

FAST Keypoint detection on excess green images for navigation line extraction in maize fields

Niramon Ruangpayoongsak* and Thanpisit Rangdeang

Production and Robotics Engineering Department, Faculty of Engineering, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand

Received 29 December 2025

Revised 27 May 2026

Accepted 9 July 2026

Abstract

Maize field interrow navigation requires high precision for mobile robot heading control because of the narrow soil area between maize rows. This paper presents a new method for extracting a navigation line in the interrow space of maize fields of an automatic granule fertilizer dispenser (AGFD) robot. The proposed algorithm is a crop row detection approach named FAST keypoint detection, which is applied to the excess green image. The FAST keypoints detected are distributed throughout the crop rows in the image. The crop row lines are fitted by the least squares method. Afterward, the navigation line is extracted. The initial region of interest (ROI) is automatically identified using YOLO-based soil segmentation and the adaptive boundary lines are generated. The RMSE and R^2 value of the navigation lines obtained from the proposed method is compared with that of the excess red- and excess green-based methods. The proposed method achieved a minimum RMSE of 10.06 pixels and a maximum R^2 value of 0.75. The results reveal that the FAST keypoint detection method on the excess green image outperforms the benchmarks methods for various stages of maize growth with respect to both higher accuracy and less computation time. The proposed method achieved a maximum RMSE improvement of 10 pixels (43.74%), reducing the error from 22.86 pixels, while requiring a computational time of only 53 ms, corresponding to 17% of that of the benchmark method. The results of the proposed method under various field conditions, including differences in illumination, soil color, and vegetation density, are also presented. The proposed method is capable of real-time implementation on the AGFD robot.

Keywords: Excess red, Crop row detection, YOLO, Interrow navigation, Adaptive ROI

1. Introduction

Maize has attracted significant interest because of its important role in global food security and economic stability. Recent studies [1] have reviewed machine learning and computer vision applications in precision agriculture, and maize is the most extensively explored crop. Crop row navigation for mobile robots in agricultural fields is a critical research topic. Due to the narrow interrow spacing in maize cultivation, navigation line extraction becomes a crucial step to ensure precise and reliable robot control.

Many researchers have recently proposed various methods for crop row navigation. RGB images and lidar have been used for navigation line extraction. Path tracking control of autonomous agricultural mobile robots was presented in [2]. The majority of the work addressed the early growth stage. Real-time image processing for crop/weed discrimination in maize fields was proposed in [3]. Crop-row detection algorithms have been developed via the Hough transform [4-6]. The development of the Hough transform involves accelerating the computation time and overcoming multiple crop rows and weed conditions.

Multiple crop rows were detected via multiple regions of interest (multiple ROIs) and Otsu's method [7]. After the second thresholding, the binarized image contained fewer weeds, and the templates identified areas where crop rows were expected. However, the method was applied to various crops without weeds in the field. Crop rows with high weed pressure were detected with templates [8]. Even though the template is highly effective at screening out the weeds, it is not suitable for ground robot navigation because the orientation of the crop row varies.

Researchers have used different FOVs of cameras to detect crop row lines and navigation systems of different maize sizes and row shapes. A single crop row is detected for a crop inspection robot driving over the middle crop row of a small maize in an image [9]. Small vehicles can be used to detect large maize plants with obvious stems [10]. Front-view images are used to extract navigation lines [11] when the stem is detectable. The maize stems could locate the position of each maize in the front-view image, but they are not visible in the top-view image. Soil area detection of the front view was used to search for leading points to control the heading of a robot for a medium-sized maize [12]. Navigation line extraction from the front-view image is appropriate for medium-sized to large maize plants.

The top-view images of different growth stages were further investigated. A feature point is extracted via multiple steps to extract the navigation line of a tiny maize plant [13]. The excess green (ExG) color index and Otsu's method were presented for crop row segmentation on the basis of adaptive multiple -ROIs of different crop row shapes [14]. Even though the methods were effective, the computation time needed to be reduced.

Recently, researchers have improved algorithms to extract navigation lines and detect crop rows via visual and lidar data [15, 16]. This method can detect crop rows under dense weed conditions but requires the collection of large datasets to train AI. Moreover, low-

*Corresponding author.

Email address: niramon.r@eng.kmutnb.ac.th

<https://doi.org/10.64960/easr.2026.265472>

cost vision-based crop row detection was further investigated. These methods were developed via edge detection and Otsu thresholding for broken crop rows [17]. However, only the middle crop row in the image was detected. In the other works, the horizontal strip was divided into the ROI to obtain feature points, and the Hough transform with guidance lines [18], and lightweight network [19] were applied to detect crop rows. In addition, a navigation line extraction method based on Multi-Mask Fusion was presented [20]. Unfortunately, the methods were applied only for early-stage or mature-stage crops.

Navigation line extraction across different maize growth stages requires further development to improve time efficiency, accuracy, and cost-effectiveness. Moreover, the application of FAST keypoint detection on excess green images has not yet been investigated. In this paper, a new method for maize navigation line extraction is developed on the basis of feature point detection of an automatic granule fertilizer dispenser (AGFD) robot [21] driving in an interrow space with a camera mounted on the top of the robot for various growth stages. The camera is placed 1.00-1.50 m (h) above ground and tilted down to 40-60 degrees (α), as shown in Figure 1. Early-stage maize fertilization is as shown in the figure. Fertilization occurs at various maize growth stages. The objectives of the research are:

- The proposed method is aimed at multiple growth stages and various field conditions.
- This work studies FAST keypoints on excess green images.

