

A Perception Pipeline for Robotic Harvesting of Green Asparagus

Gerard Kennedy* Viorela Ila** Robert Mahony*

* Australian National University, Canberra, Australia (e-mail: *firstname.lastname@anu.edu.au*).

** University of Sydney, Sydney Australia (e-mail: *firstname.lastname@sydney.edu.au*).

Abstract: The global population is expected to pass 9 billion by 2050 requiring ongoing improvements in food production methods. Robotic harvesting offers part of the solution to this challenge and has spurred research in the use of agricultural robots (AgBots) for harvesting horticultural crops over the past several decades. While there has been significant progress in automation of harvest of many crops, robotic systems for crops that require selective harvesting remain far from mature. This paper presents the first steps toward a perception pipeline for a selective green asparagus harvesting robot. We show that a novel single-view representation of information from a multi-camera system can be combined with simple temporal filtering to reliably localise asparagus spears in real-time in lab experiments and difficult outdoor conditions.

Keywords: robotics, vision, automation, harvesting, agriculture

1. INTRODUCTION

An increasing global population and a changing climate are among several factors that are expected to apply pressure to the global food chain in coming decades. The United Nations has predicted that the global population will increase to over 9 billion by 2050, and that food production will increase by 70% in response (FAO (2009)). Furthermore, it has been observed that current agricultural practices have limited scalability due to factors including the lack of arable land, water shortages, under-investment, environmental considerations, and lack of available labour caused in part by an aging global population (ACRV (2018); Duckett et al. (2018)). In short, we face a challenge to produce more with less resources.

Agricultural robots (AgBots) have been identified as part of the solution to this impending global crisis. AgBots are expected to fit into a broader picture wherein crop breeding, planting, and harvesting practices are refined to allow greater levels of automation. Increased levels of robotics in agriculture is predicted to improve efficiency throughout the food production chain.

While mechanisation of broad field farming is highly developed and robotic automation of these processes is progressing as expected, progress in selective harvesting (only harvest a crop if criteria are met) robotic system has been slow. There are still relatively few examples of successful prototypes and almost no commercial examples of selective

harvesting AgBots. The key reason for this is that most applications of selective harvesting AgBots require advanced sensor data processing to identify target crops for harvest amongst the unripe crops and surrounding plant matter. Achieving reliable identification of crops remains a challenge in difficult on-farm conditions where speed and accuracy are paramount and the environment is highly variable (ACRV (2018)). In order for robotic technology to be successfully integrated into existing farm operations and logistics they need to be able to operate at speeds and efficiencies comparable or superior to human labour. High speed harvest requires real-time (in the order of tens of milliseconds) perception and cognition as well as high speed actuators and robust mechanical designs.

In this paper we present a prototype perception pipeline for a green asparagus harvesting robot. Green asparagus grows irregularly in beds with the harvest period covering eight to ten weeks in Australia. Spears grow straight out of the ground with no obscuring foliage. The asparagus spears are harvested when they reach between 20 to 25 cm tall with taller asparagus to be removed and discarded. Since asparagus spears can grow up to 5cm per day, each row must be harvested each day during the peak growing period, and only those asparagus in the correct height range should be harvested. No commercial automated mechanical systems for asparagus harvesting exist and more than half the cost of producing asparagus is associated with the labour costs.

The proposed perception pipeline has been optimised for speed and accuracy through the use of multiple monocular cameras, a distributed compute architecture, and simple algorithms. We expect this approach to allow our harvester to reach an optional balance of speed and accuracy. The perception system is targeted to provide accurate information to a mechanical harvesting system (commercial in

* G.Kennedy and R. Mahony, are with the Australian Centre for Robotic Vision, Australian National University, Australia. V. Ila is with the Australian Centre for Field Robotics, Sydney University, Australia. This research was supported in part by the Australian Government Research Training Program Scholarship and in part by the Australian Research Council through the “Australian Centre of Excellence for Robotic Vision” CE140100016.

