2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Vision Based Guidance for Robot Navigation in Agriculture

Andrew English, Patrick Ross, David Ball, *Member, IEEE*, Peter Corke, *Fellow, IEEE*

Abstract— **This paper describes a novel vision based texture tracking method to guide autonomous vehicles in agricultural fields where the crop rows are challenging to detect. Existing methods require sufficient visual difference between the crop and soil for segmentation, or explicit knowledge of the structure of the crop rows. This method works by extracting and tracking the direction and lateral offset of the dominant parallel texture in a simulated overhead view of the scene and hence abstracts away crop-specific details such as colour, spacing and periodicity. The results demonstrate that the method is able to track crop rows across fields with extremely varied appearance during day and night. We demonstrate this method can autonomously guide a robot along the crop rows.**

I. Introduction

Food production must increase dramatically due to a growing world population [1]. Autonomous farm vehicles have a key role to play in improving farm productivity through automation of existing machinery or new autonomous robotic systems. For many farming operations such as spraying, planting, and harvesting, a crucial proficiency is accurately traversing along the crop rows. The most common commercial solution is to use precision Global Navigation Satellite System (GNSS) receivers to guide vehicles along pre-planned routes. Prior to precision GNSS guidance, the farmer would steer manually using local observations of the rows.

The high cost of precision GNSS and the benefits in using local sensory information has led to research into vehicle guidance using camera images [2]–[5]. The majority of image based crop row tracking techniques rely on green rows with darker soil in-between, as shown in Figure 1(a). These image based techniques first segment crop from soil using colour information, and then fit lines to this binary image.

However, many farm operations are carried out in more visually challenging fields where there is only stubble (dead crop residue) available as a navigation cue, as shown in Figure 1(b). In these fields, guidance becomes difficult or infeasible for segmentation-based crop tracking algorithms.

This paper presents a novel monocular vision based method to track the direction and lateral offset of the crop rows by estimating the dominant parallel planar texture in a simulated overhead view. The method assumes crops are planted in approximately straight rows on reasonably flat ground; assumptions which are held in a wide range of cropping fields, particularly in broad-acre agriculture. In order to generate the overhead view, camera attitude is estimated using an IMU and/or visual horizon tracking. Notably, the method does not perform any binary segmentation of plants and soil, nor does it require any explicit assumptions about the appearance of plants such as row spacing, periodicity, colour, and lighting. The method is applicable to a wide variety of fields with extremely varying appearance, including environments were colour segmentation would be infeasible. We demonstrate that this method can accurately guide an autonomous robot along a variety of crop rows during both day and night.

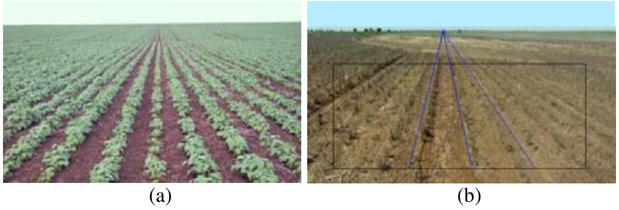
(a) (b)

Figure 1: Agricultural fields have extremely varied appearance. (a) High contrast between plants and soil makes segmenting crop rows simple. (b) Field during fallow period where binary segmentation of crop and soil based on colour would likely fail. The method described in this paper will function in both fields without modification.

The paper is organised as follows. In Section II we describe related methods of visually navigating with respect to crop rows. Section III describes our method in detail while Section IV presents our experimental setup. We present tracking results from a wide variety of crops in Section V before concluding remarks in Section VI.

II. Related Work

There are two methods to tracking crop rows, using 3D structure or using image based techniques.

Methods using 3D structure use either scanning lasers [6], or stereo vision [7], [8]. These methods require a distinct height difference between the crop and ground and so cannot be used for example on very young crops.

Reid and Searcy [2] describe an early image based row tracking method that tracked crop rows by performing a binary segmentation on near-infrared images of crop rows. Other notable examples include [3]–[5]. Most monocular vision methods first perform a binary segmentation step between plants and soil (usually based on colour), then fit lines to this binary image using various methods such as the Hough transform [9]–[11], linear regression [5] or fixed template matching [4].

Line fitting methods such as the Hough transform, however, are not robust to uncertainty in the environment and can extract ‘incorrect’ lines leading to navigation failure. Hiremath et al. [12] improved robustness by filtering together wheel odometry and camera imagery of the two closest rows using a particle filter. They also avoided specifying some environment specific parameters including row width and spacing by including them in the filter’s state space.