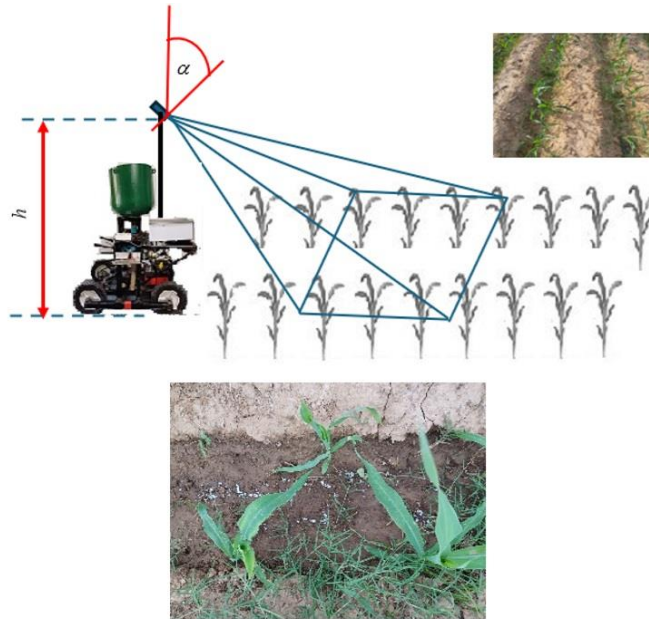


Figure 1 Interrow navigation of the AGFD robot in a maize field (top) and granule fertilizer dropped by hand (bottom)

2. Materials and methods

Figure 2 illustrates an overview of the proposed method. Two RGB images are used as inputs. The reference RGB image is selected as a representative frame for automatic ROI detection and adaptive boundary line generation. The second RGB image corresponds to the current frame and is used for navigation line extraction and adaptive boundary line generation. When no change in row orientation is detected, the template centerline and boundary lines are applied. In contrast, when an orientation change is identified, the updated centerline and boundary lines are used to ensure accurate navigation line extraction.

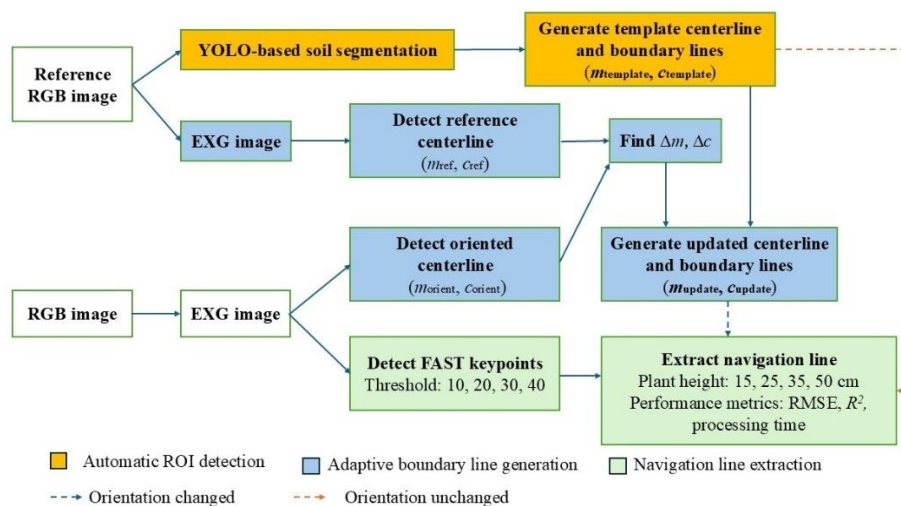


Figure 2 Overview of the proposed method

The details of each component are presented as follows. First, the generation of excess green images is described, followed by a comparison of FAST keypoint detection on grayscale and excess green images. Second, the automatic ROI detection procedure is introduced. Third, the adaptive boundary line generation method is presented. Subsequently, the navigation line extraction process is explained. Last, the influence of lighting conditions on FAST keypoint detection is then analyzed.

2.1 Excess green image generation

The RGB color model is used for reading an image captured by a camera. Afterward, the normalized r , g , and b components are determined as follows:

$$r = R/(R+G+B), g = G/(R+G+B), b = B/(R+G+B) \quad (1)$$

where R , G , and B represent the color components. The excess green image is an image generated by the following excess green index [22].

$$ExG = 1.8g - 1.1r - 0.8b \quad (2)$$

where r , g , and b represent the normalized values of the red, green, and blue components, respectively. The ExG values are plotted in Figure 3(b). The excess green index is effective for detecting crops under soil color and crop variety [7].