confidence) capable of a cycle time of 300ms per asparagus spear. The target is for the robotic harvest system to travel at 1-2 m/s, roughly an order of magnitude faster than existing systems and comparable or superior to human labour. Asparagus spears will be visible to the perception system for 500-1000ms depending on the configuration of cameras. Results are presented that demonstrate the performance of the pipeline to varying crop layout in-lab, and difficult, on-farm conditions. The key contributions of this paper are:

- Algorithm architecture capable of calibrating a multi-camera system, and detecting and localising asparagus spears in outdoor conditions
- A novel single-image representation of a scene captured by a multi-camera system
- On-farm and in-lab trials of the perception pipeline

The remainder of this paper is structured as follows: Section 2 provides a brief history of robotic harvesters, with particular focus on robotic harvesters for green asparagus. Section 3 breaks down our proposed algorithm architecture. On-farm and in-lab experimental results are presented in Section 4, before concluding remarks in Section 5.

2. RELATED WORK

The first example of considering computer vision to crop detection involved utilising light reflectivity differences to detect citrus fruit (C Schertz (1968)). The first applied system was developed in 1977 for apple detection, using a black and white camera with red optical filter along with intensity thresholding and morphological operations (A. Parrish et al. (1977)). Early applications of computer vision to crop detection utilised mainly spectral, intensity or range cameras, and pixel-wise or shape-based crop detection methods (Jimnez et al. (2000)).

Robotic harvesters are currently being developed for a range of crops that require selective harvesting, with oranges, tomatoes, apples, and asparagus being among the most frequently targeted (Bac et al. (2014)). Recent robotic harvesters often employ sensing systems including single RGB (Lehnert et al. (2017)) or RGBD (Leu et al. (2017)) camera systems, combined with 3D reconstruction and 2D or 3D segmentation algorithms. A survey of robotic harvesters conducted in 2013 of 50 distinct projects over the previous 30 years found that average harvesting success rate was 66%, and average cycle time between harvests was 33 seconds (Bac et al. (2014)).

Previous asparagus harvesting robots have typically employed either LIDAR or single camera sensors. These sensing modalities have led to perception systems that only provide information about asparagus height (LIDAR) or else rely on computationally expensive algorithms such as dense 3D reconstruction. The first green asparagus harvester, CAMIA, was developed in the 1990's using laser sensors to detect asparagus height. While CAMIA had promising initial results, unreliability of the sensors caused many asparagus to be missed and damage caused to the bed (Arndt et al. (1997)). Similar to CAMIA, the Kim Haws harvester (Haws (2019)) and Geiger-Lund harvester (Geiger-Lund (2019)) currently being developed also em-

ploy laser sensors to detect crop height, and have also been noted to miss a significant proportion of ripe asparagus (Leu et al. (2017)). MANTIS (K. Carpenter (2019)), utilises a LIDAR sensor suite and can detect asparagus at a rate of 1 every 6 seconds (K. Carpenter (2019)). Finally, the GARotics harvester employs an RGBD camera and 3D reconstruction to harvest asparagus and reported a 90% harvest success rate of detected spears, although their detection success rate is not included (Leu et al. (2017)). At the time of writing, there are no examples of commercially successful selective harvesting robots for green asparagus. We believe that key reasons for this include poor detection success rates and slow harvest cycles. These insights form the motivation for this work.

3. ALGORITHM ARCHITECTURE

A system of three PointGrey cameras is used to capture images of asparagus at a frame rate of $\sim 55\text{Hz}$. To keep lighting conditions consistent in the field our perception system will be housed within a darkened enclosure with its own internal lighting. This will enable day-time and night-time harvesting. The algorithm architecture is split into two components that operate in parallel. These components are titled Extrinsic Camera Calibration, and Perception Pipeline, and are discussed in Sections 3.1 and 3.2 respectively.

3.1 Extrinsic Camera Calibration

Extrinsic camera calibration is computed online as a separate process from the perception pipeline. It is computed at a low frequency and used to update camera pose and ground plane estimation periodically. This ensures that the cameras remain well calibrated throughout long periods of on farm use even in rough conditions. Extrinsic calibration involves computing a camera pose matrix $T \in \mathbf{SE}(3)$ for each camera. The calibration is implemented in the open-source OpenMVG framework via a Structure from Motion (SfM) algorithm computed from sparse feature points that are matched between frames (Moulon et al. (2017))¹.

