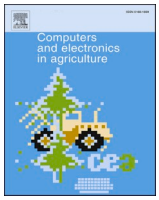
Computers and Electronics in Agriculture 197 (2022) 106941

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag



Robotic crop row tracking around weeds using cereal-specific features☆

Xiaodong Li, Rob Lloyd, Sarah Ward, Jonathan Cox, Shaun Coutts, Charles Fox *

University of Lincoln, Brayford Pool, Lincoln LN6 7TS, UK

ARTICLE INFO

Keywords:
Precision agriculture
robotics
row following
cereals

ABSTRACT

Crop row following is especially challenging in narrow row cereal crops, such as wheat. Separation between plants within a row disappears at an early growth stage, and canopy closure between rows, when leaves from different rows start to occlude each other, occurs three to four months after the crop emerges. Canopy closure makes it challenging to identify separate rows through computer vision as clear lanes become obscured. Cereal crops are grass species and so their leaves have a predictable shape and orientation. We introduce an image processing pipeline which exploits grass shape to identify and track rows. The key observation exploited is that leaf orientations tend to be vertical along rows and horizontal between rows due to the location of the stems within the rows. Adaptive mean-shift clustering on Hough line segments is then used to obtain lane centroids, and followed by a nearest neighbor data association creating lane line candidates in 2D space. Lane parameters are fit with linear regression and a Kalman filter is used for tracking lanes between frames. The method is achieves sub-50 mm accuracy which is sufficient for placing a typical agri-robot's wheels between real-world, early-growth wheat crop rows to drive between them, as long as the crop is seeded in a wider spacing such as 180 mm row spacing for an 80 mm wheel width robot.

1. Introduction

1.1. Motivation

Cereal crops account for a large portion of human foods, with wheat forming around 30% of intake in the UK after processing into flour, bread, breakfast cereals, pasta and other baked goods and processed foods (Shewry and Hey, 2015). High intensity arable farming uses substantial and increasing levels of pesticide per year (Sharma et al., 2019; Zhang, 2018). These chemicals have important environmental and economic costs (Sharma et al., 2019). In many areas of agriculture, precision autonomous robotics promises to maintain or even increase yields while drastically reducing inputs such as pesticides and fertilizers, and their associated environmental and economic costs of food production (Duckett et al., 2018).

A fundamental prerequisite of many autonomous precision agriculture tasks for cereals – including weed control, crop monitoring, fertilizing, and harvesting – is the ability to autonomously navigate across a field of row crops at different growth stages without damaging the crop. This requires a vehicle to position its wheels in the inter-row regions to avoid driving over the crop. Real-time row following will also enable implements to be positioned relative to the rows to target either the rows or the inter-row regions as needed by different operations.

Row following is well-developed for vegetable crops, where commercial systems are now available (e.g. RoboCrop, Garford Farm Machinery Ltd, https://garford.com/). With vegetables, each plant can be seen as an individual and the rows are found as lines of individuals. However, row following is much more challenging in cereal crops. As grasses, cereals are drilled much closer together, both within the row and between rows, so that the plants overlap and cannot be segmented into individual plants to find the rows.

It is important to define two types of canopy overlapping: *Intra-row canopy closure* will refer to the state in which plants within a row are overlapping, but the rows remain clearly separated and identifiable. *Inter-row canopy closure* will refer to what is usually called just 'canopy closure', and refers to the state in which plants are overlapping both within rows and also between rows. In this state, there may be no discernible rows, and all that can be seen by humans is a continuous canopy.

If the challenge of row following after intra- and inter-row canopy closure cannot be solved, then any automated robotic system will only be able to access the crop during the very early part of the growing

☆ Thanks to Simon Pearson who helped to obtain funding for the project. This work has received funding from InnovateUK grant number 105137.
* Corresponding author.
E-mail address: chfox@lincoln.ac.uk (C. Fox).

https://doi.org/10.1016/j.compag.2022.106941
Received 4 October 2021; Received in revised form 28 February 2022; Accepted 3 April 2022
Available online 23 April 2022

(a) Typical crop rows, farm 1 Field

(b) Challenging crop rows, farm 1 Image

(c) Typical crop rows, farm 2

(d) Challenging crop rows, farm 2

Fig. 1. Experimental wheat images taken on two different farms, showing a typical and challenging image from each.

season, and will be excluded from the crops during the most active growth periods, when operations such as fertilizer application are crucial.

Cereal crops may be drilled with various intra and inter row distances. Inter-row distances are usually tightly controlled, while intra-row distances may be more random. Wheat is usually drilled with inter-row spacing of between 125 mm and 180 mm, but spacing of up to 300 mm is used in niche circumstances, such as by organic growers. Intra-row distances typically vary between 30–60 mm.

Cereal crops are often growing in the presence of weeds, and indeed removing these weeds is one of the major applications of precision robotics. Blackgrass (*Alopecurus myosuroides* (Hubbard, 1984)) is a current particular weed threat to UK wheat. It is especially difficult to control because as another grass it is visually and biologically similar to wheat, and has evolved resistance to herbicides. The crop row structure may be obscured by the presence of grass weeds growing between the rows, which may – as with blackgrass – look visually similar or identical to the crop. To distinguish them may thus require fusions of knowledge from the row structure itself – i.e. that any grasses growing between the expected row locations are weeds, even if they look like the crop in the rows. This presents a chicken-and-egg problem if row locations are also being determined by the locations of the crop-like structures.

Fig. 1 shows typical images of emerging wheat rows. Notable features include missing plants making some rows hard to determine, and blackgrass weeds between the rows also confusing the row structure.

To follow cereal rows in the presence of cereal-like weeds in agricultural practice, there should be minimal need for large scale data collection and annotation; there must be structural information used to understand row structure even when visual images are heavily obscured by weeds or missing crop plants; and methods should be computationally tractable on basic low powered hardware in order to deploy them on vehicles in the field.

1.2. Related work

Visual lane detection has been studied in the context of autonomous ("self-driving") cars, with a focus on following traffic lanes defined by line markings on roads, surface and edge types. Most systems are designed based on similar image processing techniques (Rezwanul Haque et al., 2019; Lee and Moon, 2018; Song et al., 2018; Li et al., 2013; Mingfa Li and Jiang, August 2018; Yoo et al., 2017; Ozgunalp et al., 2017; Ozgunalp et al., 2017; Piao and Shin, 2017; Khalilikhah et al., 2016).

Agriculture row following has typically applied the same methods as used on roads (Latha et al., 2014; Hassanein et al., 2019; Winterhalter et al., 2018; Montalvo et al., 2012; Romeo et al., 2012; Ramesh et al., 2016; Zhang et al., 2018). These approaches largely exploit the fact that crops are green against a brown soil background. This colour contrast has been used in RGB colour-space in combination with edge detection (Latha et al., 2014). HSV colour-space, which separates colour intensity from hue, has also been used with images from aerial imagery (Hassanein et al., 2019). Crop row orientation was then determined by Principal Component Analysis of hue value (Hassanein et al., 2019). Winterhalter et al. (2018) presents a pipeline for reliable plant segmentation in any crop growth stage, which adapts the Hough transform for line detection to detect a pattern of parallel equidistant lines for tiny plants. Crop rows have also been found using simple color-space thresholding to obtain the high pressure crops segmentation (Montalvo et al., 2012). More complicated image processing pipelines use image segmentation and colour thresholds and fuzzy clustering as a first step to separate green plants from the background (Romeo et al., 2012). The crop rows were then detected based on image perspective projection and searching for the maximum accumulation of segmented green pixels along straight alignments. Ramesh et al. (2016) provides research in an alternative domain, i.e. detection of rows in crops planted as rows acquired using Low altitude remote sensing in the visible spectrum. Spectral-spatial methods were applied with K-Means clustering for spectral clustering. Zhang et al. (2018) apply a modified vegetation index and double thresholding combining the Otsu method with the particle swarm optimization to separate between the weeds and crops, utilized the position clustering algorithm and shortest path method successively to confirm the final clustered feature point set, linear regression was employed to fit the crop rows. A limited amount of work addresses both straight and curved crop rows detection, but only for the initial growth stages of crop and weed plants where the lanes/rows are isolated to each other (García-Santillán et al., 2017). The majority of the above studies use two common stages: colour-based pixel clustering to detect green crops; and line building – commonly through linear regression or the Hough transformation.