Binary segmentation of crop and plants becomes difficult when there is partial shadowing or weeds between crop rows. An approach that avoids segmentation is to search for periodic variations of the grayscale image relating to the

The authors are with the School of Computer Science and Electrical Engineering, Queensland University of Technology, Australia. (e-mail: a.english@qut.edu.au).

978-1-4799-3685-4/14/$31.00 ©2014 IEEE

periodicity of crop rows [5], [13], [14]. These approaches restrict their field of view to small patches or sets of horizontal strips since they do not estimate or account for variations in the camera pose which would lead to a "smearing" of the row signal for larger fields of view.

III. Algorithm Design

We model crop rows as an arbitrary planar parallel texture, which allows us to abstract away crop-specific details such colour, spacing and periodicity. We use a novel method of extracting the direction of the dominant parallel texture from a simulated overhead view, which we then use to track the lateral offset of the vehicle. We also estimate the attitude of the camera to ensure the overhead view is correct and stable. This allows a comparatively wide field of view without "blurring" rows together and gives improved performance in noisy environments where crop rows are faint.

Our overall approach to tracking crop rows, shown in Figure 2, is as follows.

- Pre-process the image to correct for lens distortion and downsample to improve processing speed.
- Use an IMU and/or detect the horizon in the image to estimate the vehicle's roll and pitch.
- Stabilise the image with the estimated roll and pitch.
- Warp the stabilized image into an overhead view.
- Estimate the vehicle's heading relative to the crop rows by estimating the direction of the dominant parallel texture in the overhead image.
- Correct for heading in the overhead view by skewing the image using the estimated heading.
- Generate a "frame template" by summing the columns of the skewed images.
- Estimate lateral motion relative to the crop by comparing this template to initial "crop template".

The robot operates in a local coordinate frame aligned with the crop rows and centred at the robot's position when tracking begins. The camera pose $\mathbf{P_c}$ within this coordinate frame is defined as

$$\mathbf{P_c} = [x_c, y_c, h_c, \theta_c, \phi_c, \gamma_c] \tag{1}$$

where x_c and y_c are the camera coordinates, h_c is the camera height, θ_c, ϕ_c and γ_c are the camera roll, pitch and yaw. We assume a constant h_c, since the camera is mounted on a vehicle. We do not track distance travelled in the direction of the rows (change in y_c), however all the remaining camera pose variables (x_c, θ_c ϕ_c and γ_c) are estimated.

The states θ_c, ϕ_c and γ_c are each tracked with their own one dimensional Kalman filter to improve resilience to noise and missing data. An IMU may be used to augment or replace vision based estimates of the vehicle's attitude at night time or in environments/conditions where the horizon is not clearly visible. The IMU rotational rates drive each filter's prediction step while the estimates from image processing and the gravity vector from the IMU form the filter measurements. Many vehicles may not need image stabilisation at all, for example larger unsuspended vehicles such as tractors tend to sit relatively flat compared to the local terrain, and the algorithm is somewhat tolerant to small errors in camera attitude.

The significant steps in tracking the rows are described in the following subsections.

A. Horizon Detection

In this section we describe our method of visually tracking the horizon to estimate the vehicle's roll and pitch. The broad-acre fields where this work was carried out are vast, flat and almost entirely obstacle free, allowing a clear and unobstructed view of the horizon.

Horizon tracking begins with selecting the image region within three standard deviations of estimated horizon position (estimated by the Kalman filters for roll and pitch). Pixels in this sub-image are classified as "sky" or "ground" by fitting a plane through RGB space that is equidistant to the median RGB value of the top and bottom 30% of the image. This is

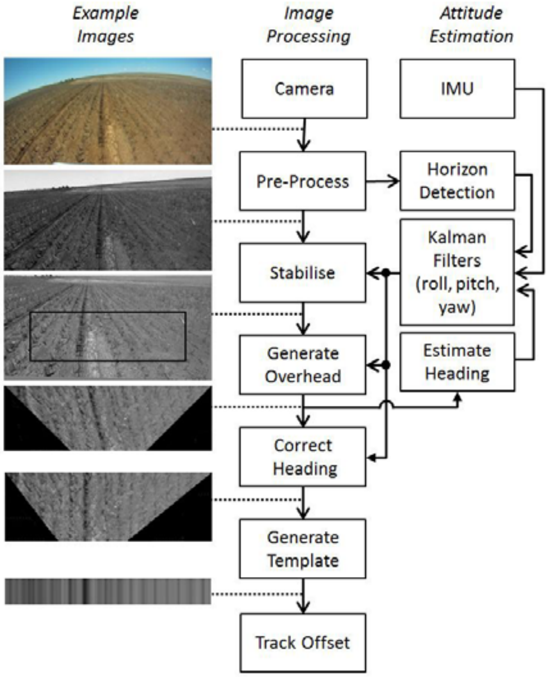

Figure 2: Block diagram of algorithm along with example images.