Given that the vast majority of matched feature points occur on the ground plane, this plane can be found via simple 3D plane fitting. The equation for the ground plane is defined as

$$ax + by + c = z$$

for 3D points $(x, y, z) \in \mathbb{R}^3$. The constants a, b, c are solved for using

$$\begin{bmatrix} x_0 & y_0 & 1 \\ x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} z_0 \\ z_1 \\ \vdots \\ z_n \end{bmatrix},$$

which can be represented as an over determined linear system $Ax = B$ and solved via the left pseudo inverse

$$x = (A^T A)^{-1} A^T B.$$

This ground plane provides a convenient 2D representation of the region of interest. The output of the extrinsic calibration is a matrix T representing the pose of each camera, and a ground plane G , all with respect to a common inertial frame.

¹ Code available at: https://github.com/kennege/openMVG/tree/develop_multi_camera_calibration

3.2 Perception Pipeline

The perception pipeline incorporates the software for receiving images continuously from the multi-camera system, and information from the extrinsic calibration, and returning locations of asparagus to be harvested. Broadly, the procedures involved in the pipeline for each time step are: generate a 2D probability map for each image, project these maps to a common ground plane computed from the extrinsic calibration, and detect potential asparagus ‘hypothesis’ on this ground plane. Finally, the hypothesis are filtered temporally across time steps to improve accuracy. The four **procedures** are discussed further in the following four subsections respectively.

Algorithm 1 Perception Pipeline

```

1: Input images from time  $t = 0 : \text{end}$ 
2: Initialise Kalman Filter
3: for ( $t = 0 : \text{end}$ ) do // for images each time step
4:   for ( $\mathbf{I}_{i=0:2}$ ) do // for images this timestep
5:     procedure 1. 2D PROBABILITY MAP( $\mathbf{I}_i$ )
6:        $\mathbf{I}_i = \text{yCBCR}(\mathbf{I}_i)$  // colour space transform
7:        $\mathbf{I}_i = \gamma(\mathbf{I}_i)$  // perform Gamma correction
8:        $\mathbf{P}_{(0,1,2),i} = \text{PCA}(\mathbf{I}_i)$ 
9:     procedure 2. GROUND PROJECTION( $\mathbf{p}_{1,i}$ )
10:       $\mathbf{P}_i = \mathbf{K}_i \mathbf{T}_G \mathbf{T}_i^{-1}$  // projection matrix
11:       $\mathbf{H}_i = \mathbf{P}_i[:, [0,1,3]]$ 
12:       $\mathbf{G}_i = \text{warp}(\mathbf{p}_{1,i}, \mathbf{H}_i)$  // backward projection
13:   procedure 3. FIND HYPOTHESIS( $\mathbf{G}_{i=0:2}$ )
14:      $\mathbf{G} = \sum_{i=0}^2 \mathbf{G}_i$  // combine images
15:      $\mathbf{G} = \text{thresh}(\mathbf{G})$ 
16:      $\mathbf{G} = \text{removeClutter}(\mathbf{G})$ 
17:      $\mathbf{L}_{1:n} = \text{fitLines}(\mathbf{G})$  // fit lines 0:n
18:      $\mu_{1:m}, \Sigma_{1:m} = \text{hypothesis}(\mathbf{L}_{i=1:n}, \mathbf{L}_{j=1:n}, \mathbf{L}_{k=1:n})$ 
19:   procedure 4. FILTER HYPOTHESIS( $\mu_{1:m}, \Sigma_{1:m}$ )
20:      $\mu_{1:m}, \Sigma_{1:m} = \text{KalmanFilter}(\mu_{1:m}, \Sigma_{1:m})$ 
21:     for  $h=0:m$  do // for each hypothesis  $h$ 
22:       if ( $m_h > \text{thresh} \ \& \ 20\text{cm} < l_h < 25\text{cm}$ ) then
23:         Mark  $h$  for harvesting

```

2D Probability Map: For each timestep images from each camera are first processed to produce a representation of probability of asparagus. Our approach involves colour space conversion from RGB to yCBCR, gamma correction, and Principle Component Analysis (PCA). The yCBCR colour space has been chosen empirically based on the relative contrast of plant matter to background. Gamma correction was computed with $\gamma = 1.5$. The first three principal axis are generated, and the second axis is used as the output probability map.

