lettuce field in Cambridgeshire, UK. These final tests followed on from over 10 visits to the field with well over 300 lettuce harvested. The Vegebot was positioned at the start of a lettuce lane, the lettuces within the viewport of the overhead camera were detected and picks attempted. Once attempts had been made to pick all feasible lettuces, the platform was moved forward down the lane to the next unpicked rows. Each lettuce position, and false positives or negatives were recorded, together with the number and trajectory of all pick attempts. Finally, each lettuce was inspected for damage, in particular for the stalk being cut too close to the lettuce body. In total, 69 lettuces were detected by the vision system, 60 were in range of the robot arm and harvesting attempted with 31 lettuce harvested



(a) (b)

Figure 13: a) Accuracy of the classification network with changes in image brightness and image contrast. b) The Confusion matrix showing the classification performance of lettuce.

successfully. A video of the Vegebot in operation was recorded [2].

5.3.1 End-Effector Trajectory

During the final field experiments, 69 qualified lettuces were detected by the vision system. Of these, attempts were made to pick 60, the remainder being out of range of the robot arm. 31 pick attempts were successful, with 29 failures, almost entirely due to the weight of the end effector causing mechanical failures on the arm which made attempting harvesting impossible.

The 31 successful trajectories of the end effector are shown in grey in Figure 14, with a representative trajectory highlighted in black. This representative trajectory shows a single experiment which reflects the desired trajectory and demonstrates the different parts of the harvesting process. The breakdown of the time series into the processes from Figure 4 is shown. The X, Y and Z coordinates are shown with respect to the base of robot platform, with X pointing forwards in the direction of travel, Y pointing to the left and Z pointing up.

With the exception of the *Grasp-Cut* section, all of the other trajectory sections were slowed considerably by the burden of the end effector weight on the robot arm. This led to an average cycle time of 31.7s. Critically, the rate limiting step, the grasping and cutting reliably only required 2 seconds. Thus, using a lighter end effector, for example constructing from a lighter material such as carbon fibre, or using a stronger arm could lead to a significantly lower cycle time.

The trajectories clearly show the impact of the force feedback, with the robot arm descending in the Z axis at a consistent rate until the force threshold is met. This shows that the end height of arm varies considerably for different lettuce, showing how using force feedback allows a consistent height to be achieved. There is also slight variability in the X and Y axis close to when the force threshold is reached as the end effector self levels on the ground.