2.2 FAST keypoint detection on gray and excess green images

The FAST (features from accelerated segment test) algorithm is a corner detection method that can be used to extract feature points [23]. The threshold value of FAST can be adjusted. The gray and excess green images are shown in Figures 3(a) and 3(b), respectively. In the gray image, the tones of the color of the soil and crop are not easily distinguishable. On the other hand, in the excess green image, the color of the crops is gray and white, and the color of the soil is black. The FAST keypoints (red dots) are distributed throughout all the crops in the gray image but are distributed throughout only the crop in the excess green image because of the contrast between white and black, as shown in Figures 3(c) and 3(d). The FAST detection threshold values of 25 and 50 are shown in Figures 3(e) and 3(f), respectively. At a threshold value of 25, there are more keypoints than at a threshold value of 50. The higher the threshold value is, the smaller the number of keypoints detected.

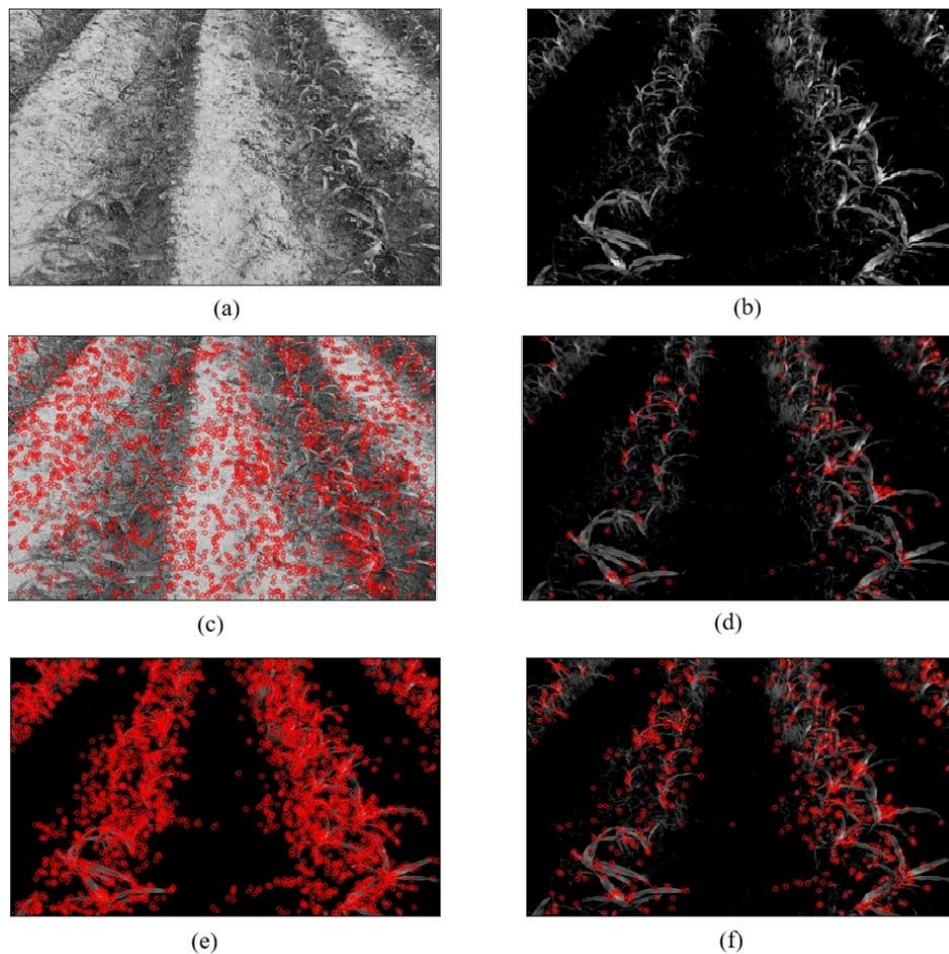


Figure 3 Processed images (a) Gray image, (b) excess green image, (c) FAST keypoints on the gray image, (d) FAST keypoints on the excess green image, (e) FAST keypoint detection (threshold = 25), and (f) FAST keypoint detection (threshold = 50)

2.3 Automatic region of Interest (ROI) detection

The ROI is selected because four rows of maize appear in the image, but only two rows are used to determine the navigation line. The boundary lines (green), the left and right rows of the fitted line (red), the generated centerline (blue), and the manually drawn centerline (yellow) obtained from the color image are clearly visible in Figure 4. The region of interest is below the green line because we focus only on the two maize rows in the middle. This boundary line must be adjusted in the late growth stage of maize because of the expansion of leaves covering the interrow space.

YOLO (You Only Look Once) has been widely adopted for perception tasks in maize field environments. In previous work, the road-segmentation module determined the robot's navigable space by using YOLOv5 to segment soil regions [24]. In this study, YOLOv8 is employed to perform soil segmentation, followed by contour detection and morphological operations to refine the extracted pathways.

Images from the Roboflow database [25] were used to train the YOLO11n-seg model. The dataset was partitioned into three subsets to support model training, hyperparameter tuning, and final evaluation: 267 training images (68.64%), 74 validation images (19.02%), and 48 testing images (12.34%). The model was trained for 100 epochs. The F1-confidence curve of the YOLO soil segmentation training and the prediction of the maize field image is shown in Figure 5.