Ground Plane Projection: The known camera poses and ground plane provided by the extrinsic calibration are used to project each image plane I to the common ground plane, labeled G (see Figure 2). This is done by first computing the camera projection matrix $P \in \mathbb{R}^{3 \times 4}$ using

$$P = K T_G T^{-1},$$

where K is the intrinsic camera matrix computed offline via the Matlab Camera Calibration app (Matlab (2019)). The camera pose $T \in \mathbf{SE}(3)$ is obtained from the extrinsic calibration, and $T_G \in \mathbf{SE}(3)$ is chosen to transform the

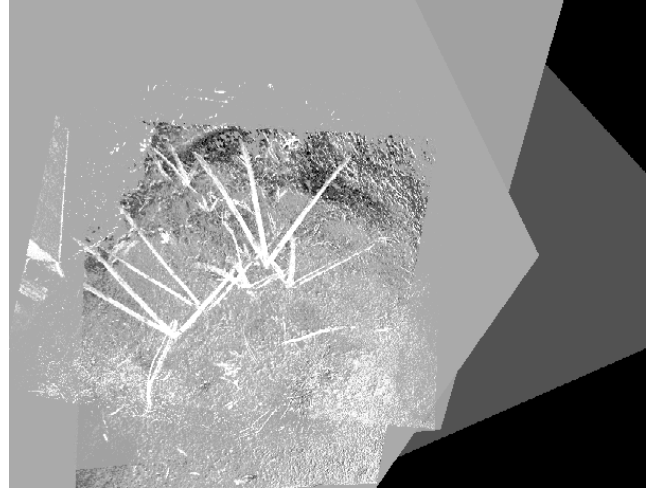


Fig. 1. Ground Plane image obtained by projecting the 2D probability map from each camera view to a common ground plane.

inertial frame of the extrinsic calibration such that it is aligned with the z -axis orthogonal to G . Let $P = [p_1, p_2, p_3, p_4]$ with each $p_i \in \mathbb{R}^3$ representing a column of P . The first three columns p_1, p_2, p_3 represent the vanishing points in G with respect to the x , y , and z axis respectively, in homogeneous image coordinates. As we have defined T_G such that the z -axis of the new inertial frame is orthogonal to the ground plane then $p_3 \approx (0, 0, 1)^T$, where \approx denotes equality up to scale. Set $H = [p_1, p_2, p_4] \in \mathbb{R}^{3 \times 3}$. This is an homography between G and I . Points in each image frame I can thus be mapped to G using a backward homography projection,

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \approx H \begin{bmatrix} x_g \\ y_g \\ 1 \end{bmatrix}$$

for $i \in I$ and $g \in G$.

An example ground plane image is provided in Figure 1. This image is created using the scale 1 pixel = 1mm. In this figure, each asparagus is represented by a ‘chicken-foot’ object corresponding to the same asparagus projected from three different views. This is a novel method of encoding multi-camera information in a single image plane. We believe that the extra information encoded in this image (three views of each object in one image) has the potential to allow machine learning algorithms to extract high quality estimates of object location and pose, compared to typical single image-based algorithms that perform estimates from single RGB images. While the goal is to eventually apply such algorithms for 3D localisation, this step is currently implemented via the algorithm outlined below.

Hypothesis Generation: The ground plane image is thresholded and morphological operations (closing with the OpenCV line structuring element (Bradski (2000))) are applied to remove ‘clutter’. Lines are fit to each remaining morphological ‘blob’ via a least-squares approach. Area A , centroid c , and eccentricity e are also computed for each blob. All line intersections are computed, and a hypothesis of an asparagus 2D location within the ground plane is represented by every set of three intersections.