[2] https://youtu.be/UR-7LBdI7Z4

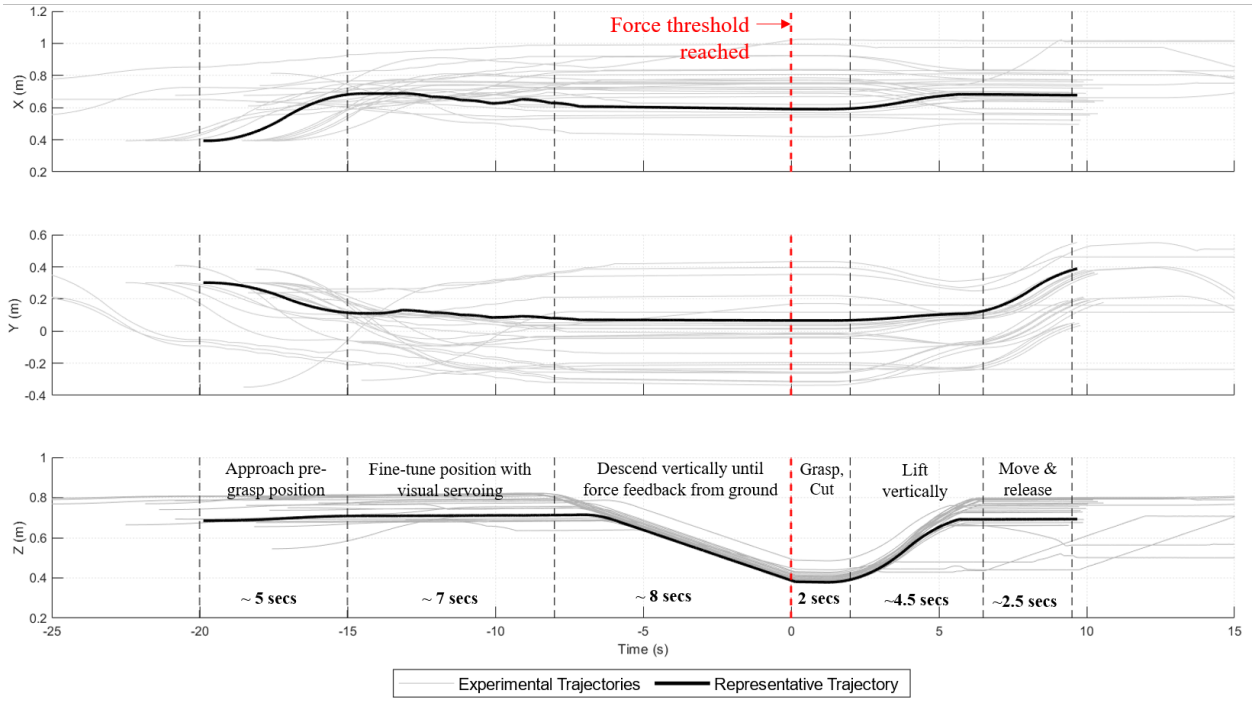


Figure 14: End effector trajectories when undergoing the field experiments. Shows all trajectories centred on cutting (at 0 seconds) and an example representative trajectory. The vertical divisions correspond to the different stages of the Pick Sequence from Fig. 4

.

Table 6: Overall system performance in the harvesting tests. Total lettuces attempted considers only lettuces within restrictions imposed by arm strength.

Metric	Result	Definition
Total Ground Truth Lettuces	69	
Total Lettuces Detected	61 (1 false positive)	
Total Lettuces Attempted	32	
Total Lettuces Detached	31	
Detachment Success	96.9%	$\frac{Number\ of\ successfully\ picked\ qualified}{Number\ of\ detected\ qualified}$
Harvest Success	88.2%	(Lettuce Localisation Success) x (Detachment Success)
Cycle Time	31.7s, $\sigma^2 = 32.6$	Complete cycle time from lettuce to next
Damage Rate	38%	$\frac{Number\ of\ lettuce\ harvested\ in\ unsaleable\ condition}{Total\ number\ harvested}$
Leaves to be Removed	0.75, $\sigma^2 = 1.42$	Average leaves to be removed to achieve scaleability
Total lettuces attempted	69	

5.3.2 Overall Harvesting Performance Metrics

The results of the field experiments are shown in Table 6. Considering all the harvesting attempts, the detachment success if found to be 51.6% (31 out of 60 lettuces correctly identified, excluding false positives). However in 28 cases the harvesting failure was due to practical restrictions (weight of the arm, practical workspace of the robot arm and the range of the overhead camera viewport), such that it was physically not possible to pick some lettuce. If the limitations of the arm are ignored, and the denominator reflects only those lettuces within the practical workspace, then the Detachment Success rises to 96.9% (31 out of 32). In other words, with one exception, if the arm could reach the lettuce, the end effector could pick it. Although this is a considerable exception, it could be simply achieved by using a robot arm with increased torque output.

Examples of the harvested lettuce are shown in Figure 15 showing high quality cuts and also showing those with unwanted outer leaves or damage. The distribution of the lettuces which required extra leaves to be removed, extra cutting attempts and the cycle time is shown in Figure 16. The cycle time varies greatly depending on how far the arm needs to travel from lettuce to lettuce, exacerbated by end effector weight slowing the movements. Most commonly, one extra leaf needed to be removed to achieve supermarket perfection. In some cases extra cuts were required. This was often due to the leaves of the lettuce and movement of the lettuce head within the cutting area . Additionally, the cuts were generally a little too close to the body to be acceptable in the current market.

The average Cycle Time was 31.7 seconds, with a variance of 32.6 seconds. Again, this value was largely due to the limitations of the arm and the weight of the end effector. Of the trajectory sections in Figure 14, all but the short Grasp-Cut section (2 seconds) have their speed limited by the arm's payload capacity. A much reduced Cycle Time should be achievable with a stronger arm or lighter end effector. In addition, around a quarter of the cycle time is taken by the Fine Tuning of the end effector position. Any improvements to the accuracy of the overhead camera localisation would further reduce the overall cycle time.

Reducing the Damage Rate (38%) will require further experimentation. Supermarket chains, the largest wholesale lettuce buyers, have strict standards for the length of the cut stalk to improve the vegetable's appearance in packaging. According to these standards, aesthetic rather than relevant to the lettuce's

Figure 15: Examples of harvested lettuce showing some with an ideal cut, unwanted outer leaves and damaged outerleaves.