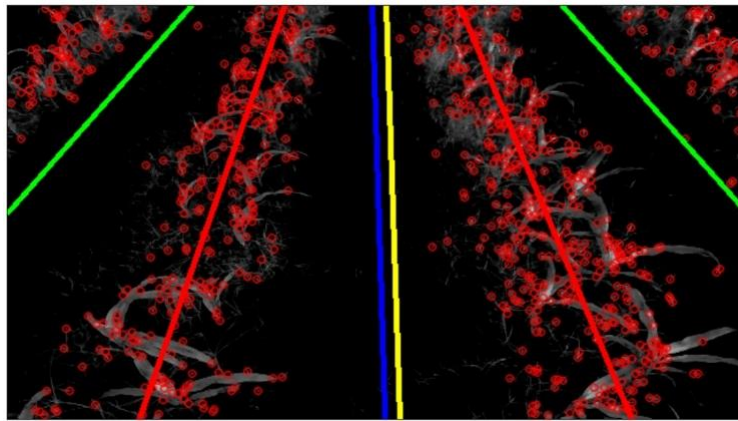


Figure 4 Boundary lines (green), the left and right rows of fitted lines (red), the generated centerline (blue), and the manually drawn centerline (yellow)

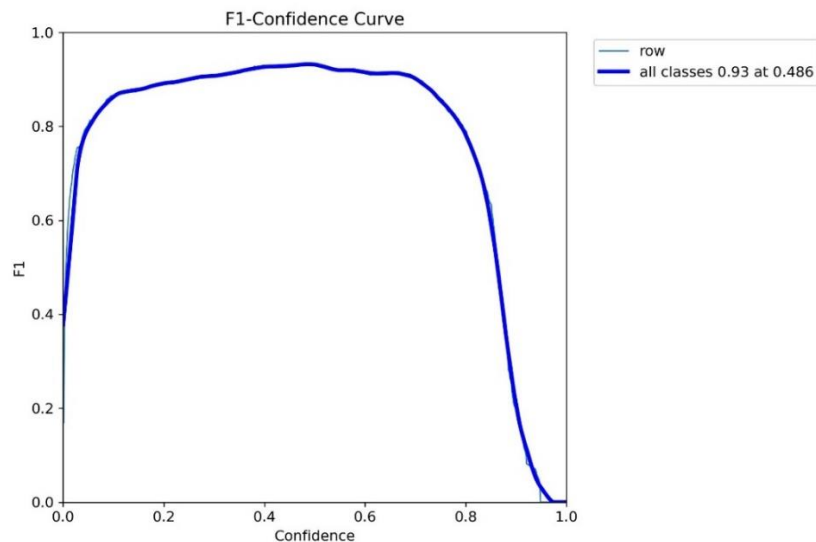


Figure 5 F1-confidence curve of YOLO segmentation

The original image is shown in Figure 6(a). The boundary line generation process begins with YOLO-based segmentation of the soil region, as illustrated in Figure 6(b). Multiple soil segments may be detected; therefore, the leftmost segment is selected (Figure 6(c)) to compute the centerline of the soil area (Figure 6(d)). A line-fitting procedure is then applied to generate the left boundary line (green), as presented in Figure 6(e). The same procedure is performed on the rightmost soil segment, as shown in Figures 6(f)–6(h). Finally, the centerline (red) is generated, as illustrated in Figure 6(i).

The initial ROI is defined once at the start of the operation and is subsequently used for robot navigation. Because the geometry of the soil area varies across growth stages, the corresponding ROI also changes. Owing to its strong detection performance, YOLO can accurately segment the soil area under different growth conditions. Moreover, even when the camera tilt angle or mounting height is adjusted, the soil region can still be reliably detected.