The previous studies addressed crop row following primarily as a segmentation task, segmenting images into crops and background soil, assuming that the crops exist in clear rows. In contrast, the present study is concerned with more noisy, cereal cases in which the row structure is not already clean from single images due to canopy closure and missing crops. Row following is a multi-feature detection and tracking problem that remains a challenge for conventional computer vision in

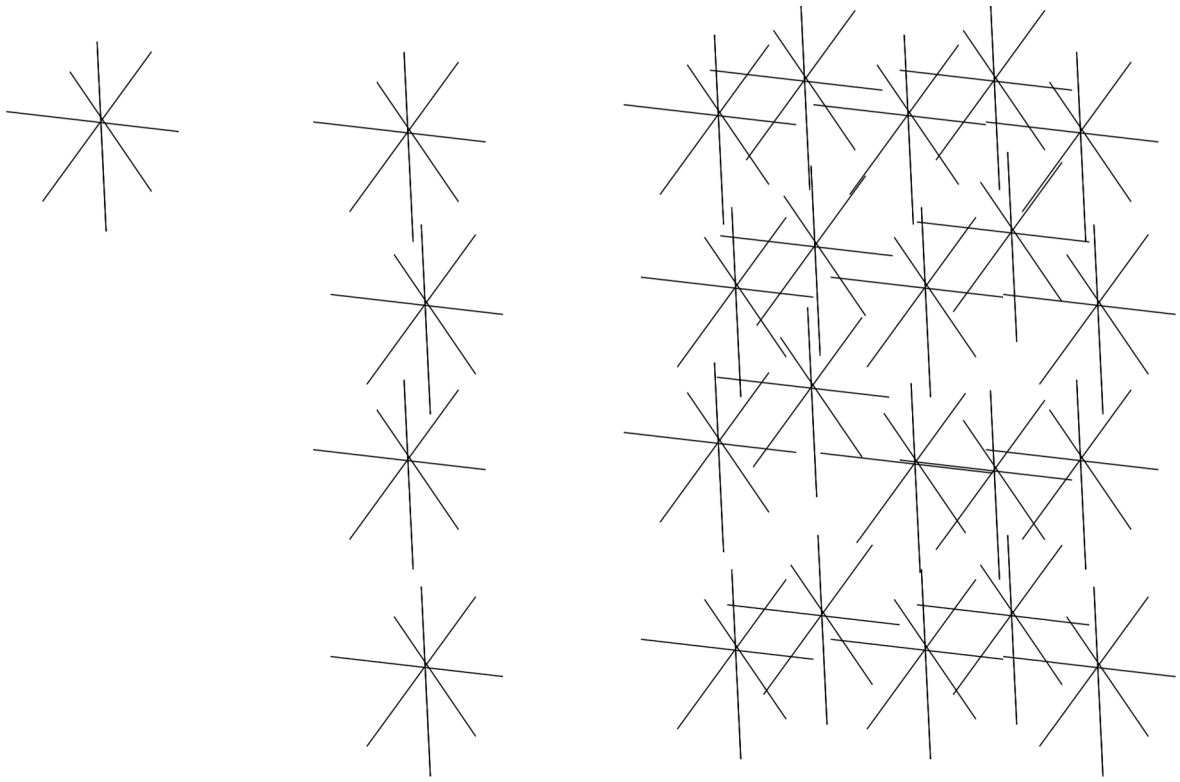


Fig. 2. Leaf direction structure. (a) a single plant, viewed from above, has leaves in all directions. (b) a row of plants have vertically aligned leaves. (c) Overlapping rows have vertically aligned leaves along the row center, but not at location between rows.

environments of ambiguous color, shape and texture, which change with ambient light conditions. Although modern supervised machine learning such as deep neural networks is good at identifying individual objects, conventional machine vision approaches are still needed to group these objects into lanes and understand their relationship to each other. Apart from the time cost of data collection and processing required by supervised learning methods (preparation, sampling, labeling, framework building, training, iterative validating and testing, etc.), which has become the major constraint on quick development, there is also a need for improvements on full implementation in the efficiency and accuracy for specific targets. In some settings, such as low-power compute systems in the field, reliance on deep neural network hardware is still not ideal, and conventional feature based systems may be cheaper and simpler. However, current color-based segmentation methods alone, without structure knowledge, lack the capability to detect overlapping objects within the same color domain. This is an important constraint on the potential of precision agriculture because major crops like wheat show high levels of overlap and occlusion through most of the growing season, with further complications coming from grass weeds that are similar in both color and shape, which do not respect the row structure (Moss, 2017).

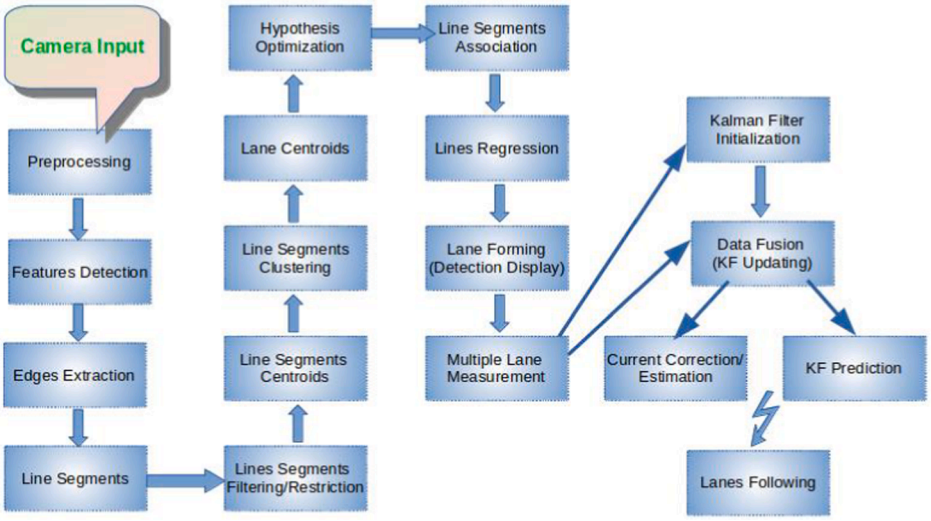


Fig. 3. System design.

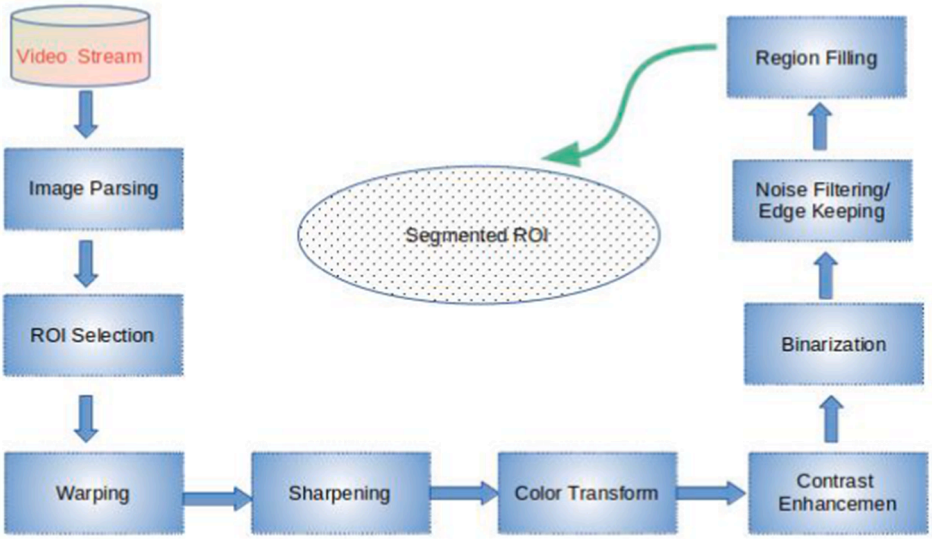


Fig. 4. Image pre-processing.