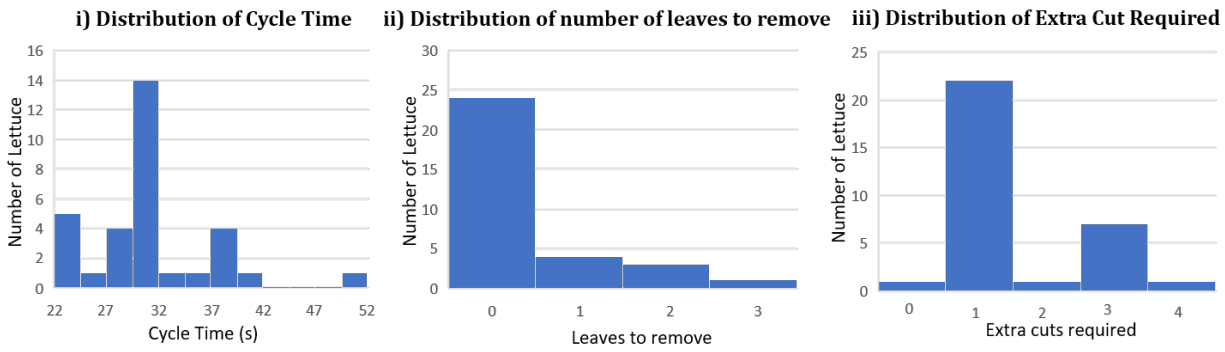


Figure 16: Distribution of the cycle times, leaves to remove and extra cuts required for the various lettuce harvesting experiments.

suitability for eating or not, the end effector often missed the ideal length, cutting in most cases slightly too close to the lettuce head. Of the 32 picks, only 2 actually resulted in inedible lettuces. Improvement can probably be made by refining the force feedback mechanism and perhaps introducing field-dependent depth calibration at the start of each session. This remains for future work.

Again, buyer standards dictate that a packaged lettuce should not have too many superfluous leaves in the packaging. At present, a human harvester will deftly remove a few leaves after each pick before passing the lettuce onto the harvesting rig. The end effector left the picked lettuce with an average of 0.75 additional leaves that are undesirable by these standards. These would have to be removed further down the production chain by hand, or in an automated fashion.

It is worth noting that both the metrics for Damage Rate and Leaves to Be Removed could be substantially improved by permitting a greater range of appearance of the vegetable on supermarket shelves. Until the robot improves, this suggests a dual pricing strategy, with a higher price paid by the consumer for a 'perfect' hand-picked lettuce and a lower price for a more variable but quite edible robot-picked one.

6 Discussion

There is much remaining work required to achieve a iceberg lettuce harvester for commercial operation. Existing challenges include visual analysis, precise manipulator control, harvesting rig development, and reduction of the overall cycle time and costs. In this work the focus was not to develop a commercial product, but to demonstrate proof-of-concept experiments which provide research outcomes which can aid future development of agricultural robotics systems not only for iceberg lettuce, but many other crops. This section discusses the design rationale behind the development process and in particular the visual processing strategies which were chosen and how these approaches can be used to aid future work in this field.

The final prototype of Vegebot is a result of more than 15 iterations and on-site field tests which were carried out in the UK harvest seasons (July-Sept) between 2016-18, and also countless lab based experiments. In each iteration, new software and hardware redesigns were tested in the field, data gathered and results compared. The development approach adopted was to produce a modular system to enable rapid integration and testing of the architecture systematically. Frequent field tests were used to provided feedback and to identifying the improvements required. As a consequence of this approach, the physical design changed radically from week to week (see Figure 7). This process was kept grounded by the use of standard harvesting metrics (Bac et al., 2014) to monitor progress. The authors believe that this iterative approach is more likely to yield robust, field-worthy robots than careful upfront design based on an idealized version of the problem.

As an example of the approach taken, the available visual datasets of lettuces were not ideally suited for an optimal vision system. Two separate datasets, one for localisation and one for classification, were both of reasonable quality in themselves but in an ideal world would have been combined into one integrated whole. Rather than spend time and resources gathering yet another dataset to replace them, the Vegebots neural networks were quickly adapted to make use of what was available. This enabled the robot to detect lettuces correctly, solving the problem for the time being and allowing work on the overall system to continue. With future iterations and online data-gathering this architecture could be simplified once again into a single, tighter CNN architecture.

It is noteworthy that a vision system based on a standard convolutional neural network architecture was able to achieve the localisation results that it did, given the difficulty of the task for a human harvester. Many of the previous harvesting robots detailed in Section 2 required vision systems carefully tailored to the fruit or vegetable in question (eg. detecting colour or depth). For example, broccoli heads are detected using

an elaborate pipeline of RGB-D sensors, point clouds and feature extraction in Kusumam et al. (2016) and radicchios using hand-crafted features and particle filters in Foglia and Reina (2006). CNNs, together with some rapid and informal data gathering, proved 'good enough' for the non-trivial localisation of iceberg and may turn out to be sufficient for other crops.