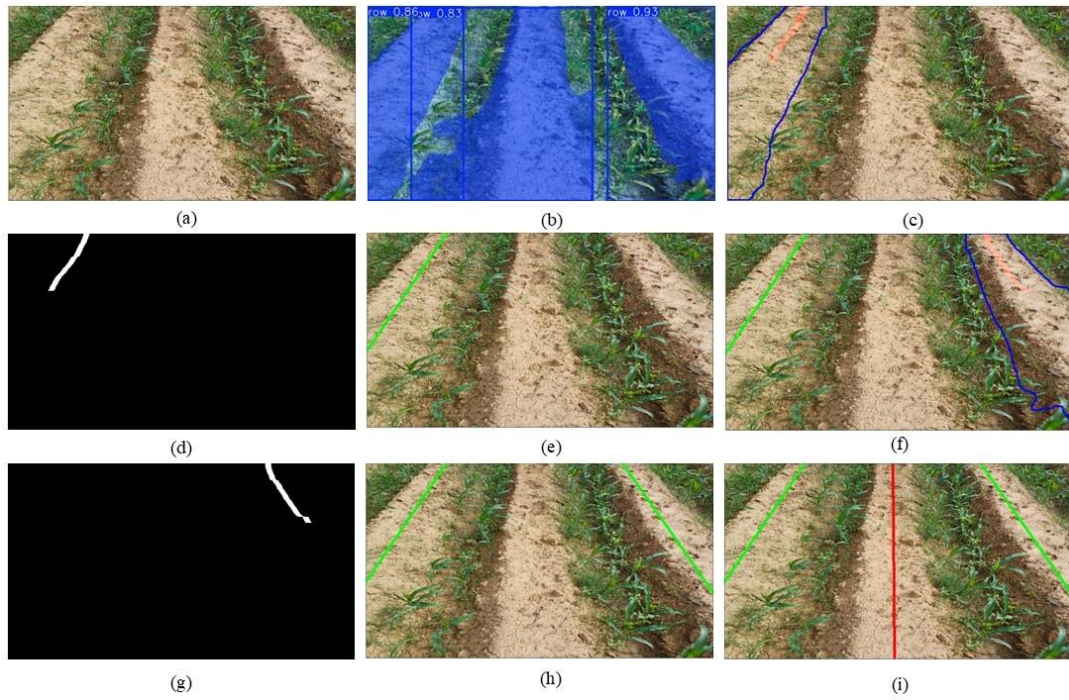


Figure 6 Boundary line generation process: (a) original image, (b) YOLO-based soil segmentation, (c) left soil segment, (d) centerline of the left soil segment, (e) left boundary line generation, (f) right soil segment, (g) centerline of the right soil segment, (h) right boundary line generation, and (i) centerline generation.

2.4 Adaptive boundary line generation

As the AGFD robot navigates through the interrow space, the apparent orientation of maize rows in the captured images can change. Despite these variations, the robot must consistently maintain a central position within the soil region between rows. This requirement necessitates adaptive boundary line generation, ensuring that the estimated boundaries remain aligned with the row orientation as it shifts.

As shown in Figure 7, the images on the left depict the binary outputs with the detected centerline (blue), while the images on the right present the corresponding color images with both the centerline (red) and boundary lines (green). The top row illustrates the reference frame, and the bottom row shows the oriented frame. In the color images, the centerline represents the midpoint between the two green boundary lines.

In the binary image, centerline detection is performed only in the upper-middle region of the frame, as leaf density is higher in the lower portion of the image. The centerline is extracted from a binary image generated by blurring the excess green image, followed by Otsu's thresholding, dilation, and erosion. The centerline is then estimated by selecting three representative points and applying line fitting using the following equations [14]:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i, \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i \quad (3)$$

$$b = \frac{\sum_{i=1}^N x_i y_i - N \bar{x} \bar{y}}{\sum_{i=1}^N x_i^2 - N \bar{x}^2}, \omega = \bar{y} - b \bar{x} \quad (4)$$

The centerline is depicted as a reference centerline (blue). When a new frame is acquired, the change in orientation is computed by comparing the slope and intercept of the centerline in the reference frame ($m_{\text{ref}}, c_{\text{ref}}$) with those of the centerline in the oriented frame ($m_{\text{orient}}, c_{\text{orient}}$).

$$\Delta m = m_{\text{orient}} - m_{\text{ref}} \quad (5)$$

$$\Delta c = c_{\text{orient}} - c_{\text{ref}} \quad (6)$$

The orientation change of the centerline in the binary image, expressed as ($\Delta m, \Delta c$), is used to adjust the centerline and boundary line templates obtained in the previous section ($m_{\text{template}}, c_{\text{template}}$). Applying these adjustments yields updated center and boundary lines ($m_{\text{update}}, c_{\text{update}}$) that compensate for the observed change in row orientation.

$$m_{\text{update}} = m_{\text{template}} + \Delta m B \quad (7)$$

$$c_{\text{update}} = c_{\text{template}} + \Delta c B \quad (8)$$

In Equations (7) and (8), the vectors m and c each comprise three components corresponding to the centerline and the left and right boundary lines, respectively. $m_{\text{template}} = [m_{\text{template},c}, m_{\text{template},l}, m_{\text{template},r}]^T$, $m_{\text{update}} = [m_{\text{update},c}, m_{\text{update},l}, m_{\text{update},r}]^T$, $c_{\text{template}} = [c_{\text{template},c}, c_{\text{template},l}, c_{\text{template},r}]^T$, $c_{\text{update}} = [c_{\text{update},c}, c_{\text{update},l}, c_{\text{update},r}]^T$, and $B = [1 \ 1 \ 1]^T$. The centerline (red) positioned is used to separate the FAST keypoints into left and right groups for navigation line extraction.

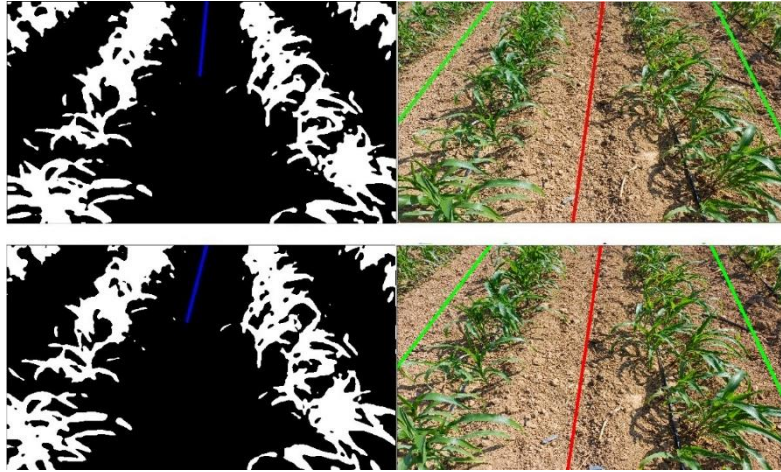


Figure 7 Adaptive boundary lines generation. Left: binary image showing the soil-based reference (top) and oriented (bottom) centerline (blue). Right: corresponding color image illustrating the template (top) and updated (bottom) centerline (red) together with the boundary lines (green).