The hypothesis mean μ_h is computed as the centroid of the triangle spanned by the three intersections,

$$\mu_h = \frac{1}{3}(a + b + c) \quad (1)$$

where $a, b, c \in \mathbb{R}^2$ represent the three intersections of the given hypothesis. The hypothesis covariance Σ_h is represented by the Steiner circumellipse of the triangle (Wolfram (2019)) and the score of each triangle edge, via

$$\Sigma_h = \begin{bmatrix} \frac{e}{3} \sum_{i=0}^2 S_i & 0 \\ 0 & \frac{f}{3} \sum_{i=0}^2 S_i \end{bmatrix} \quad (2)$$

where e and f refer to the major and minor axis of the Steiner circumellipse respectively and S_i refers to a score of triangle side i . We propose a likelihood score S to be

$$S = \left| 1 - \frac{e + \frac{A}{\mu_A}}{2} \right|,$$

where A and e are the area and eccentricity of the current asparagus ‘blob’ respectively and μ_A is found empirically to be the area of a typical asparagus ‘blob’ in pixels. This score is designed to provide a normalised measure of how well the eccentricity and area of the detected blob matches a typical asparagus. A hypothesis is added to the system state if it satisfies the condition

$$\left(\frac{1}{3} \sum_{i=0}^2 S_i < 0.6 \right) \wedge (t < 1) \wedge (\mu_h \in I_b) \wedge (\|\Sigma_h\| < 1),$$

where $t = \left| \frac{\bar{l} - \mu_l}{\sigma_l / \sqrt{3}} \right|$, $\|x\|_2$ is the L2-norm of x , and

$$\bar{l} = \sum_{i=0}^2 2 \|\mu_h - c_i\|_2 \quad (3)$$

is the average length of the three asparagus ‘blobs’ in the hypothesis (in pixels), μ_l and σ_l are found empirically to be the mean and standard deviation in asparagus ‘blob’ length, c is the centroid of the current asparagus ‘blob’, and $x \in I_b$ indicates that x lies within the ground plane image, on an asparagus ‘blob’. This condition was chosen empirically based on a range of tests run on farm and in-lab data. Asparagus hypothesis height in meters ($l \in \mathbb{R}$) can be obtained using Equations (1) and (3) and

$$l = \frac{1}{3} \sum_{i=0}^2 \frac{\frac{l_i}{1000} z_i}{\frac{\sqrt{\mu_h^2}}{1000} + \sqrt{x_i^2 + y_i^2} + \frac{l_i}{1000}} \quad (4)$$

where x_i, y_i, z_i represent the position of camera i with respect to G , and the scale factor of $\frac{1}{1000}$ is used due to the imposed scale of $1\text{mm} = 1$ pixel in the ground plane image.

Using this formulation, the probability encoded in the 2D map is propagated to morphological blobs, then to lines via the line scores S , then to 2D location hypothesis via μ_h and Σ_h . Finally, these hypothesis are filtered temporally with a Kalman Filter.

3.3 Kalman Filtering

A 2D Kalman Filter (KF) is used to track hypothesis temporally through successive ground plane images. Given our expected harvester speed we expect each asparagus

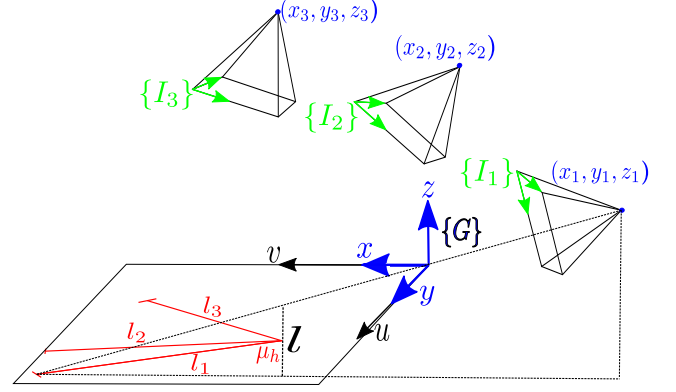


Fig. 2. Relationship between image frames I_i and ground plane G . Where l_i is the 2D length of the asparagus in G projected from image plane I_i , and μ_h is the 2D position estimate and l is the height estimate.

spear to be in view for 30-50 frames. Temporal filtering allows hypothesis to be refined and pruned over this period.