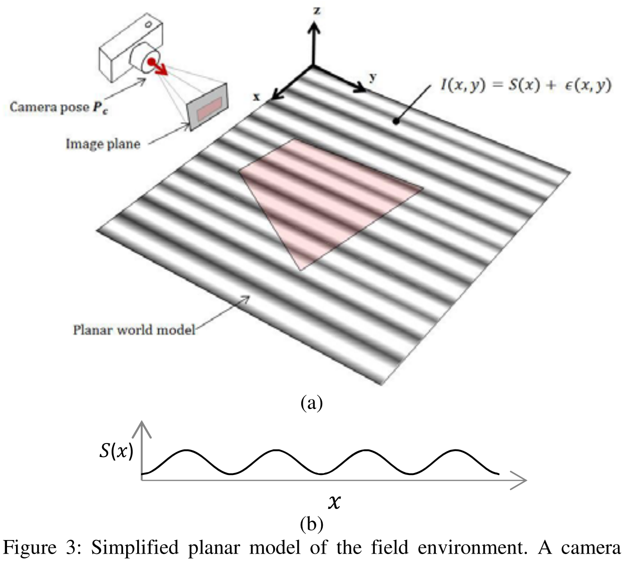

Figure 3: Simplified planar model of the field environment. A camera with pose $\boldsymbol{P_c}$ views the planar world which is represented by the intensity function $I(x,y)$. (b) Plot of $S(x)$.

similar to[15], however the plane in RGB space is calculated per-image rather than pre-calculated. The horizon is found by fitting a straight line to the binary image using RANSAC.

B. Generating the Overhead View

Next we construct a simulated overhead view of the crop rows to make crop rows parallel. The undistorted grayscale image is rotated and cropped so that the horizon lies at the top of the image. A rectangular region of interest is then warped to an overhead view using a reverse perspective transform. Since the horizon position is now fixed, the homography matrix defining this transform is also fixed, so the warping becomes a pre-computed pixel remapping.

C. Estimating Row Direction

Our method of measuring the heading angle γ_c is based on a novel method of finding the direction of the dominant parallel texture (the crop rows) in the overhead image. In short, this method involves iteratively skewing the overhead image by varying angles, summing the skewed image along the columns, then calculating the variance of the resulting vector. The skew angle which results in the greatest variance is our estimated the heading angle γ_c. In this section we present a theoretical justification for this method based on a simple a model for approximating the appearance of the crop rows.

We approximate the field as a horizontal planar surface with the crop rows represented by an arbitrary one dimensional "crop template" function $S(x)$ which is projected onto the plane in the direction of the rows (along the y axis) as shown in Figure 3. The intensity at any point on the plane is defined as

$$I(x,y) = S(x) + \varepsilon(x,y). \tag{2}$$

where the $\varepsilon(x,y)$ term is an unknown noise function that represents the component of the crop appearance that does not fit the linear parallel pattern model (e.g. texture in the crop and ground, missing plants, weeds etc.). If we model $\varepsilon(x,y)$ as a random variable with zero mean, then the crop template $S(x)$ is recoverable from the intensity function $I(x,y)$ by integrating along the direction of the rows since this will "integrate out" the noise.

$$S(x) = \lim_{y_0 \to \infty}\left(\frac{1}{y_0}\int_0^{y_0} I(x,y)dy\right) \tag{3}$$

The navigation method presented in this paper is centred on recovering and tracking an estimate of the crop template function $S(x)$ from the overhead image. To do this we perform the integration in (3) on the overhead image, however we must first estimate the camera heading with respect to the rows.