2.5 Navigation line extraction

The keypoints in the ROI are divided into two groups: the left row and the right row. If the keypoints lie on the left half of the image and below the left boundary line, then the keypoints belong to the left row. If the keypoints lie on the right half of the image and below the right boundary line, then the keypoints belong to the right row. The least square fit is performed twice: one for the left row and the other for the right row. Two points from each line on the same row pixel (top and bottom) are subsequently used to calculate the midpoint of the navigation line as follows:

$$x_{top} = (x_{top,L} + x_{top,R})/2, \quad (9)$$

$$x_{bottom} = (x_{bottom,L} + x_{bottom,R})/2, \quad (10)$$

$$m_{nav} = (y_{bottom} - y_{top})/(x_{bottom} - x_{top}), \quad (11)$$

$$c_{nav} = y_{bottom} - m_{nav}x_{bottom}, \quad (12)$$

$$y = m_{nav}x + c_{nav}, \quad (13)$$

where m_{nav} and c_{nav} represent the slope and intercept, respectively, of the navigation line; x_{top} denotes the midpoint between $x_{top,L}$ and $x_{top,R}$, the column pixel on the top row of the image; and $x_{top,L}$ and $x_{top,R}$ represent the left pixel and right pixel, respectively, of the top row of the fitted line. Similarly, $x_{bottom,L}$ and $x_{bottom,R}$ denote the left pixel and right pixel, respectively, of the bottom row of the fitted line. Afterward, x_{top} , x_{bottom} , m_{nav} , and c_{nav} are calculated via Equations (9–12). y_{top} and y_{bottom} represent the top row pixel and bottom row pixel, respectively. The navigation line equation is shown in Equation (13).

2.6 FAST keypoint detection at different illumination intensities

The effect of illumination intensity on FAST keypoint detection is presented. Owing to different illumination intensities during the day, maize field images at different times are captured to compare the FAST keypoint detection results. The images captured from various illumination intensities (from top to bottom) are shown in Figure 8. At 10:30, the illumination intensity is high, the FAST algorithm detects a large number of keypoints. At 13:30, the illumination intensity further increases, leading to an even greater number of detected keypoints. At 16:30, the illumination intensity is much lower than that at 10:30, and the number of keypoints are much lesser. At 18:30, the illumination intensity is very low (sunset is at 19:00), and the FAST can detect fewer keypoints than at 16:30. The FAST threshold value is 40 for all the images. These results indicate that fewer keypoints are detected under low illumination, whereas high illumination produces a substantially larger number of keypoints. Notably, FAST keypoints can still be detected even at very low illumination levels. The proposed method is specifically designed for maize and exhibits robustness across various growth stages, field conditions, and soil color variations [7].

2.7 Data collection and test platform

The maize field test images were collected from various fields in Kanjanaburi Province, Thailand. The reference RGB images were collected under favorable field conditions at different maize growth stages, where the maize rows were continuous and illumination was sufficient.

A Kinect V2 camera was employed in this study. The algorithm was implemented in Python 3.12.6 via OpenCV version 4.10.0 and runs on a laptop with an 11th Gen Intel(R) Core(TM) i5-1135G7 processor at 2.40 GHz and 16.00 GB of RAM.