1.3. Contributions

To achieve the goal of real-time, in-field, cereal row following, in the presence of grass weeds, we regard row following as continuous lane detection and tracking across a series of images. The two novel contributions of this paper are:

A first key concept, presented in Section 2.3, is then to overcome inter-row canopy overlapping. Assuming that individual plants have leaves projecting in all directions, and that their inter-row spacing is tightly controlled but their intra-row spacing is random, then we observe as in Fig. 2 that the leaves growing parallel to the row directions tend to align along the row centers. The inter-row areas may be covered by leaves projecting from neighbouring rows, but these must be leaves which are growing in directions different from the row directions. Due to the higher randomness of intra-row spacing than inter-row spacing, they do not tend to align with one another as the row-directed leaves do. In Fig. 2(c), the vertical leaves clearly align with one another and these lines show the positions of the row centers, while the inter-row areas are filled with more randomly directed leaves. We exploit this regularity by searching for a number of *line segments*, to obtain the maximum accumulation of segmented pixels along vertical "beam-forming" alignments, by applying an angle (e.g. 60°-120°) constraint as line segments selection for the determination of lane candidates.

A second key concept, presented in Section 2.4, is the use of mean-shift clustering (Keinosuke and Hostetler, 1975), an unsupervised machine learning method that provides an adaptive lane separation strategy, with neighboring line segments associated with corresponding lanes (using a dictionary structure).

Standard methods are then used on the above representations. Linear regression is then utilized for the line fitting to form the separated crop rows. Data association and Kalman filtering is then used to perform row following by associating and tracking the multiple detected lanes across frames. The design is intended to be easily integrated with future robotics applications by outputting a row offset which can be used for steering correction.

2. Methods

The overall design is depicted in Fig. 3 and details of each step are given in the following subsections. The numerical parameters used in standard image processing steps are given in the text. The values of these parameters are based on those used in the previous work, where they are quite standard across similar agriculture applications. Using these values as starting points, the given values were obtained after some manual optimisation based on a selection of sample images.

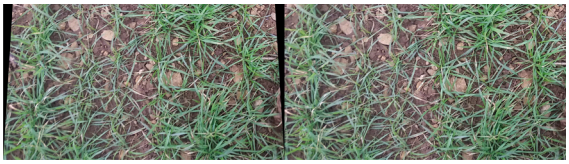


Fig. 5. Cropped ROI and rectification in crop field.

2.1. Image Preprocessing

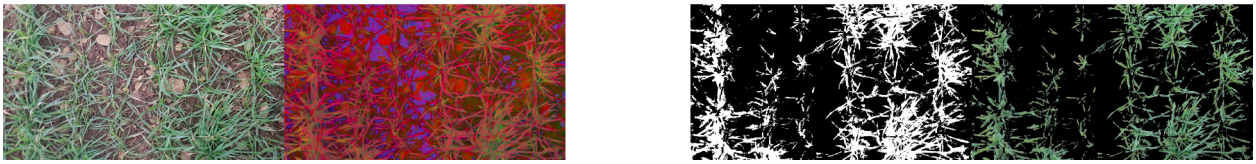
A preprocessing stage comprising standard image processing operations as previously used in the literature is first applied to separate green pixels (crops and weeds) from the rest of the image (soil, stones and other unexpected objects). It aims to reduce inherent noise from the camera sensor and blurring from its motions, and environmental effects including variable illumination due to weather conditions and time of day. The image processing pipeline is shown in Fig. 4 (showing a "drill-down" of the Preprocessing step of Fig. 3).

Image Parsing. Individual image frames are extracted from the video for sequential processing, as in Fig.

ROI Selection. When a vehicle is following rows in the center of a field, the whole image will be filled with crop rows. But when it is near the edge of the field or crop, there will be distractors at the edges of the images. Even when the camera is angled down, it is also sometimes possible for the horizon and sky to appear when driving uphill. We therefore crop the image to a 320 × 240 pixel region of interest in front and center, corresponding to around 1 m × 0.5 m of ground, and 3–5 crop rows, depending on the camera motion and row spacing.

De-warping. Perspective de-warping is used to transform the images into a top-down view. In the default camera perspective, objects further away from the camera appear smaller and the row lines appear to converge the further they are from the camera, which is not a true representation of the real world. The perspective transforms the perspective of the image as vertical. The right image in Fig. 5right) illustrates this transformation.

Sharpening. Image warping may introduce blur effects (e.g. Fig. 5right)). Image sharpening and filtering is applied to compensate for this blurring. ($cv2.filter2D : kernel = [-1, -1, -1], [-1, 9, -1], [-1, -1, -1]$]).



(a) RGB and HSV Transformation (b) Image in green domain Masking

Fig. 6. Transformation between RGB and HSV Space in crop field.

(a) Input Iimage (b) Thresholded image

Fig. 7. Adaptive thresholding with Gaussian kernel.

Colour transformation. RGB images are converted to HSV colour space as shownn in Fig. 6. RGB colour space is sensitive to light intensity which may change due to time of year, clouds and time of day. In our case crop rows are dominated by the colour green. HSV makes segmentation by pixel-wise thresholding into green vs non green areas more resilient to these factors ($[20, 25, 25] < RGB < [100, 255, 255]$). This operation provides a mask-like gray scale image as in Fig. 6left). Those masks have good matches for plant areas as in Fig. 6right).

Contrast enhancement. Histogram normalization (from OpenCV) is next applied. This compensates for some effects of overall lighting intensity.

Binarization. In order to further normalize different lighting and contrast conditions, image binarization is applied via an adaptive thresholding which varies depending on the characteristics of its neighborhood pixels ($cv2.adaptiveThreshold : maxval = 255, ADAPTIVE\_THRESH\_GAUSSIAN\_C, blcksiz = 5, bordsize = 2$). Different thresholds are obtained for different regions of the same image which gives better results for images with varying illumination for color based object detection. The threshold value is calculated by Gaussian-weighted sum of the neighborhood, an example is given in Fig. 7, using 5×5 kernel size.

Noise filtering/ Edge keeping. A denoise step is applied with bilateral filtering (Tomasi and Manduchi, 1998). This replaces the intensity of each pixel with a weighted average of neighborhood pixel values, removing unwanted noise but preserving features such as edges. ($cv2.bilateralFilter : dims = 5, sgmcolor = 75, sgmspace = 75$)).

Region filling. Erosion and dilation are applied to remove undesirable small green and non-green regions arisen as side effects in previous steps. ($cv2.morphologyEx : MORPH\_OPEN, kernel = \mathbf{1}_{5\times5}$).

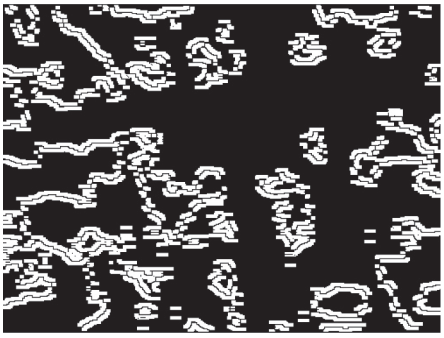
2.2. Feature detection

A major problem faced in this domain is how to tackle leaves that cross over between rows so as to isolate and identify individual lanes from each other. We target features based on the observation that cereal crops, being grasses, have long, narrowed ellipsoidal leaves, with leaves crossing between rows tending to have horizontal orientations. The solution is then to eliminate line features generated from horizontal leave edges, and only select those forward facing 'beam-form' ones of crop/grass rows, therefore to overcome the ambiguity in multiple rows separation.