We define the n column by m row grayscale overhead image as $O(i,j)$, and note that it is a rotated and scaled view of the row intensity function $I(x,y)$ with crop rows angled by our heading γ_c. We can therefore recover a scaled version of the crop template, $S(x)$ - which we name $\hat{S}(i)$ – by integrating the overhead image in a direction parallel to the crop rows. That is

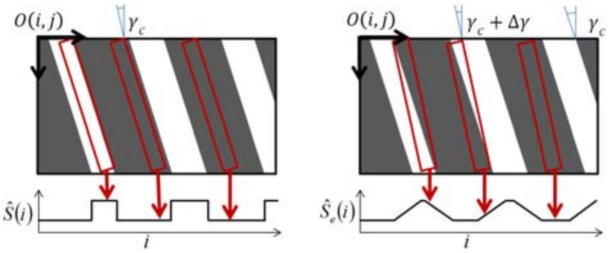

Figure 4: (a) Integrating the overhead image $O(i,j)$ in the direction of the crop rows γ_c gives an estimate of the crop template $\hat{S}(i)$. (b). Integrating the $O(i,j)$ in a direction misaligned by $\Delta\gamma$ gives an erroneous template $\hat{S}_e(i)$ which is a smoothed version of $\hat{S}(i)$ and has a lower variance.

$$\hat{S}(i) \approx \frac{1}{m}\int_0^m O(i - j\tan(\gamma_c), j)dj \tag{4}$$

However γ_c is thus far unknown. If we apply equation (4) with a heading error of $\Delta\gamma$ we will get an erroneous crop template signal

$$\hat{S}_e(i) = \frac{1}{m}\int_0^m O(i - j\tan(\gamma_c + \Delta\gamma), j)dj \tag{5}$$

$$= \frac{1}{k}\int_0^k \hat{S}(i-\tau)d\tau \tag{6}$$

$$\text{where } k = m\tan(\gamma_c + \Delta\gamma) - m\tan(\gamma_c). \tag{7}$$

When calculating the erroneous signal $\hat{S}_e(i)$ we are integrating the image along a direction that is misaligned with the rows. This means the erroneous crop template $\hat{S}_e(i)$ will be a "smoothed" version of the true crop template $\hat{S}(i)$ since we are summing through several shifted copies of $\hat{S}(i)$ as shown in Figure 4. Equation (6) describes convolution with a rectangular window of width k and height 1/k, which is a box filter, a well-known low-pass filter. As our heading error $\Delta\gamma$ approaches zero, this filter will pass all frequencies. We can therefore estimate the value γ_c by maximizing the variance of the erroneous crop template $\hat{S}_e(i)$ since the variance of $\hat{S}_e(i)$ will be maximum when all frequencies of the true template $\hat{S}(i)$ are unattenuated. That is

$$\gamma_c = \underset{\gamma}{\operatorname{argmax}}\left(\operatorname{var}\left(\hat{S}_e\right)\right) \tag{8}$$

In practice, we implement (8) on the overhead image in a different but equivalent way. As mentioned previously, we instead iteratively skew the overhead image by varying angles γ using an affine transform, sum the columns, then calculate the variance of the resulting vector to find the skew angle that results in the greatest variance. This is simple to implement and fast to execute since well optimised functions for skewing and summing the columns of images already exist in many image processing frameworks such as OpenCV and MATLAB. Additionally, we also take the derivative of the vector, as well as apply a low-pass filter before computing the variance, which has been experimentally found to improve performance by rejecting frequencies that do not contain crop row information. The low-pass filter was tuned experimentally to allow identical parameters to work well across all fields tested. We apply skews angles of between +/- 30 degrees in increments of 1 degree. This estimate is then refined using gradient ascent since it observed that the variance function is locally convex around the true yaw value. The variance values calculated at 1 degree increments are reused to detect the presence of a valid row texture as explained in section E.

D. Estimating Row Offset

While estimating the heading in the previous section we recovered a crop template function $\hat{S}(i)$ from the overhead image. This is the signal that we track directly to estimate lateral deviation of the camera perpendicular to the direction of the rows.

We only track relative offsets from the first frame. The first vector $\hat{S}(i)$ that we recover is the *crop template.* Subsequent templates, called *frame templates* are matched against the crop template using Zero Normalised Cross Correlation (ZNCC). We also search over a small range of scales to compensate for rugged terrain where the camera will bounce up and down, since this will change the camera height and "stretch" the frame template.

When the frame template ZNCC match is below a threshold, the crop template is changed to the current frame template. This will cause a small amount of drift in the estimated offset when changing templates, but allows tracking to continue despite changes in crop appearance.