The KF is applied to the following process model:

$$X_k = X_{k-1} + U_k + W_k$$

with observations of the form:

$$Z_k = X_k + V_k$$

where

$$X_k = \begin{pmatrix} x_k \\ y_k \\ l_k \end{pmatrix}, U_k = \begin{pmatrix} u_k \\ v_k \\ 0 \end{pmatrix},$$

$$W_k \sim \mathcal{N}(0, Q_k), V_k \sim \mathcal{N}(0, R_k),$$

$$Q_k = \begin{bmatrix} \Sigma_{u_k} & 0 & 0 \\ 0 & \Sigma_{v_k} & 0 \\ 0 & 0 & 1 \end{bmatrix}, R_k = \begin{bmatrix} \Sigma_{h,k} & 0 \\ 0 & \Sigma_{l,k} \end{bmatrix}$$

and k denotes timestep, x, y represent the 2D location of an hypothesis in ground plane coordinates, with associated measurements provided by μ_h from Equation (1), l represents the height of the asparagus hypothesis obtained in pixels using Equation (3) and converted to meters using Equation (4). The covariance Σ_h is found using Equation (2), and Σ_l represents the covariance of the height estimate. The covariances Q_k and R_k represent process and measurement noise respectively. Data association of asparagus hypothesis between frames is obtained via Lucas-Kanade Optical Flow (LKOF). Estimates for u_k, v_k, Σ_{u_k} and Σ_{v_k} are obtained empirically from the pixel-wise displacement obtained from the LKOF.

An asparagus hypothesis is considered to be valid if it is tracked for a set number of frames (referred to as ‘track length’ and labeled m_h in Algorithm 1). If a hypothesis is validated and its height meets the threshold condition it is marked for cutting. The 3D cutting point is fixed at 2cm above the ground plane.

4. EXPERIMENTS

4.1 On-farm

Image data has been captured at a nearby asparagus farm on two occasions (Figure 3). This has allowed for testing



Fig. 3. On-farm day and night experiments.

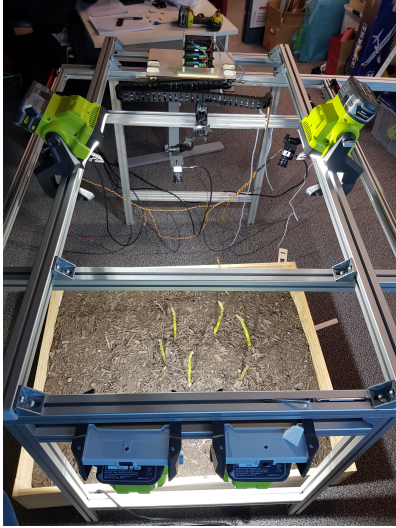


Fig. 4. In-lab rig for harvesting simulation. A sliding bar attached to a motor system allows the camera system to move over an asparagus bed at the expected speed of the harvester.

of the performance of the physical perception system, and robustness of the perception pipeline to outdoor conditions.

Figure 5 presents an example result of hypothesis generation and filtering using on-farm data. While initial hypothesis detections are noisy, the KF reliably filters erroneous detections via an upper limit on the state covariance. Initial small covariance values are due to the size of the Steiner circumellipse. Covariances grow due to process noise, and when an hypothesis is not tracked between frames, and covariances decrease when a data association is made. During harvesting, an hypothesis will be considered valid if it has a certain track length. The on-farm dataset contains 30 frames, and the required track length was set to $m_h = 3$. Using this track length, the three hypothesis shown in Figure 5(h) were validated. At this point the asparagus height estimate will be obtained from the KF and harvested if it is tall enough. The 3 tall spears met the target track length ($m_h = 3$) after 8 frames and were marked for harvesting. The 2 smaller targets and a horizontal spear are not detected, but these would not meet the harvest threshold. No other locations surpassed the target track length for the remainder of the dataset (30 frames).

4.2 In-lab

Datasets simulating the perception system moving over an asparagus bed have been captured in-lab via the

fabricated rig shown in Figure 4. A sliding bar attached to a motor system allows the camera system to move over an asparagus bed at the expected speed of the harvester. Using this rig, the results shown in Table 1 were generated.