Considering the mechanical development, by making field testing central to the project, the robot design naturally adapted itself to real-world commercial conditions. Vegebot operates in the same fields and along the same lane layout as human harvesters. Neither the environment nor the crop itself was altered in any way to facilitate the automated harvesting. By contrast, solutions using water knives (Simon (2017)) require careful selection of the crop variety and modifications to the way they are planted. Vegebot-derived solutions could be gradually deployed alongside existing methods, rather than requiring major changes to existing practices. The control and calibration software was repeatedly simplified to provide a solution that worked robustly in the field. Sensors were stripped out, not added. Complex algorithms to model in 3D and determine the optimal cutting position were replaced with mechanical legs that provided force feedback from the ground, giving the robot a simple signal on when to cut. A design change was considered an improvement whenever a mechanical feature or software module was eliminated. In the long-term, this preference for simplicity over sophisticated solutions may prove limiting, yet Vegebot has already achieved important results. The use of standard metrics as proposed by (Bac et al., 2014) kept the project on track and focused on steady, incremental improvements. The authors feeling is that the iterative, simple approach can yield yet many more dividends before being exhausted.

As the project stands, the *damage rate*, caused by cutting the lettuce stem too short, is too high for supermarket standards, although the harvested vegetables are perfectly edible. The most recent sample size of 69 lettuces was enough to confirm this as the next problem to address (hundreds of lettuces had been harvested over previous iterations). Future versions of Vegebot will need to address and improve the damage rate, perhaps with visual feedback from the harvested lettuces dynamically adjusting the force threshold at which the cut is made. In parallel, the end effector needs to be made lighter to achieve a human-level *cycle time*, possibly by manufacturing with carbon fibre, or by using an alternative, stronger cartesian arm design.

In summary, the adaptation of neural networks to pre-existing datasets and the use of simple, low-sensory, environmental feedback may prove useful in other harvesting projects. The authors key recommendation would be rapid iteration with radically different hardware designs, testing in the field as often as possible and relentlessly simplifying and using the standard metrics to stay on track.

7 Conclusions

This paper presented a proof of concept platform called Vegebot that demonstrated an automated and potentially autonomous approach to harvesting lettuces. The vision system, mechanics and control strategy were described and the experimental results detailed.

The goals of the project were to achieve a robust localisation and classification, to achieve a cycle time comparable to humans and to avoid damage to harvested lettuces. The localisation and classification were reasonably robust, as demonstrated by a localisation success of 91% and a classification accuracy of 0.82 when tested on a significant test-data set. The average cycle time on Vegebot (31.7s) was restricted by the weight of the end effector and thus currently slower than humans, but could be easily improved in subsequent versions made from lighter materials. Although the harvest success rate was high (88.2%) the damage rate was poor (38%). The sample size of 60 lettuce demonstrates potential and identifies that future work is required to reduce the damage rate. Further optimization is required to meet supermarket standards.

In comparison to other work in this research ecosystem we have demonstrated a number of new approaches and techniques for agricultural robotics. In using a 2-stage CNN we have used an 'out-of-the box' vision and learning system for a specific agricultural problem as opposed to creating a bespoke system for this particular problem. This is different from many state of the art solutions Berenstein et al. (2010); Ren et al. (2015). We have also explored how this approach can make best use of the available data-sets and can implement end to end data collection, training and testing. Additionally, in the development of the mechanical components of the harvesting system we have shown how the environmental constraints can be exploited. This has been shown to help achieve a consistent cutting height. This use of the environment, and designing mechanical systems to work within an existing agricultural environment is different to many other approaches. This presents an approach to achieve robustness in challenging agricultural environments.

While the immediate future would appear to be robot arms attached to harvesting rigs, an autonomous Vegebot is also a distinct possibility. While its capacity would clearly be more limited, it would have agility in the sense of responding quickly to sudden spikes in demand. Marshalling a human team and a harvesting rig can be difficult at short notice and may be overkill for unexpected but smaller orders, whereas an autonomous Vegebot could be conveniently sent into the field to fulfill them. Outside of harvesting time, it could also be used for data gathering. The vision and learning system in combination with the end effector system provides the potential for individual plant harvesting. This could increase crop and harvesting efficiency.