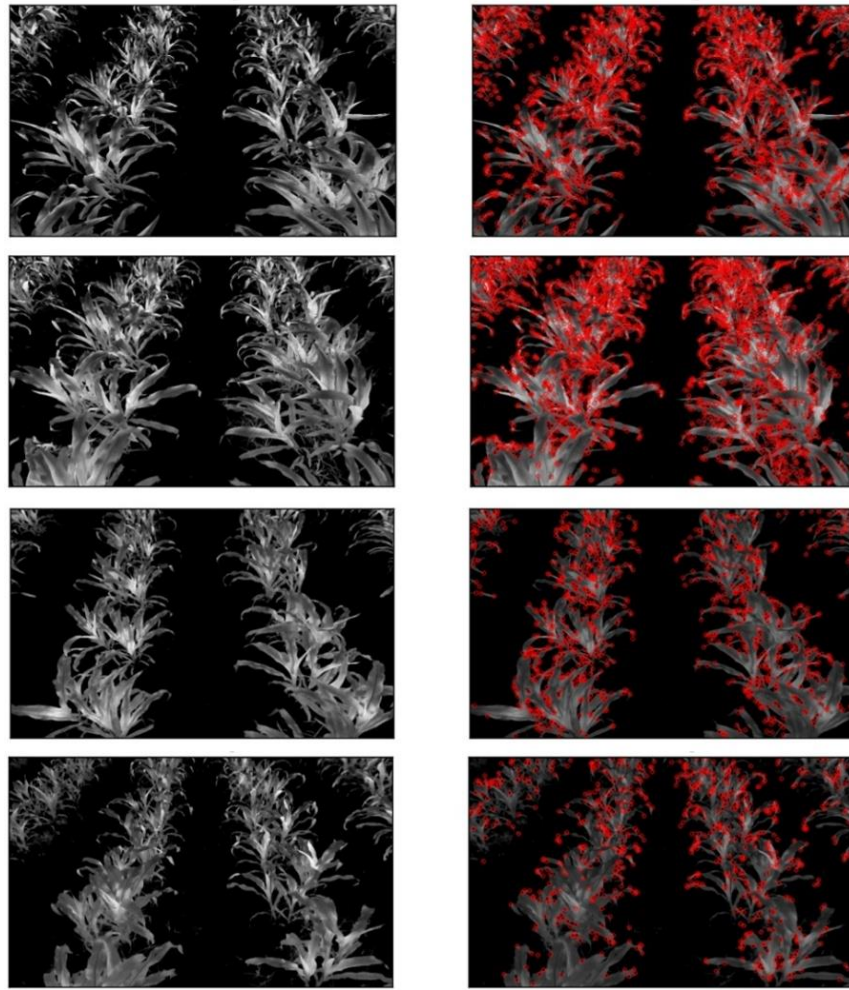


Figure 8 Images captured at different illumination intensities at 10:30, 13:30, 16:30, and 18:30 on a sunny day (from top to bottom); excess green image (left) and the FAST keypoints (right)

2.8 Benchmarks

The excess red and excess green methods, which utilize Otsu's method, serve as benchmarks. A soil detection approach based on the excess red index. The RGB image is converted to an excess red image.

$$ExR = 2.1r - g - b \quad (14)$$

Afterward, the excess red image is blurred to reduce noise. A binary image is obtained via Otsu's threshold. The interrow navigation line is extracted from the soil segment in the middle of the image.

Zhou's method is a crop row detection approach that is based on the excess green index [14], which is used as a benchmark and outperforms Jiang's method [7] and Zhang's method [13]. The excess green index is effective at identifying areas of green color in an image. However, Zhou's method requires a good initial region of interest.

3. Results

In the performance evaluation, the accuracy of the navigation line, the computation times for the proposed method and benchmarks, and the results of the proposed method for various field conditions are presented.

3.1 Results of different growth stages

The growth stages of maize are shown in Figure 9. The heights of the crops in Periods I, II, III, and IV were approximately 15, 25, 35, and 50 cm, respectively. In Period IV, the boundary line is adjusted for the appropriate ROI. Note that in each period, the camera height and tilt angle were adjusted to maintain a consistent perspective. The lighting conditions and soil colors differ across periods because the images were acquired from different fields. Note that the maximum crop height is 50 cm, due to limitations in the camera position.

Table 1 presents the RMSE of the navigation line calculated from image processing from sequence images. The benchmarks are ExR and Zhou's method [14] (ExG). The FAST threshold value in the proposed method ranges from 10 to 40. In Period I, the accuracy of the ExG+FAST approach is comparable to that reported by Zhou. In Periods II, ExG+FAST outperforms the methods of Zhou and ExR when using threshold values of 10. In Periods III-IV, ExG+FAST outperforms the methods of Zhou and ExR when using threshold values of 40.

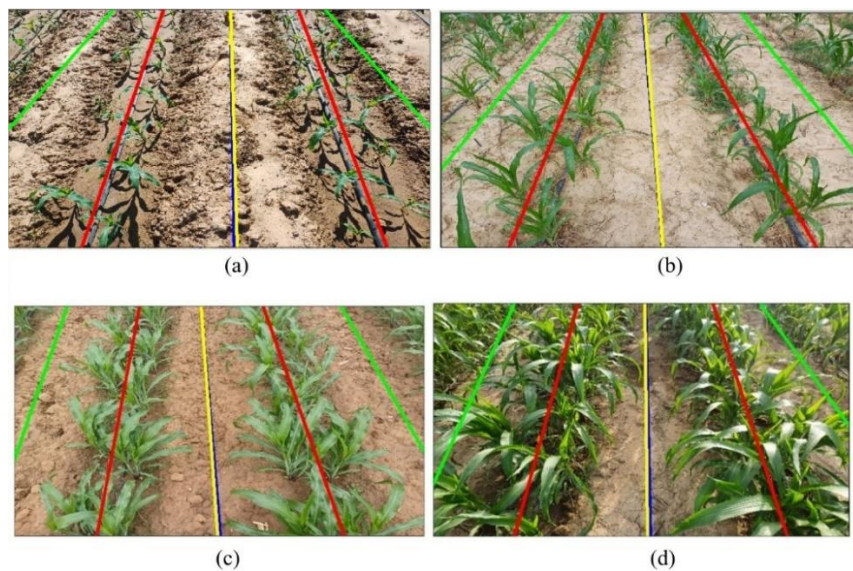


Figure 9 Maize fields with different growth stages from different plots, 100 images were collected for each period from Mueang District, Kanchanaburi Province: (a) Period I (collected on 18 September 2025 at 15:17), (b) Period II (collected on 26 September 2025 at 12:21), (c) Period III (collected on 15 June 2025 at 10:23), and (d) Period IV (collected on 6 June 2025 at 17:20)

Table 1 RMSE (pixels) of the extracted navigation line.