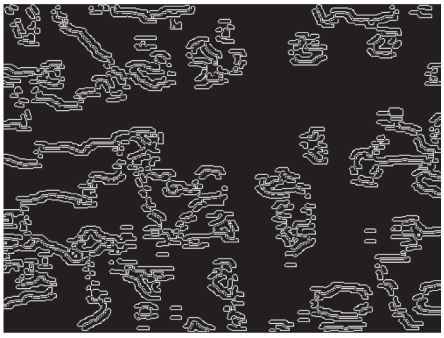
To achieve this a feature extraction module is added before the lane formulating, i.e. in color space, we can conduct the directional feature extraction using as 'X-gradient' feature by Sobel filtering before further applying the edge feature detection with canny edge detector as the image from crop/grass field has presented very rich ominous features. In the spatial domain, we use a Hough transform to produce oriented line segments. We then apply the directional line segments selection, where each line segment's slope is calculated, only restricted directional slope selected, so the favorable line segments (pure forward facing) can be

(a) Vertical gradient image produced by Sobel filter (b) Vertical edge feature image produced by Canny filter on (a)

Fig. 8. Vertical Sobel map and edge feature image.

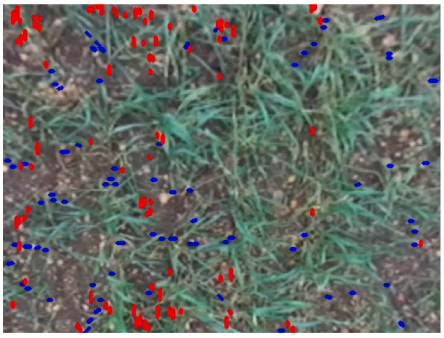


(a) Horizontal gradient image produced by Sobel filter

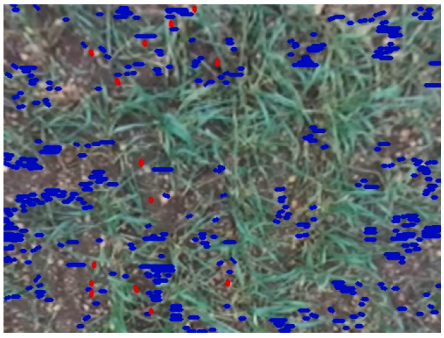


(b) Horizontal edge feature image produced by Canny filter on (a)

Fig. 9. Horizontal Sobel map and edge feature image.



(a) Hough local line segments features from the vertical Sobel-Canny features



(b) Hough line segments features from the horizontal Sobel-Canny features.

Fig. 10. Hough line segment features extracted from the Sobel-Canny outputs (blue = horizontal line segment features, red = vertical line segment features). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

therefore obtained.

2.2.1. Edge feature extraction

A Sobel filter (Duda and Hart, 2000) is used for feature selection. The Sobel filter has a natural directional selection when calculating gradients and is used to obtain gradients along vertical or horizontal image ($cv2.Sobel : kernel\_size = 5$).

A Canny edge detector (Canny, 1986) is then applied to the Sobel output to calculate gradients (derivatives), using non-maximum suppression to find the local maxima in order to obtain the 'thinned edge image' which is the key elements for the line extraction ($cv2.Canny : minthres = 50, maxtres = 200$).

Examples of gradients and edges are shown in Figs. 8 and 9.

2.2.2. Line segments feature extraction

We extract pixels that may belong to a crop row line. The crop row lines can be considered as noisy green lines over a non-green soil background. This is performed using a local, windowed version of the Hough transform. Hough transforms are usually applied to whole images to look for large-scale lines but are here applied instead to small windows to look for small localised lines which are considered to be features.

The probabilistic modification of the Hough transform (Kiryati and Bruckstein, 1991; Jiri Matas and Kittler, 2000) is used to enable it to run in real time ($cv2.HoughLinesP : \rho = 2, \theta = 1^{\circ}, threshold = 50, minLineLength = 4, maxLineGap = 4.$) In this modification, a set of n pixels from the edge of the image are selected for determining four parameters ($x0, y0, x1, y1$). This four-parameter vector contains the line extremes, by which the slope and intercept of each line segment can be calculated.

Fig. 10 shows the result of this processes applied to the gradient-edge images of Fig. 8 and Fig. 9. Vertical (a) and horizontal (b) line segments are clearly distributed differently, with the vertical segments tending to align with the crop rows centers as previously illustrated in Fig. 2.

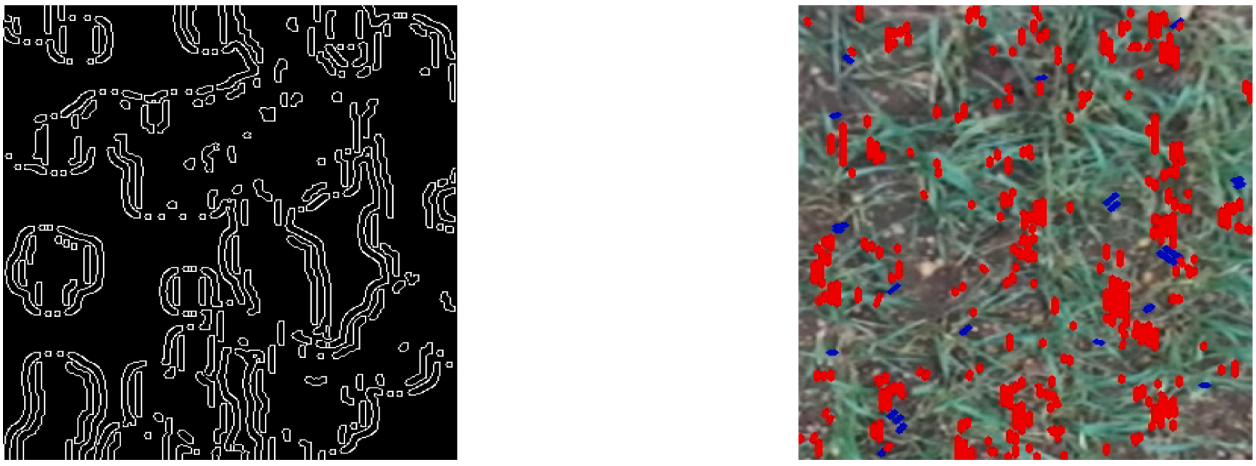
2.3. Line segments filtering

It can been seen that Hough line segments extracted on gradient-edge features as shown in Fig. 10 are still not perfect 'pure' in one-direction, i.e. Hough lines (red) obtained from vertical Sobel-Canny feature mixed with certain amount horizontal lines (blue), same with Hough lines (blue) obtained from horizontal Sobel-Canny feature, which still have a number of vertical ones (red) remained, even omni-directional features are indeed largely filtered out under the directional edge features extracted by Sobel-Canny strategy. It is worth pointed out that, in the ideal case, the edges extracted can be narrowed down by more specific directional gradient, which helps refining lines detection in the case of horizontal crop canopy occluded crossover between neighboring rows, but the Hough line formed from edge features is still not perfect directionally restricted as expected. This can be regarded as specific noise added in, the further processing is needed, i.e. line segments filtering is brought in, and which is based on the first key concept that the regions of crop rows tend to have local Hough line features oriented in the same direction as the rows, while leaves overlapping between rows tend to be horizontally oriented.

As the most forward-facing Hough lines obtained in the line detection module intend to carry potential information of each individual lane, so directional line segments can be selected to form forward-facing rows. To achieve this we calculate the slope of each line segment, and only select line segments with angles between 60° and 120° as lane line

(a) Edge Features (b) Hough Lines

Fig. 11. Edge Features (a) and filtered Hough lines (b) (in red) from weedless wheat crop image. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)



(a) Edge features (b) Hough lines

Fig. 12. Edge features (a) and filtered Hough lines (b) (in red) from wheat with blackgrass image. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

candidates.

Figs. 11 and 12 demonstrates the edge features (a) and lines detection (b) projected onto the visible image using the corresponding Hough transform. For clarity, the lines are drawn on top of wheat crop image, where the restricted Hough lines are in red.