Agriculture is an industry where margins are low; cost efficiency and time efficiency is key. To make the approach presented viable, the cycle time would need to be reduce to that comparable to humans. However, using a robotic system would enable certain advantages such as a more flexible work force and night-time operation. The techniques and approaches here have been applied to iceberg lettuce, however, the concepts could be applied to other harvesting and robotic agriculture situations. Further work to investigate wider applicability, and developing a more universal harvesting system would increase both commercial and research impact.

Acknowledgments

This project was possible thanks to EPSRC Grant EP/L015889/1, the Royal Society ERA Foundation Translation Award (TA160113) and also the support and valuable time input from G's Growers. In particular, we are extremely grateful to Charlie Kisby, John Currah, James Green and Jacob Kirwan from G's Growers for their support and assistance. We would also like to thank Dr. Alex Jones from the Sainsburys Laboratory and many who have contributed to the iterations of Vegebot: Andre Rosendo, Fabio Giardina, Claudio Ravasio and Vivian Wong.

References

Bac, C. W., Hemming, J., van Tuijl, B., Barth, R., Wais, E., and van Henten, E. J. (2017). Performance evaluation of a harvesting robot for sweet pepper. *Journal of Field Robotics*, 34(6):1123–1139.

Bac, C. W., van Henten, E. J., Hemming, J., and Edan, Y. (2014). Harvesting robots for high-value crops: State-of-the-art review and challenges ahead. *Journal of Field Robotics*, 31(6):888–911.

Berenstein, R., Shahar, O. B., Shapiro, A., and Edan, Y. (2010). Grape clusters and foliage detection algorithms for autonomous selective vineyard sprayer. *Intelligent Service Robotics*, 3(4):233–243.

Botterill, T., Paulin, S., Green, R., Williams, S., Lin, J., Saxton, V., Mills, S., Chen, X., and Corbett-Davies, S. (2017). A robot system for pruning grape vines. *Journal of Field Robotics*, 34(6):1100–1122.

Cubero, S., Alegre, S., Aleixos, N., and Blasco, J. (2015). Computer vision system for individual fruit in-

spection during harvesting on mobile platforms. In *Precision agriculture'15*, pages 3412–3419. Wageningen Academic Publishers.

De-An, Z., Jidong, L., Wei, J., Ying, Z., and Yu, C. (2011). Design and control of an apple harvesting robot. *Biosystems engineering*, 110(2):112–122.

Edan, Y., Han, S., and Kondo, N. (2009). Automation in agriculture. In *Springer handbook of automation*, pages 1095–1128. Springer.

Evert, van, F., Samsom, J., Polder, G., Vijn, M., Dooren, van, H., Lamaker, E., Heijden, van der, G., Kempenaar, C., Zalm, van der, A., and Lotz, L. (2011). A robot to detect and control broad-leaved dock (rumex obtusifolius l.) in grassland. *Journal of Field Robotics*, 28(2):264–277.

Foglia, M. M. and Reina, G. (2006). Agricultural robot for radicchio harvesting. *Journal of Field Robotics*, 23(6-7):363–377.

Hajjaj, S. S. H. and Sahari, K. S. M. (2016). Review of agriculture robotics: Practicality and feasibility. In *Robotics and Intelligent Sensors (IRIS), 2016 IEEE International Symposium on*, pages 194–198. IEEE.

Harrell, R., Adsit, P. D., Munilla, R., and Slaughter, D. (1990). Robotic picking of citrus. *Robotica*, 8(4):269–278.

Hayashi, S., Shigematsu, K., Yamamoto, S., Kobayashi, K., Kohno, Y., Kamata, J., and Kurita, M. (2010). Evaluation of a strawberry-harvesting robot in a field test. *Biosystems engineering*, 105(2):160–171.

Hughes, J., Scimeca, L., Ifrim, I., Maiolino, P., and Iida, F. (2018). Achieving robotically peeled lettuce. *IEEE Robotics and Automation Letters*.

Hui, J. and Hui, J. (2018). Object detection: speed and accuracy comparison (faster r-cnn, r-fcn, ssd, fpn, retinanet and yolov3).

Kemp, C. C., Edsinger, A., and Torres-Jara, E. (2007). Challenges for robot manipulation in human environments [grand challenges of robotics]. *IEEE Robotics & Automation Magazine*, 14(1):20–29.

Kiani, S., Azimifar, Z., and Kamgar, S. (2010). Wavelet-based crop detection and classification. In *Electrical Engineering (ICEE), 2010 18th Iranian Conference on*, pages 587–591. IEEE.