We do not explicitly track offset with respect to any particular crop rows in this work. If this is required (i.e. to centre the vehicle over a row) it can be achieved by shifting the crop template sideways to align its centre with the nearest local maximum or minimum representing a crop row.

Our offset is tracked in pixels at a point on the ground ahead of the camera. The location of this point and the estimated offset can be converted to metric distances if the height of the camera, the camera parameters, and the coordinates of region of interest are known.

E. Detecting Rows

There are sections of the field where there are no crop rows to track, for example at the ends of the field or in bare patches of field. We detect these situations by examining the output of the iterative search for the heading value in (8) where we calculated the variance of the frame template for various values of γ between +/- 30 degrees in increments of 1 degree. This gives a vector of variance values $v(\gamma)$. An image is deemed to have a valid row texture if it passes the following test

$$valid = \begin{cases} 1, & thresh < \dfrac{\max(v(\gamma))}{\operatorname{mean}(v(\gamma))} \\ 0, & otherwise \end{cases} \tag{9}$$

For this work we use a threshold of 2. This heuristic is based on the observation that if there is no directional texture in the overhead image, then all values of $v(\gamma)$ will be similar. However, if there is a strong directional texture in the overhead image (i.e. rows clearly present) then $v(\gamma)$ will have a strong peak for the correct value γ and will drop off either side of that peak.

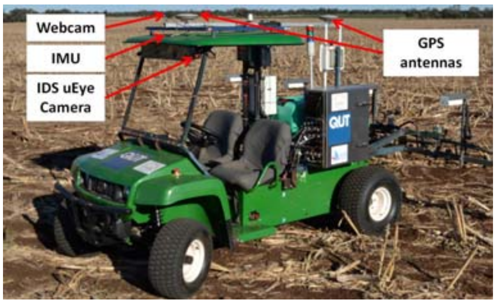

Figure 5: Robotic test platform used in this study with sensors labelled.

If a frame fails this test, the heading measurement is not passed to the Kalman filter. If many sequential images fail this test we can fall back on another method of positioning or cease tracking and command the vehicle to stop.

IV. Experimental Setup

The experiments were performed on a variety of field types on a broad-acre farm near Emerald in Australia. Example images from these fields are shown in Figure 6.

The platform used in this study is a robotic platform developed for spraying weeds on broad-acre fields and is based on a John Deere Gator TE electric utility vehicle modified for autonomous operation. Computation is handled by a pair of standard PCs running Ubuntu 12.04 and the open-source Robot Operating System (ROS) middleware. The row tracking algorithm is implemented in C++ using OpenCV. We view the crop with an IDS uEye CP, or a low cost web camera the Microsoft LifeCam Cinema. We also use a low cost IMU, the CH Robotics UM6 to help estimate the camera attitude. We use a precision RTK-GPS/INS system (a Novatel FlexPack with Tactical Grade IMU) to verify the results.

Our first experiments were designed to show the ability to track rows in a wide variety of conditions. The robot was driven manually along crop rows and sensor data was post-processed with the algorithm described to output the crop offset and angle. The estimated offset is projected back to the GPS antenna position using the estimated heading so that we can compare results to the Novatel GPS.

The second experiment looks at the availability of the row tracking signal on a sorghum stubble field using data recorded during a two hour experiment covering six hectares.

Finally we show results from a closed loop experiment where the robot was guided along crop rows. We used the offset output by our algorithm to control the steering angle of the robot via a Proportional-Integral (PI) controller to follow straight crop rows at a constant speed and examine the path taken along the straight crop rows using the GPS.

V. Results

A. Open Loop

We present open loop results in Figure 6 from four different crops: wheat, sorghum stubble (both day and night), and chickpeas with wheat stubble. For each dataset we show an example image and the corresponding heading corrected overhead view with overlaid plots of the crop and frame templates. Additionally we show a plot of the estimated offset alongside the offset measured via the GPS system.

Each environment differs dramatically in appearance, however row tracking was successful in each case. Parameters were left unchanged between datasets except for parameters related to the camera. All datasets were recorded with an IDS camera except for the chickpeas which was recorded with the webcam to demonstrate our method functions with very low cost sensors.