2.4. Line segments clustering

2.4.1. Crop row modeling

A crop row may be viewed as composed of many image features distributed according to some statistical distribution. Those features are here assumed to be independent of each other. Given the distribution of $P(X|R)$ of feature points $X = \{x1, x2, \ldots, xn\}$ and known row region constraints for $P(R)$ as prior knowledge, we may apply Bayes' rule to obtain $P(R|X)$, over the lane line positions.

2.4.2. Clustering

Mean Shift (Keinosuke and Hostetler, 1975) is an unsupervised clustering method which using kernel density estimation (KDE) (Parzen, 1962; Sheather, 2004) which seeks to model the probability distribution of a data set. Coupled with Bayesian modeling, Mean Shift estimates the underlying probability density function (PDF) using a mixture Gaussian component in a non-parametric way. In general, placing a Gaussian kernel on each point in the data set, Mean Shift adds the individual kernels all together to generates a probability density function. Here, the different kernel bandwidth chosen will have result vary. It is categorized as non-parametric algorithm, the only parameter being used is the *bandwidth*, which smooths PDF (e.g a lower bandwidth can result more number of clusters).

Mean Shift is applied to each feature point iteratively to shift it in the direction of the local maxima specified by the kernel until being converged, i.e. the algorithm ends when all points have reached the corresponding maxima of the underlying distribution estimated by the chosen kernel. The set of points shifted to a certain local maximum is identified as a cluster.

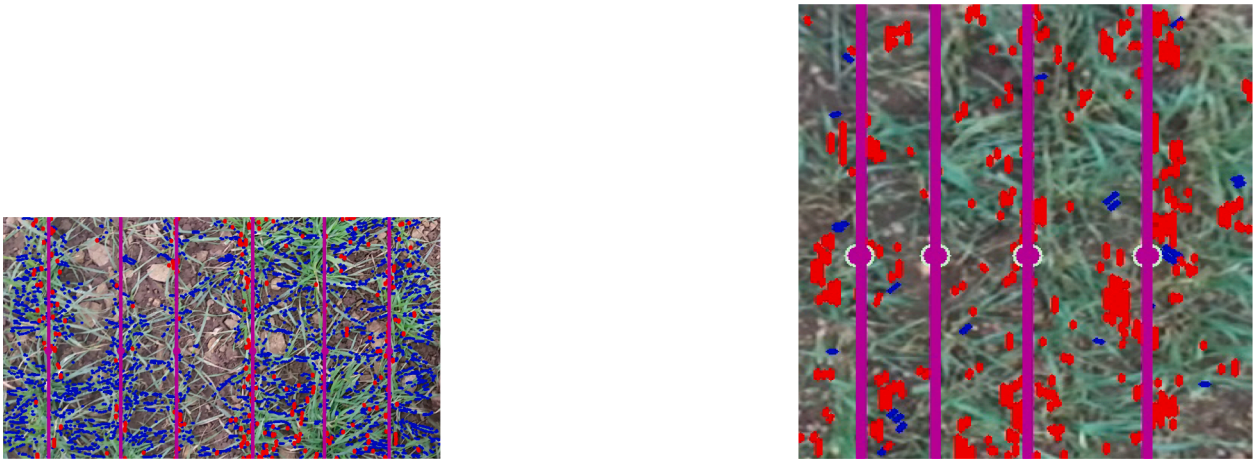
With the crop row aligned (forward facing) line segments are selected, upon the calculated centroids as feature points for those line segments, in this study, Mean Shift is applied to cluster the feature points into various categories corresponding to the crop row positions, i.e. the regions with a lot of points together, more points together mean higher density, hence each line segment can be assigned to a corresponding crop row later.

The Mean Shift algorithm scans line candidates, searching for groups of line segments that are spaced within a range of crop row widths which are adapted as the only tuning parameters of kernel bandwidth, it can be optimized based on prior information on crop row spacing, canopy width and camera position, etc. such as bandwidth of 25–100 pixels correspond to physical 150–300 mm in our experimental study.

Having no constraints of re-assumed number of lanes available, Mean Shift provides an adaptivity to the variant terrain environment as there is no certainty of presumed number of consecutive crop rows in advance always presenting in each frame, i.e. lane broken or faded is quite often from time to time in the real farm.

2.5. Lane centroids

The output of the Mean Shift operation provides clusters and their centroids. Those cluster centroids can be regarded as crop row centroids, which are taken as hypothesis of horizontal lane separation.



(a) Rows detected on weedless crop (b) Rows detected on weedy crop

Fig. 13. Row detection in weedless and weedy crops.

2.6. Hypothesis optimization

Mean shift provides basic data cluster estimation. But the pre-set or estimated standard variance of clustering data sets can not be perfectly accurate, together with internal data noise they can cause errors in both the number of centroids estimated and their location. To refine these estimates we developed a recursive data sorting routine combined with lane bandwidth to eliminate close neighbor's ambiguity, i.e. centroids in the cluster are to be internally validated via the neighborhood closeness checking. For example, if the distance of two points is less than a threshold (e.g. if row width is scaled by 0.618, the Golden section), they will be merged into one by averaging those two values. These operation will continue recursively until the end.

2.7. Line segments association

When this lane centroid pattern is found, the centroid point is labeled as a lane marking. Using those cluster centroids, the nearest neighbor association principle is utilized to retrieve all line segment centroids associated with corresponding lane/rows. Each individual lane line cluster is grouped separately and stored in a data structure of dictionary.

2.8. Lines regression

By recalling the lane line segments (centroids) coordinates stored in the database using dictionary key, we then apply liner regression for the data fitting, which provides the estimation of slope and intercept from the straight Hough line segments. The individual lane is therefore to be identified with drawing accordingly. In used fitting model $yi = mjxi + bj, mj$ is the slope (i.e. direction) of the j^{th} crop row, bj is the intercept, j was identified in the nearest neighbor clustering step. Both mj and bj are fit to the set of line segments using least squares with yi as the position of line segment i on the y-axis, xi as the position of the i^{th} line segmentation on the x-axis, and bj is therefore the intercept for the j^{th} row line sitting on the y-axis.

2.9. Lane forming

Lanes forming represents the hypothesis of the lane positions and trajectory relative to the natural crop row in the farmland. It provides the offsets to the last frame, for use in visualisation and as a (future) autonomous robot control signal.

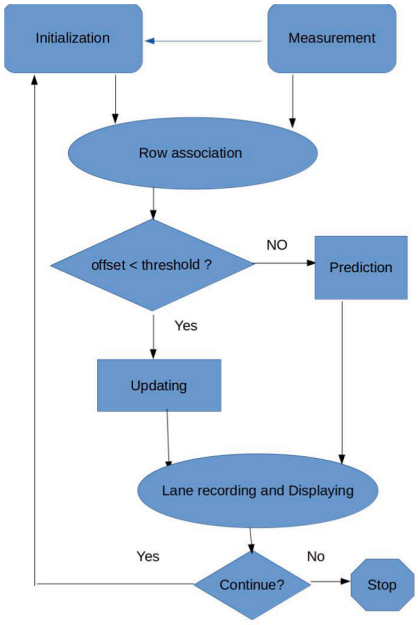


Fig. 14. Kalman filter tracking design.

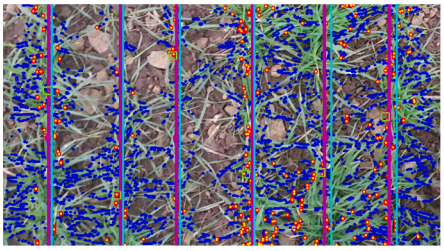
2.10. Multiple lane measurement

We store those clusters in a dictionary data structure associated with lane label for convenient search when needed. The lane median or center coordinate values can also be fused with corresponding row cluster in the dictionary, i.e. it is the actual correction or updating with previous cluster from mean shift when integrated with Kalman Filter (Musoff, 2000) later.