Kurita, H., Iida, M., Cho, W., and Suguri, M. (2017). Rice autonomous harvesting: Operation framework. *Journal of Field Robotics*, 34(6):1084–1099.

Kusumam, K., Krajnik, T., Pearson, S., Cielniak, G., Duckett, T., et al. (2016). Can you pick a broccoli? 3d-vision based detection and localisation of broccoli heads in the field.

Lottes, P., Hörferlin, M., Sander, S., and Stachniss, C. (2017). Effective vision-based classification for separating sugar beets and weeds for precision farming. *Journal of Field Robotics*, 34(6):1160–1178.

Luo, L., Tang, Y., Zou, X., Wang, C., Zhang, P., and Feng, W. (2016). Robust grape cluster detection in a vineyard by combining the adaboost framework and multiple color components. *Sensors*, 16(12):2098.

Mehta, S. and Burks, T. (2014). Vision-based control of robotic manipulator for citrus harvesting. *Computers and Electronics in Agriculture*, 102:146–158.

Mehta, S. S., MacKunis, W., and Burks, T. F. (2016). Robust visual servo control in the presence of fruit motion for robotic citrus harvesting. *Computers and Electronics in Agriculture*, 123:362–375.

Monta, M., Kondo, N., and Shibano, Y. (1995). Agricultural robot in grape production system. In *Robotics and Automation, 1995. Proceedings., 1995 IEEE International Conference on*, volume 3, pages 2504–2509. IEEE.

Nagrani, A. (2016). Deepfarm: Lettuce image classification using deep learning. undergraduate project, university of cambridge.

Nieuwenhuizen, A., Hofstee, J., and Van Henten, E. (2010). Adaptive detection of volunteer potato plants in sugar beet fields. *Precision Agriculture*, 11(5):433–447.

Oetomo, D., Billingsley, J., and Reid, J. F. (2009). Agricultural robotics. *Journal of Field Robotics*, 26(6-7):501–503.

Ottaway, J. N. (1996). Lettuce harvesting method and apparatus to perform the sames.

Ottaway, J. N. (2009). Method and apparatus for harvesting lettuce.

Rajendra, P., Kondo, N., Ninomiya, K., Kamata, J., Kurita, M., Shiigi, T., Hayashi, S., Yoshida, H., and Kohno, Y. (2008). Machine vision algorithm for robots to harvest strawberries in tabletop culture greenhouses. *Engineering in Agriculture, Environment and Food*, 2(1):24–30.

Reddy, N. V., Reddy, A. V. V., Pranavadithya, S., and Kumar, J. J. (2016). A critical review on agricultural robots. *International Journal of Mechanical Engineering and Technology*, 7(4).

Redmon, J. (2013–2016). Darknet: Open source neural networks in c. http://pjreddie.com/darknet/.

Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.

Scarfe, A. J., Flemmer, R. C., Bakker, H., and Flemmer, C. L. (2009). Development of an autonomous kiwifruit picking robot. In *Autonomous Robots and Agents, 2009. ICARA 2009. 4th International Conference on*, pages 380–384. IEEE.

Shepardson, E. and Pollock, J. (1974). Lettuce harvesting apparatuss.

Silwal, A., Davidson, J. R., Karkee, M., Mo, C., Zhang, Q., and Lewis, K. (2017). Design, integration, and field evaluation of a robotic apple harvester. *Journal of Field Robotics*, 34(6):1140–1159.

Simon, M. (2017). Robots wielding water knives are the future of farming.

Van Henten, E., Van Tuijl, B., Hoogakker, G.-J., Van Der Weerd, M., Hemming, J., Kornet, J., and Bontsema, J. (2006). An autonomous robot for de-leafing cucumber plants grown in a high-wire cultivation system. *Biosystems Engineering*, 94(3):317–323.

Van Henten, E. J., Hemming, J., Van Tuijl, B., Kornet, J., Meuleman, J., Bontsema, J., and Van Os, E. (2002). An autonomous robot for harvesting cucumbers in greenhouses. *Autonomous Robots*, 13(3):241–258.

VOTT, M. (2018). Visual object tagging tool: An electron app for building end to end object detection models from images and videos. https://github.com/Microsoft/VoTT. Online; accessed 19-Aug-2018.

A Software

The software (see Figure 17a) was written on the Kinetic release of Robot Operating System (ROS). Custom ROS modules for Vegebot were written in Python and are bundled as the package vegebot[3]:

- vegebot_commander this node is responsible for receiving user commands from the web-based user interface front-end and either executing them or passing them to the appropriate node.
- lettuce_detect this node encapsulates the code that classifies and localises lettuces from a 2D image. It calls the two deep neural networks running on Darknet.
- lettuce_sampler this node supplies sample 2D lettuce imagery for testing purposes when not in the field.