Table 1 contains the results of six experiments, each with a different configuration of asparagus spears. Example configurations are shown in Figure 6. A dense 3D reconstruction computed in OpenMVG is used as a pseudo ground truth in this experiment. RMSE (mm) is the root mean squared error in the KF estimate, ξ_k for a given hypothesis, taken with respect to the estimate provided by the 3D reconstruction, ζ_k . That is

$$\text{RMSE} = \sqrt{1/N \sum (\xi_k - \zeta_k)^2}.$$

Exp	RMSE (mm)
1	0.137
2	1.829
3	0.128
4	1.778
5	0.207
6	0.333
Overall	0.735

Table 1. Results of In-lab Experiments. **RMSE (mm)** is the root mean squared error in the KF estimate.

Each dataset in Experiments 1-6 comprises of ~ 30 time steps, and the required track length is once again set to $m_h = 3$. No horizontal spears were detected during the experiments. This is due to the ground plane projection, from which horizontal spears do not produce the ‘chicken-foot’ representation of vertical asparagus. This is advantageous as only upright spears are targeted for harvest. This experiment indicates that the proposed pipeline finds asparagus cutting points robustly with a high level of accuracy.

5. CONCLUSION

In this paper we have presented a perception pipeline for robotic harvesting of green asparagus. The pipeline utilises simple algorithms and a multi-camera rig to obtain estimates of asparagus location in real-world conditions. A novel single-view ground plane projection is presented as a novel approach to condensing multi-camera information into a single image. Results indicate that the pipeline produces accurate location estimates, and performs well in on-farm conditions.

ACKNOWLEDGEMENTS

The authors would like to acknowledge Alex Martin and Dr. Tao Hu for their help in developing the presented hardware and software, and in the undertaking of experiments.

REFERENCES

- A. Parrish, E., J., and K. Goksel, A. (1977). Pictorial pattern recognition applied to fruit harvesting. *Transactions of the ASAE*, 20, 0822–0827. doi:10.13031/2013.35657.