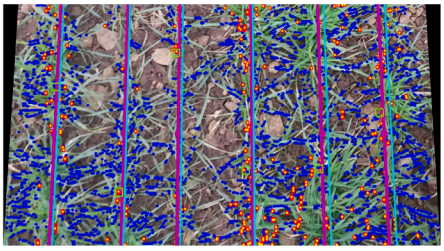
Finally we 're-warp' each image back into the original perspective view for display. Examples of lane detection with both crop and grass images are shown in Fig. 13, where (a) is the detection result from wheat crop field and (b) is for grass.

2.11. Tracking with Kalman filter

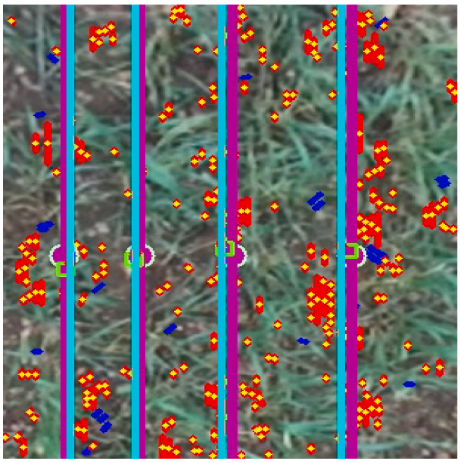
Crop rows are not always measurable due to the breaks in the crop rows which can be caused by sudden changes in soil conditions, mechanical inconsistency in the crop drilling machine and pest damage. We use standard Kalman Filters to track the centroid of each row in a video



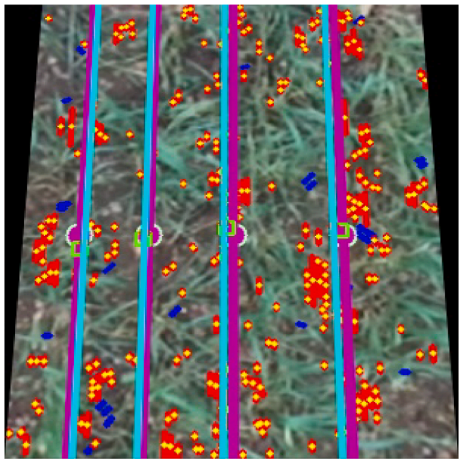
(a) Row tracking on weedless crop



(b) Rows tracking on weedess crop in Original View



(c) Rowst racking on weedy crop



(d) Row tracking on weedy crop in original view

Fig. 15. Rows detection (Purple), tracking (Light Blue) and line segments marked with centroids (Yellow).

stream, that is robust to these changes.

The Kalman filter tracks lane centroids. Several copies of the Kalman filter are used, one to track each individual crop row. The centroid horizontal pixel value is used as tracking 'ID' which is being updated in each processing cycle. This is convenient in controlling and updating, as in Fig. 14. The Kalman Filter is initialised with parameters for the crop row spacing and FOV of camera, and the chosen number of lanes to be tracked. Each row is a standalone tracking instance.

In the Kalman Filter's correction stage, the measurements are integrated with prediction through system matrix (transition, control and measurement) to obtain the optimized gain and state residual. There is a crucial prior operation - data association, i.e. the existing target to match the corresponding new measurement. We use a notable strategy here: The lane label is represented by lane centroids subject to the nearest neighbor principle with golden section theorem (0.618 or its division) applied to the lane gap as threshold. Each time when the lane is updated, the label is also updated accordingly.

The method provides robust lane detection and prediction in the case of row discontinuity. Using both crop and grass image as input, an example is shown in Fig. 15.

3. Results

3.1. Data Set

New crop row data was collected with a ZED camera (www.stereolabs.com), with the algorithm run on NVidia Jetson TX2 embedded computer (www.nvidia.com), both mounted on a mobile, manually remote controlled, robot platform, shown in Fig. 17, driven along wheat rows at 5 km/h. The camera was mounted at a fixed orientation and set to a fixed zoom. It was difficult for the operator to strictly follow the narrow lane gaps (15 cm) in the crop. so the row positions drift during the video frames, as would be the case in an autonomous system which needs to continually make row corrections.

Table 1
Data description.

Set	Weather, Condition	Time	Heading
1	Sunny, Breezy, Soft	12:00	North
2	Overcast, Breezy,Uneven	10:00	East
3	Sunny, Breezy, Uneven	10:30	North
4	Overcast, Breezy, Uneven	13:00	South
5	Changing, Breezy, Muddy	14:00	West
6	Changing, Windy, Uneven	14:50	Southeast
7	Overcast, Windy, Bumping	15:45	Southwest
8	Overcast, Breezy, Uneven	13:30	West
9	Changing, Breezy, Bumping	12:30	Northwest
10	Changing, Breezy, Uneven	12:30	Southeast
11	Sunny, Breezy, Uneven	11:20	East
12	Changing, Windy, Bumping	10:40	East
13	Overcast, Breezy, Uneven	14:40	North
14	Overcast, Breezy, Bumping	15:00	South
15	Changing, Windy, Bumping	11:30	Southwest
16	Cloudy, Windy, Bumping	12:00	Northeast
17	Changing, Windy, Uneven	12:50	Northwest
18	Changing, Windy, Uneven	13:30	East
19	Changing, Windy, Bumping	12:00	East
20	Changing, Windy, Bumping	13:00	West

Twenty data sets were collected under varying weather, headings, times of day, and terrains, as listed in Table 1. These were selected to check for effects such as shadows and lighting conditions and motion. The data was manually annotated with ground truth row positions. Sizes of the rows in each set are shown in Table 2. Each data set is a complete drive along a single row length. (The number of frames in a data set can thus be calculated from the length, speed, and frame rate, for Example 100 m $=$ 12minutes $=$ 720 s $=$ 21600 images.).

Table 2
Average error in datasets.

Data Set	Row Length (m)	Row Width (mm)	Mean frame Error(mm)	Stdev frame error (mm)
Data Set 1	80	150	11.5	11.7
Data Set 2	75	300	22.2	21.5
Data Set 3	60	300	15.5	13. 6
Data Set 4	500	300	20.8	23.9
Data Set 5	340	300	33.6	31.8
Data Set 6	240	300	29.3	25.3
Data Set 7	140	150	31.6	18.5
Data Set 8	270	300	25.0	26.1
Data Set 9	150	180	25.7	17.8
Data Set 10	95	125	12.7	10.8
Data Set 11	100	300	34.7	26.7
Data Set 12	350	300	53.4	34.5
Data Set 13	360	300	28.0	23.2
Data Set 14	400	300	37.2	33.1
Data Set 15	100	150	28.8	16.0
Data Set 16	150	150	27.9	16.8
Data Set 17	110	125	16.0	11.9
Data Set 18	130	125	20.0	12.7
Data Set 19	150	300	40.2	30.7
Data Set 20	140	300	37.6	28.1

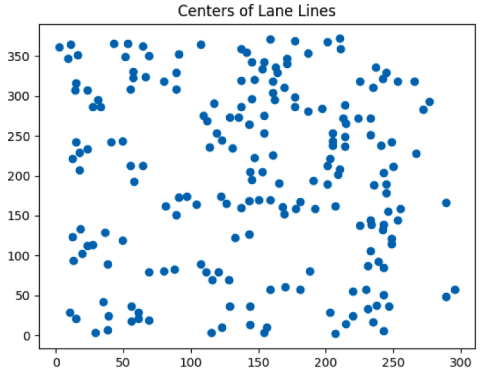
3.2. Baseline methods

Initial pilot experiments using simple re-implementations of conventional image processing methods as reviewed above, such as blob-detection, edge-detection, and segmentation (from the OpenCV library) all failed in this environment.

We then performed an ablation baseline by using only the first four stages of the pipeline in Fig. 3: pre-processing, features detection, edge extraction, and line segments. Fig. 16(a) shows a typical failure case from this configuration. The line segments are scattered randomly across the image, without any constraints it is impossible to form the directional lanes. Because the row consists of many individual plants, there is no single object with regular rigid structure. The Hough transform is

Fig. 17. Robot data collection platform.

(a) Lane lines detection



(b) Centers Distribution Line Segments

Fig. 16. Baseline Methods on RGB Image.