[3]https://bitbucket.org/robotlux/vegebot/src/master/

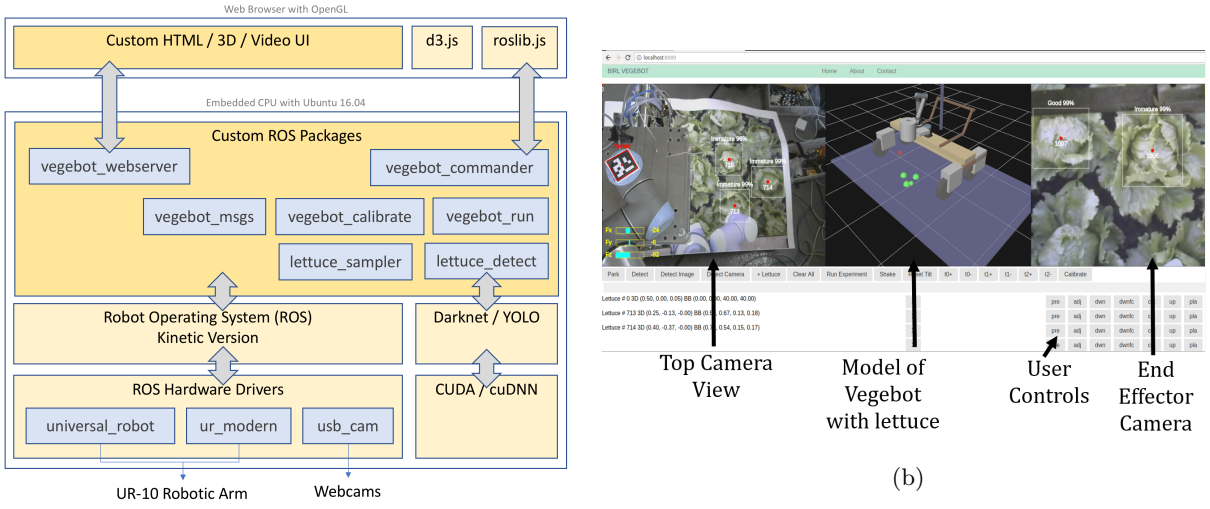


Figure 17: a) The software architecture of Vegebot showing the structure and various packages used. b) The web-based user interface for Vegebot.

- vegebot_msgs this node defines the custom ROS messages used for inter-node communication, including lettuce hypotheses.
- vegebot_webserver this node serves the HTML front-end user interface to the robot operator.
- vegebot_run this module contains the 3D model of the Vegebot (in URDF format) and the scripts for launching the entirety of the software under different conditions.

Standard ROS hardware drivers (universal_robot, ur_modern and usb_cam) are used to drive the UR10 arm and the webcams. A standard installation of Darknet (Redmon, 2016) with YOLOv3 was accelerated by CUDA drivers version 9 to provide image detection services. The HTML user interface (see Figure 17b) can be operated on the same control laptop or remotely, via an on-board WiFi router. The two cameras stream live video to the user interface and bounding boxes and classes for the detected lettuces are overlaid. The position of the calibration marker is also shown. The roslib.js library provides an interactive 3D model of the robot which displays the real robot's movements. The force feedback on the end effector is shown by three bar graphs to the left of the display. Detected lettuces are added dynamically as menu items to the screen, using the d3.js library. The operator can test individual actions (such as 'move to pre-grasp position') or simply select a detected lettuce and instruct Vegebot to pick and place it.

B Calibration Details

The full calibration sequence was as follows and is summarized in Figure 18.

1. Manually position the end effector over any lettuce X using standard UR10 controls.
2. Manually raise the end effector vertically until approximately 10cm clear of the lettuce.