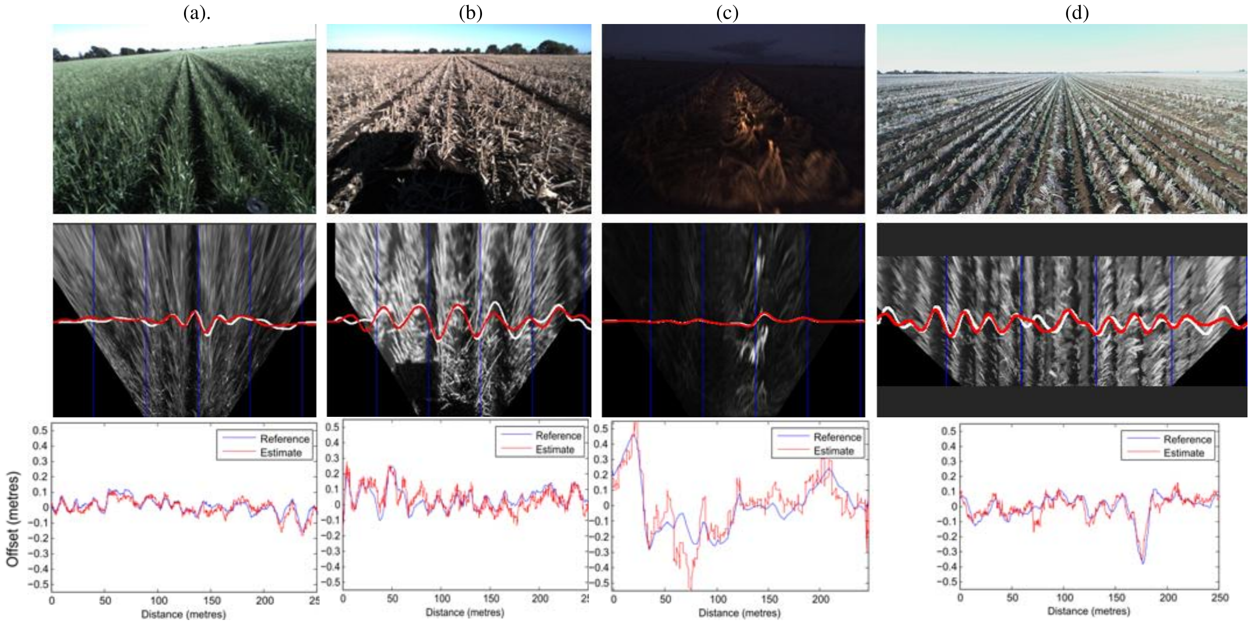

Figure 6: Results for open loop experiments in four different environments. Columns represent (a) wheat (b) sorghum stubble (c) sorghum stubble at night under headlights (d) chickpeas with wheat stubble. The second row shows heading corrected overhead images with plots overlaid of the crop template (red) and current frame template (white). The third row shows the offset estimated by both the row tracking algorithm and the GPS/INS system. Datasets(a)-(c) were captured with IDS uEye. Dataset (d) captured with Microsoft Lifecam Cinema webcam.

RMS errors for each dataset are shown in Table 1. The wheat and chickpeas datasets shows the least offset error which we attribute to the narrow spacing and relatively clear crop template. The chickpeas with wheat stubble was one of the more challengin g datasets since we are tracking two crops simultaneously that occasionally shift relative to each other forcing template changes. The sorghum dataset showed a comparatively large offset which is attributed to the wide row spacing, and occasionally ill-formed rows. Notice the partial shadow from the vehicle in this dataset did not affect the tracking. Night time tracking unsurprisingly showed the largest offset errors, and would likely benefit from additional headlights to widen the well-lit area to more than one crop row.

TABLE I. RMS ERROR FOR OPEN LOOP EXPERIMENTS (M)

(a). Wheat	*(b). Sorghum Stubble*	*(c). Sorghum Stubble Night*	*(d). Chickpeas with Wheat Stubble*
0.034	*0.060*	*0.100*	*0.048*

B. Tracking Availability

Figure 7 shows the availability of the row tracking signal using data captured from the robot while being guided via GPS to cover a six hectare field by performing five metre swaths traveling at approximately 5km/h.