(a) Data Set 1

(b) Data Set 2

(c) Data Set 3

(d) Data Set 4

(e) Data Set 5

(f) Data Set 6

(g) Data Set 7

(h) Data Set 8

(i) Data Set 9

(j) Data Set 10

(k) Data Set 11

(l) Data Set 12

Fig. 18. Sample images from the datasets.


(a) Data Set 13


(b) Data Set 14


(c) Data Set 15


(d) Data Set 16


(e) Data Set 17


(f) Data Set 18


(g) Data Set 19


(h) Data Set 20

Fig. 19. Sample images from the datasets.

unable to provide line segments along the proposed rows.

Further illustration of the failure is given by the corresponding distribution of centers for line segments in Fig. 16(b). The center points appear to be randomly scattered across the ROI.

Apart from the common problems encountered in outdoor image acquisition is the unevenness of the lighting conditions across both time and space, variations of the sun elevation contribute to changes in the global illumination of the scene as observed by the camera, the critical domain-specific problem was found to be that the canopy of crops can be very occluded in fields, therefore the crop images can be difficult to segment using only outline features using traditional processing methods which normally work well on canonical image detection or classification. (See Fig. 17).

3.3. Full proposed method

The full proposed method was then tested. The results depicted here are based on our proposed algorithms and pipeline working on output of camera in 720HD and 30fps RGB model. Typical images extracted from the camera for each set are shown in Figs. 18, 19.

3.3.1. Examples of per-frame results with crops

Wheat crop images at this growing stage have small leaves in some fields, which posed another challenge in applying our canopy feature based method. Fig. 20 shows a few example frames' results on the real-time recorded videos from each of the first five sets. The yellow lines here are the estimation for multiple rows through our system. The average time cost of running through single frame is below 0.05s on an Intel i7 3.1 GHz 64bit CPU.

3.3.2. Statistical analysis on large datasets

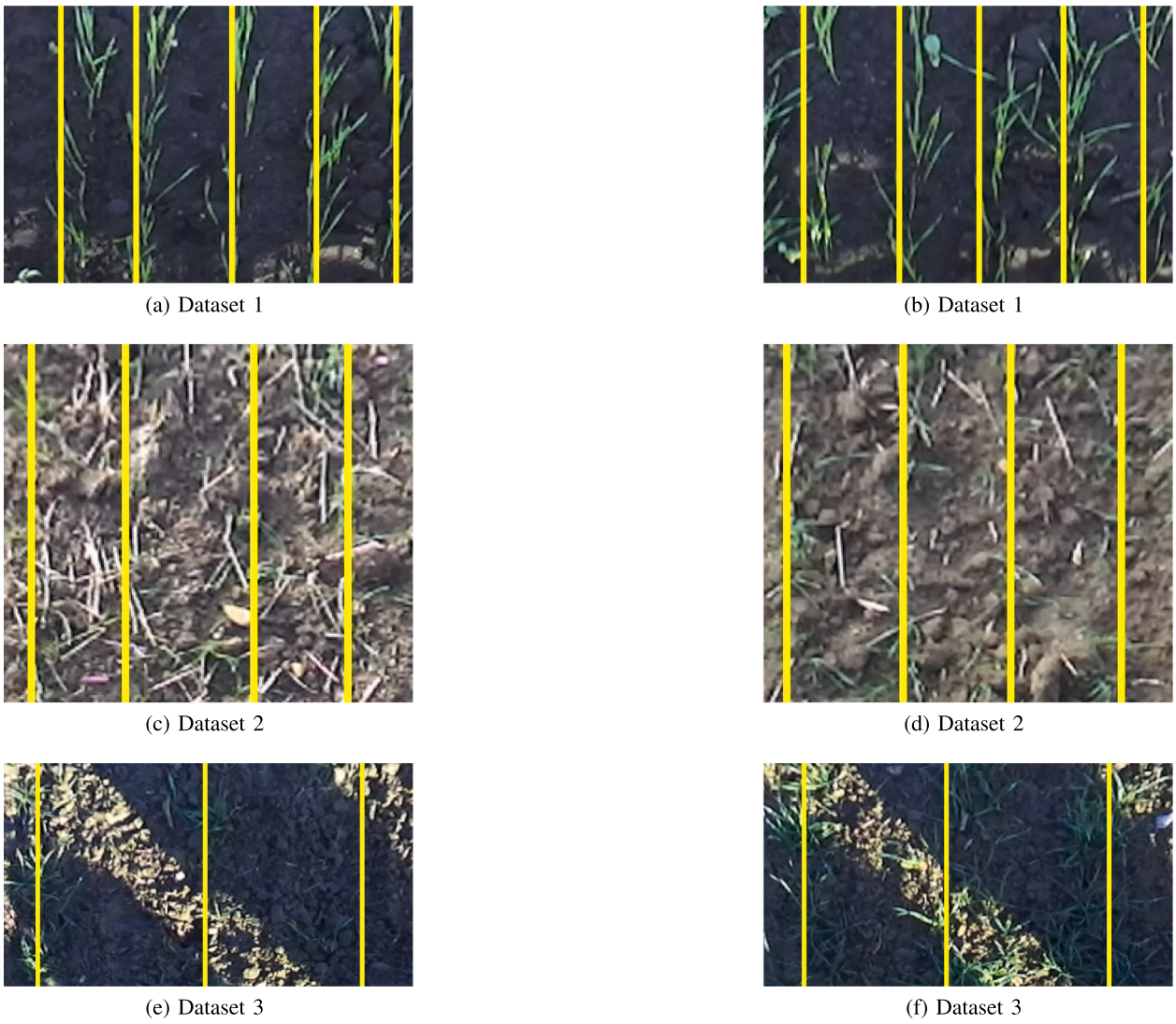
For each data set, a *frame error* was computed for each of its frames f,

$$ef = \frac{1}{L(f)} \sum j = 1^{L(f)} |lj(f) - \widehat{l}j(f)|,$$

where $L(f)$ is the number of ground truth lanes visible in frame f, $lj(f)$ is the horizontal location of the jth lane in frame f, and $\widehat{l}j(f)$ is the estimate of $lj(f)$ output by the algorithm. The frame error measures the average distance of an estimated lane from its ground truth location in the frame. This is the millimeters distance from the ground truth row centers to the estimated ones during each data set. If the line-following method was used to control a vehicle, it would be the number of millimeters which its wheels would be expected to drift from the true row center. The means and standard deviations of these frame errors, over each data set, are shown in Table 2.

4. Discussion

The new method estimates row centers to accuracies suitable for the control of a wheeled vehicle aiming to drive in between rows in some farming systems. Wheat is typically drilled at spacings from 125–180



(a) Dataset 1 (b) Dataset 1

(c) Dataset 2 (d) Dataset 2

(e) Dataset 3 (f) Dataset 3

Fig. 20. Lane tracking examples from datasets 1–3.

mm, and the worst case average error is about 50 mm. To determine if a wheeled vehicle would run over crops, the combined width of a wheel and a crop row must be considered. The main use case is for early growth crops where the crop width is small so the wheel width is the main consideration. For wheat drilled at 180 mm spacing, the 50 mm error would avoid crop damage if the combined wheel and crop width were 80 mm of less. For wheat drilled at 150 mm spacing, the combined width would need to be 50 mm. For wheat drilled at 125 mm spacing, the combined width would need to be 25 mm. Typical agri-robot wheel widths are around 80–100 mm width, so it is possible for them to operate using the algorithm in some of the wider spaced crops, though not in the narrowest (125 mm) crops. Farming systems wanting to make use of the algorithm and similar robots should take such calculations into account when planning what spacing to use, and consider the economic trade-offs involved.

The above calculations are based on the mean frame errors, but the standard deviations of frame errors show that there is variability of performance within the data sets, which may result in damage so some portion of crop even when the mean error is within the wheel width limits. These values may be used to derive more detailed agri-systems tradeoffs if required.