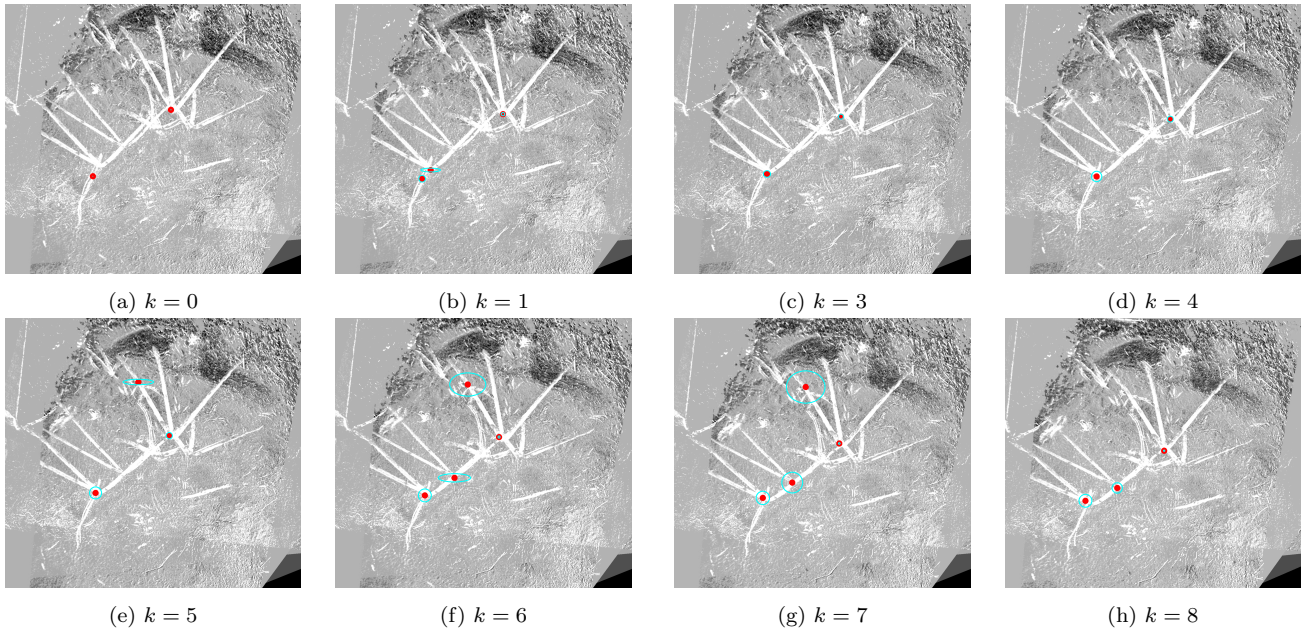


Fig. 5. Kalman filtering of hypothesis using the first 6 frames of on-farm data. The 3 tall spears met the target track length ($m_h = 3$) after 8 frames and were marked for harvesting. The 2 smaller targets are not detected, but these would not meet the harvest threshold. A horizontal spear is also not detected. No other locations surpassed the target track length for the remainder of the dataset (30 frames). Covariance ellipses increase between frames if the hypothesis is not detected, and decrease if the hypothesis is detected.



Fig. 6. In-lab Experiments. Only upright asparagus should be targeted for harvesting.

- ACRV (2018). A robotics roadmap for australia. https://www.roboticvision.org/wp-content/uploads/Robotics-Roadmap_FULL-DOCUMENT.pdf.
- Arndt, G., Rudziejewski, R., and A. Stewart, V. (1997). On the future of automated selective asparagus harvesting technology. *Computers and Electronics in Agriculture - COMPUT ELECTRON AGRIC*, 16, 137–145. doi:10.1016/S0168-1699(96)00033-6.
- 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. doi:10.1002/rob.21525. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21525>.
- Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.
- C Schertz, G.B. (1968). Basic considerations in mechanizing citrus harvesting. *Transactions of the ASAE*, 343–346.
- Duckett, T., Pearson, S., Blackmore, S., and Grieve, B. (2018). Agricultural robotics: The future of robotic agriculture. *CoRR*, abs/1806.06762. URL <http://arxiv.org/abs/1806.06762>.
- FAO (2009). Global agriculture towards 2050. http://www.fao.org/fileadmin/templates/wsfs/docs/Issues_papers/HLEF2050_Global_Agriculture.pdf. Accessed: 2019-05-23.
- Geiger-Lund (2019). Geiger-lund selective asparagus harvesters. <http://www.asparagusharvester.com/Drawing-Model-H-24.html>. Accessed: 2019-05-23.
- Haws, K. (2019). Haws harvester. <http://hawsharvester.com/>. Accessed: 2019-05-23.
- Jimnez, A., Ceres, R., and Pons, J. (2000). A survey of computer vision methods for locating fruit on trees. *Trans. ASAE*, 43. doi:10.13031/2013.3096.
- K. Carpenter, E. Glasgow, T.R.D.S. (2019). Mantis robotic asparagus harvester. <https://sites.google.com/site/mrsdproject201213teamc/home>. Accessed: 2019-05-23.
- Lehnert, C., English, A., McCool, C., Tow, A.W., and Perez, T. (2017). Autonomous sweet pepper harvesting for protected cropping systems. *IEEE Robotics and Automation Letters*, 2(2), 872–879. doi:10.1109/LRA.2017.2655622.
- Leu, A., Razavi, M., Langstdtler, L., Risti-Durrant, D., Raffel, H., Schenck, C., Grser, A., and Kuhfuss, B. (2017). Robotic green asparagus selective harvesting. *IEEE/ASME Transactions on Mechatronics*, 22(6), 2401–2410. doi:10.1109/TMECH.2017.2735861.
- Matlab (2019). Single camera calibration app. <https://au.mathworks.com/help/vision/ug/single-camera-calibrator-app.html>.
- Moulon, P., Monasse, P., Marlet, R., and Others (2017). Openmvg. an open multiple view geometry library. <https://github.com/openMVG/openMVG>.
- Wolfram (2019). Steiner circumellipse. <http://mathworld.wolfram.com/SteinerCircumellipse.html>.