The three major diversions from the path were due to the robot avoiding obstacles. The row signal was successfully acquired on the majority of straight sections (blue dots in Figure 7). Since we do not track row headings outside +/-30 degrees, we cannot track crop rows while turning around, or while avoiding obstacles (orange dots in Figure 7). The majority of the remaining locations where a valid row tracking signal was unavailable (red dots in Figure 7) can be attributed to either the headland turning area where rows are planted in multiple directions, an unplanted grassy area around a power pole, and a contour bank (raised ridge for diverting water) which violates the flat ground assumption made by the row tracking algorithm. The remaining places where row tracking is lost are locations where the parallel texture was too weak for our algorithm to detect the direction. Figure 8 shows example images from this test of

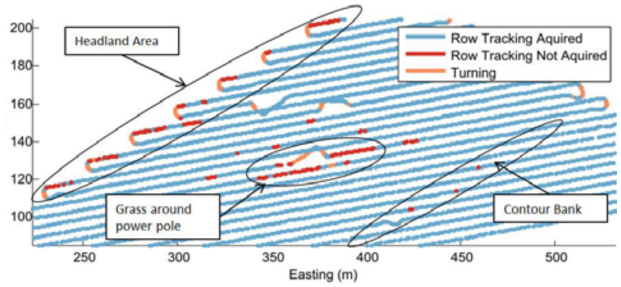

Figure 7: Availability of the row tracking signal during a six hectare coverage operation. Most tracking failures are due to the presence of the headland area, grass around a power pole and a contour bank (ridge). We only show part of the coverage pattern focus on the most relevant results.

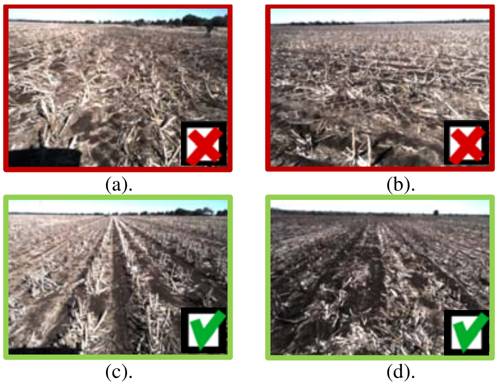

Figure 8: Example frames from tracking availability experiment. (a) Unsuccessful tracking in headland. (b). Unsuccessful tracking due row heading > 30 degrees. (c). Successful tracking with relatively clear rows. (d) Successful tracking with relatively unclear rows.

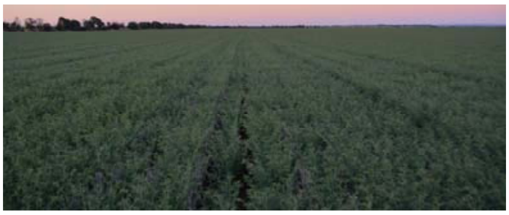

Figure 9: Example of field with row texture too weak to reliability track.

frames where tracking was both unavailable and available. The sorghum stubble is a particularly challenging environment for visually navigating with respect to rows since the stubble can be quite chaotic. We envisage row tracking to be complemented by other navigation sensors such as wheel odometry and low-cost GPS to enable navigation to bridge these gaps in row tracking. Row tracking will fail in fields with insufficient parallel texture. Figure 9 shows an example of such as field where row of chickpeas have grown together to almost entirely cover the soil, and was found to not provide enough parallel texture for consistent and reliable tracking.

C. Closed-Loop Robot Guidance

Figure 10 and Figure 11 show the offset error while guiding the vehicle along both wheat and sorghum stubble rows. The robot successfully guided itself along the crop rows in both cases. The RMS error is 28mm and 120mm for the wheat and sorghum stubble respectively. We attribute the larger error for the sorghum stubble to the much wider row spacing and noisier row signal. Interestingly the RMS error is approximately one tenth of the row spacing in both cases. The largest diversion at around 300m occurs while the robot is driving at an angle over a contour bank (ridge). The rows are likely to not have been straight in this location since GPS guided tractors commonly do not compensate for the tilt of the vehicle as they drive at an angle over contour banks causing the planted rows to wobble. Following these wobbles in the crop rows is advantageous for many operations and highlights the usefulness of using local sensory information.

VI. Conclusion

This paper demonstrated a method of positioning an agricultural vehicle relative to crop rows that is easily applicable to a wide variety of crop types without modification. Importantly it requires no segmentation step, and is able to function in environments where the crop signal is weak or noisy. The method requires an estimate of the camera attitude with respect to the ground, which in this work is assumed to be level. A possible solution to enable operation in sloping fields is to use only the higher frequency components of the measured roll and pitch to correct for local bumps since the average vehicle attitude over a longer time scale will likely follow the slope of field. It was also observed during testing that the method is somewhat tolerant of errors in roll and pitch, and in many instances tracking was successful without any stabilisation at all. Future work will attempt to quantify the sensitivity of the method to errors in camera attitude and will look at methods of relaxing the need for an external stabilisation.