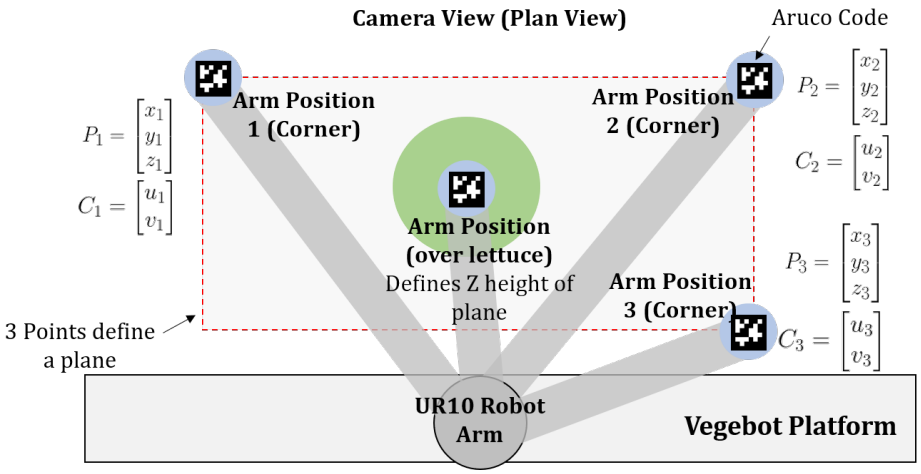


Figure 18: Calibration method, showing how position and camera co-ordinates are gained from 3 positions to allow a mapping from camera to real world co-ordinates to be achieved.

3. Trigger automatic calibration:
   (a) The centre pixel of the bounding box for lettuce X in the end effector camera is recorded as the target centre pixel for fine tuning (the camera is not centred in the end effector for space reasons)
   (b) The calibration records the vertical position of the end effector (Z axis in ROS) and assumes this to be the height of the plane containing all future "pre-grasp" positions.
   (c) The end effector then moves to three positions at the edges of the viewport, in the same horizontal plane. Each position is recorded in terms of the X,Y,Z of the end effector in the robot arm's coordinate frame and in terms of the u,v centre pixel of the detected Aruco marker.

The three calibration positions define a horizontal plane with respect to the ground, around 10cm over the tops of the lettuces. Given any pixel u,v in the viewport, the corresponding x,y,z in the horizontal plane can be found by linear interpolation between these three points. The UR10's built-in inverse kinematics were then used to move the end effector into position in the "Approach pre-grasp position" phase of the Pick Sequence (see Figure 4). For further details of the calculations, see Appendix B.

This rough positioning proved robust enough to move the end effector into the pre-grasp position, but not to exactly centre it accurately over the top of the lettuce. At this point, the end effector "fine tunes" the position using a simple visual servoing method. The bounding box of the target lettuce is now visible in the end effector video feed (see Figure 17b, right hand video feed for an example), the centre point is calculated and then the arm is moved in the horizontal plane (along the X and Y axes) until this centre point coincides roughly with the target pixel recorded in step 3a of the calibration sequence. The end effector is now positioned over the centre of the target lettuce and can then descend vertically.

While the full calibration sequence involves human input to position the end effector over a sample lettuce, the re-sampling of the horizontal plane itself is automatic and could be triggered without human intervention on an as-needed basis, for instance when the 'fine tuning' phase of the trajectory starts to take too long or to fail.

The calibration procedure is always undertaken when the Vegebot is positioned at the start of a lettuce lane. When the platform is manually moved between harvesting sessions, there is a human decision (see Figure 4)

on whether re-calibration is required, if for example the change in terrain has caused the relative position of the platform to the field to change. This can be seen in the increasing amount of time taken to fine tune the end effector position. Long term, this process would be automated.

Three calibration points in robot space (see Fig. 18) are found (P_1, P_2, P_3) and their equivalent viewpoint co-ordinate are found in the camera space (C_1, C_2, C_3). Any viewpoint co-ordinate, C_t (u_t, v_t) can be expressed as the sum of two vectors:

$$\overline{C}_t = a\overline{C}_2 + b\overline{C}_3 \quad where \begin{array}{r} \overline{C}_2 = C_2 - C_1 \\ \overline{C}_3 = C_3 - C_1 \\ \overline{C}_t = C_t - C_1 \end{array} \tag{1}$$

The values of a and b can be found as:

$$b = \frac{\bar{v}_t - \dfrac{\bar{u}_t \bar{v}_2}{\bar{u}_2}}{\bar{v}_3 \dfrac{\bar{u}_3 \bar{v}_2}{\bar{u}_2}} \tag{2}$$

and

$$a = \frac{\bar{v}_3 - b\bar{u}_3}{\bar{u}_2} \tag{3}$$

This allows an equivalent point in robot space to be found:

$$\begin{aligned} \bar{P}_t &= P_t - P_1 \\ &= a\bar{P}_2 + b\bar{P}_3 \end{aligned} \tag{4}$$

Such that the point C_t transformed into robot space can be calculated by:

$$P_t = P_1 + a\bar{P}_2 + b\bar{P}_3 \tag{5}$$