The results show that the method is able to identify crop rows even under circumstances of highly occluded leaves crossing over between adjacent rows *inter-row canopy closure*. It can detect any number of crop lines in the farm fields, the row line classification works reasonably well. It works with either high or low image resolutions and is able to operate in real time on relatively low cost hardware.

System performance was robust to shadows, as long as there was enough features visible in the image. This is seen because there is no large degradation in performance in the datasets taken driving away from the sun in sunny conditions where shadows are visible in every frame (data sets 1 and 3). Although shadow can deteriorate certainty of brightness and color balancing— a major problem in outdoor scenes – by adjusting the position and field of view of camera, it can be largely overcome as in above data set 1 and 3, which show even better results when the camera was lower to ground.

There is no evidence for effects on performance from row spacing, which is surprising as it might be thought that narrower rows are harder to track, having more times when overlaps occur. A possible explanation is that unlike other methods, the algorithm does not rely on finding brown soil between rows at all – rather it works entirely using leaf directions which are formed into rows, and this remains equally informative as spacing changes.

As is common in machine vision, many parameters were used in the basic image processing steps. The values used were based on those from previous studies with some manual optimisation. It is possible that some performance gains might be obtained by a larger automated parameter optimisation, though we note that most previous work consistently uses

similar values which suggests that such gains would be minor.

Future work is needed to understand how the method could be integrated into active control of autonomous vehicles to keep them in-row, and the feedback effects of deviations from physical row driving on the input data, as this was performed manually in the present study. The tests assume that the robot drives approximately perpendicialy to the rows at all times, and the method in its present state would fail if the robot became oriented very differently from this. A simple way for practical robots to correct for this would be to retain GNSS paths from the seeding tractor, and use them to correct the physical orientation if it becomes lost in this sense. The method has only been tested here in the use case of straight rows, as is common for wheat crops. It is likely to hold up for gentle curves, i.e. those which cannot be distinguished from straight lines in the limited, oriented, view of the camera, because these present exactly the same task as actual straight lines. Further work could test this empirically and develop new extensions to handle tighter curves.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Canny, J., 1986. A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell., PAMI-8, Nov 1986.

Duckett, T. et al., 2018. Agricultural robotics: The future of robotic agriculture. UK-RAS Network White Papers, Jun 2018.

Duda, R., Hart, P., 2000. Sobel-feldman operator. Pattern Classif. Scene Anal. 271–272.

García-Santillán, I., Guerreo, J.M., Montalvo, M., Pajares, G., 2017. Curved and straight crop row detection by accumulation of green pixels from images in maize fields. Precision Agric. 19, 01.

Hassanein, M., Khedr, M., El-Sheimy, N., 2019. Crop row detection procedure using low-cost uav imagery system. Int. Arch. Photogram., Remote Sens. Spatial Inform. Sci.

Hubbard, C.E., 1984. Grasses: A guide to their structure, identification, uses and distribution in the british isles by charles edward hubbard. Penguin Books Ltd.

Jiri Matas, C.G., Kittler, J., 2000. Robust detection of lines using the progressive probabilistic hough transform. Comput. Vis. Image Underst. 78, 119–137.

Keinosuke, F., Hostetler, L., 1975. The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Trans. Inform. Theory, Jan 1975.

Khalilikhah, M., Balali, V., Heaslip, K., 2016. Using stationary image based data collection method for evaluation of traffic sign condition. Int. J. Transport. Sci. Technol. 5 (4), 248–256.

Kiryati, Yen, Bruckstein, A., 1991. A probabilistic hough transform. Pattern Recogn. 24, 303–316.

Latha, P., Amarnath, A.B.V., Kumar, G.V., 2014. Image processing in agriculture. Int. J. Innovat. Res. Electr., Electron., Instrum. Control Eng. 2 (11), 1475–1490.

Lee, C., Moon, J.H., 2018. Tracking for real-time applications. IEEE Trans. Intell. Transp. Syst. 99, 1–6.

Li, Q., Chen, L., Li, M., Shaw, S.-L., Nüchter, A., 2013. A sensor-fusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios. IEEE Trans. Veh. Technol. 63 (2), 540–555.

Mingfa Li, Y.L., Jiang, M., 2018. Lane detection based on connection of various feature extraction methods. Adv. Multimedia.

Montalvo, M., Pajares, G., Guerrero, J., Romeo, J., Guijarro, M., Ribeiro, A., Ruz, J., Cruz, J., November 2012. Automatic detection of crop rows in maize fields with high weeds pressure. Expert Syst. Appl. 39 (11), 11889–11897.

Moss, S., 2017. Black-grass (alopecurus myosuroides): why has this weed become such a problem in western europe and what are the solutions? Outlooks on Pest Management, 28.5:207–212.

Musoff, P.Z.H., 2000. Fundamentals of Kalman filtering: A practical approach. American Institute of Aeronautics and Astronautics, Incorporated, 2000.

Ozgunalp, X.A.U., Fan, R., Dahnoun, N., 2017. Multiple lane detection algorithm based on novel dense vanishing point estimation. IEEE Trans. Intell. Transp. Syst. 18, 621–632.

Ozgunalp, X.A.U., Fan, R., Dahnoun, N., 2017. Multiple lane detection algorithm based on novel dense vanishing point estimation. IEEE Trans. Intell. Transp. Syst. 18, 621–632.

Parzen, E., 1962. On estimation of a probability density function and mode. Ann. Math. Stat. 33, 1065–1076.

Piao, J., Shin, H., 2017. Robust hypothesis generation method using binary blob analysis for multi-lane detection. IET Image Proc. 11, 1210–1218.

Ramesh, K., Chandrika, N., Omkar, S., Meenavathi, M., Rekha, V., 2016. Detection of rows in agricultural crop images acquired by remote sensing from a uav. Int. J. Image, Graph. Signal Process. 8 (11), 25.

Md. Rezwanul Haque, M.M.I., Kazi Saeed Alam, H.I., Shaik, M.E., 2019. A computer vision based lane detection approach. Int. J. Image, Graph. Signal Process., March 2019.

Romeo, J., Pajares, G., Montalvo, M., Guerrero, J., Guijarro, M., Ribeiro, A., 2012. Crop row detection in maize fields inspired on the human visual perception. Scient. World J. 2012.

Sharma, A., Kumar, V., Shahzad, B., Tanveer, M., Sidhu, G.P.S., Handa, N., Kohli, S.K., Yadav, P., Bali, A.S., Parihar, R.D., et al., 2019. Worldwide pesticide usage and its impacts on ecosystem. SN Appl. Sci. 1 (11), 1446.

Sheather, S.J., 2004. Density estimation. Stat. Sci. 19, 588–597.

Shewry, S., Hey, P.R., 2015. The contribution of wheat to human diet and health. Food and Energy Security.

Song, W., Yang, Y., Fu, M., Li, Y., Wang, M., 2018. Lane detection and classification for forward collision warning system based on stereo vision. IEEE Sens. J. 18 (12), 5151–5163.

Tomasi, C., Manduchi, R., 1998. Bilateral filtering for gray and color images. In Sixth international conference on computer vision (IEEE Cat. No. 98CH36271), pages 839–846. IEEE, 1998.

Winterhalter, W., Fleckenstein, F.V., Dornhege, C., Burgard, W., 2018. Crop row detection on tiny plants with the pattern hough transform. IEEE Robot. Autom. Lett. 3 (4), 3394–3401.

Yoo, J., Lee, S., Park, S., Kim, D., Feb 2017. A robust lane detection method based on vanishing point estimation using the relevance of line segments. IEEE Trans. Intell. Transp. Syst. 18, 3254–3266.

Zhang, W., 2018. Global pesticide use: Profile, trend, cost/benefit and more. Proc. Int. Acad. Ecol. Environ. Sci., 8(1):1.

Zhang, X., Li, X., Zhang, B., Zhou, J., Tian, G., Xiong, Y., Gu, B., 2018. Automated robust crop-row detection in maize fields based on position clustering algorithm and shortest path method. Comput. Electron. Agric. 154, 165–175.