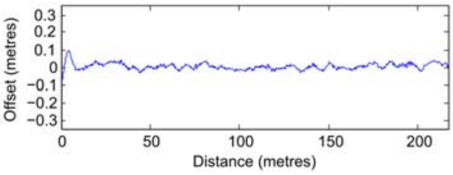

Figure 10: Offset error during closed loop experiment while guiding the robot along wheat rows

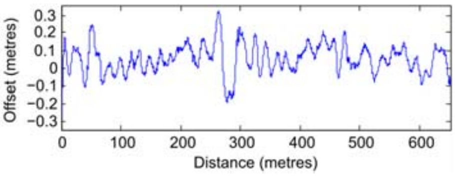

Figure 11: Offset error during closed loop experiment while guiding along sorghum stubble rows.

Additionally, future work will investigate more sophisticated methods of tracking the offset of crop templates, as well as fusing row tracking information with other navigation sources such as low-cost GPS and odometry.

Acknowledgment

This work was supported in part by the Australian Research Council Linkage Project LP110200375 “Robotics for Zero Tillage Agriculture” awarded to the Queensland University of Technology, SwarmFarm Robotics and the Australian Centre for Field Robotics.

References

[1] K. Wiebe, “How to feed the world in 2050,” in *Insights from an expert meeting at FAO*, 2009.

[2] J. Reid and S. Searcy, “Vision-based guidance of an agriculture tractor,” *Control Syst. Mag. IEEE*, April 1987.

[3] J. Billingsley and M. Schoenfisch, “The successful development of a vision guidance system for agriculture,” *Comput. Electron. Agric.*, no. 2, pp. 147–163, 1997.

[4] N. D. Tillett, T. Hague, and S. J. Miles, “Inter-row vision guidance for mechanical weed control in sugar beet,” *Comput. Electron. Agric.*, vol. 33, no. 3, pp. 163–177, Mar. 2002.

[5] H. T. Søgaard and H. J. Olsen, “Determination of crop rows by image analysis without segmentation,” *Comput. Electron. Agric.*, vol. 38, no. 2, pp. 141–158, Feb. 2003.

[6] A. Ruckelshausen, P. Biber, M. Dorna, H. Gremmes, R. Klose, A. Linz, F. Rahe, R. Resch, M. Thiel, D. Trautz, and U. Weiss, “BoniRob: an autonomous field robot platform for individual plant phenotyping,” *Precis. Agric.*, vol. 9, p. 841, 2009.

[7] M. Kise and Q. Zhang, “Development of a stereovision sensing system for 3D crop row structure mapping and tractor guidance,” *Biosyst. Eng.*, vol. 101, no. 2, pp. 191–198, Oct. 2008.

[8] K. Hanawa, T. Yamashita, Y. Matsuo, and Y. Hamada, “Development of a Stereo Vision System to Assist the Operation of Agricultural Tractors,” *JARQ-JAPAN Agric. Res. Q.*, vol. 46, 2012.

[9] B. Åstrand and A.-J. Baerveldt, “A vision based row-following system for agricultural field machinery,” *Mechatronics*,Mar. 2005.

[10] J. a. Marchant, “Tracking of row structure in three crops using image analysis,” *Comput. Electron. Agric*, Jul. 1996.

[11] T. Bakker, H. Wouters, K. van Asselt, J. Bontsema, L. Tang, J. Müller, G. van Straten, and K. Van Asselt, “A vision based row detection system for sugar beet,”*Comput. Electron. Agric.*, Jan. 2008.

[12] S. Hiremath, F. van Evert, G. Heijden, C. J. F. ter Braak, and A. Stein, “Image-Based Particle Filtering For Robot Navigation In A Maize Field,” in *Workshop on Agricultural Robotics, IROS 2012, Vilamoura, Portugal, 11-10-2012.*, 2012.

[13] T. Hague and N. Tillett, “A bandpass filter-based approach to crop row location and tracking,” *Mechatronics*, vol. 11, 2001.

[14] H. J. Olsen, “Determination of row position in small-grain crops by analysis of video images,” *Comput. Electron. Agric.*, vMar. 1995.

[15] S. Thurrowgood, D. Soccol, R. Moore, D. Bland, and M. V Srinivasan, “A vision based system for attitude estimation of UAVs,” in *IROS 2009. IEEE/RSJ International Conference on*, 2009.