Method	Period I	Period II	Period III	Period IV	Average
ExG+FAST (threshold=10)	12.64	12.86	12.82	16.58	13.73
ExG+FAST (threshold=20)	13.20	13.63	12.82	19.26	14.73
ExG+FAST (threshold=30)	14.56	19.40	12.98	16.18	15.78
ExG+FAST (threshold=40)	13.21	24.98	10.06	15.69	15.98
Zhou [14] (ExG)	12.86	22.86	13.09	21.09	17.47
ExR	23.90	21.36	15.54	21.06	20.47

In Period I, the RMSE of the ExG+FAST method ranged from 12.64 to 14.56 pixels. The maize plants are sparse and exhibit uneven early-stage growth. The leaves are widely scattered, causing keypoints to be distributed farther apart, which negatively affects the line-fitting process. When the threshold is 10, the number of detected keypoints becomes sufficient, resulting in the smallest RMSE values.

In Period II, reducing the FAST threshold from 40 to 10 leads to a noticeable decrease in RMSE. The RMSE of the ExG+FAST method during this period ranges from 12.86 to 24.98 pixels. Compared with Period I, maize plants exhibit greater leaf density, resulting in a higher number of detected keypoints. Lower threshold values further increase keypoint density, and when keypoints are located closer to one another, the line-fitting process becomes more stable and accurate. The spatial distribution of keypoints thus reflects a trade-off between the structural regularity of the crop rows and the degree of leaf clutter.

In Periods III and IV, the lowest RMSE is achieved at a threshold value of 40. During these stages, maize leaf density is extremely high, yet a threshold of 40 still yields a sufficient number of keypoints for stable line fitting. At lower threshold values, however, the keypoints become excessively dense, which decreases navigation-line accuracy. This degradation occurs because large, overlapping leaves extend across the interrow space, creating irregular and indented contours. These leaf-induced indentations distort the keypoint distribution and adversely affect the line-fitting process, resulting in reduced accuracy when lower threshold values are used.

It should be noted that when the threshold value exceeds 40, the performance of the proposed method decreases. In contrast, when the threshold value is below 10, the performance remains comparable to that obtained with a threshold of 10, although the computational time increases.

The average RMSE across all periods for the ExG+FAST method is approximately 13.73 pixels for threshold values of 10. The highest RMSE, 15.98 pixels, occurs at a threshold value of 40. These results indicate that, overall, higher threshold values lead to reduced accuracy in navigation line extraction. The average RMSE of the ExG+FAST method is lower than that of the benchmark methods. For both benchmark methods, plant indentation and weed interference can either contract or expand the perceived plant boundary, consequently shifting the estimated navigation line away from its true position. Furthermore, the RMSE obtained with ExG+FAST is considered acceptable for practical applications, given that the processed images have a width of 640 pixels. This error will be used as an input for the robot heading controller.

Table 2 reports the R^2 values of the computed navigation lines. For the ExG+FAST method, the R^2 values range from -0.37 to 0.75 across all threshold settings and periods. Notably, the R^2 values of Zhou's method are negative for all periods. In Periods I, the R^2 values of ExG+FAST (0.42 – 0.75) are higher than that of ExR (0.19). In Period II–IV, R^2 values of ExG+FAST lies between 0 and 1 , with the exception of Period III at a threshold value of 30 . In Period IV, the R^2 of ExG+FAST (0.10) becomes comparable to that of ExR method (0.12). Overall, the average R^2 values of ExG+FAST range from 0.19 to 0.52 .

When both RMSE and R^2 are considered, distinct performance trends are observed across the growth periods. In Period I, EXG+FAST with a threshold value of 10 achieves an R^2 value of 0.75 with an RMSE value is 12.64 , outperforming the benchmark methods. In Period II, ExG+FAST with a threshold value of 10 again outperforms the benchmark in terms of both RMSE (12.86) and R^2 (0.54). In Periods III, ExG+FAST at a threshold value of 40 demonstrates performance in both metrics compared with the other methods. In Periods IV, ExG+FAST at a threshold value of 40 yields the lowest RMSE (15.69), while its R^2 value (0.10) remains comparable to that of the ExR method.

Note that in Period III, the negative R^2 value (-0.37) of ExG+FAST indicates that the model predictions performed worse than using the mean of the target data as a predictor. Although the maize plant was well aligned with the center line and exhibited a smooth shape of vegetation edge, the FAST keypoints were broadly distributed across the leaves. However, the fitted line produced low estimation accuracy, resulting in performance inferior to the mean-value baseline.

The R^2 value decreased markedly in Period IV because vegetation occupied a larger proportion of the image, causing the FAST keypoints to be more widely distributed along the horizontal axis. This distribution reduced the accuracy of the fitted line and consequently diminished the performance of the ExG+FAST method during Period IV. These findings suggest that the proposed method is more suitable for application during Periods I–III.

Table 2 R^2 of the extracted navigation line.

Method	Period I	Period II	Period III	Period IV	Average
ExG+FAST (threshold=10)	0.75	0.54	0.55	0.21	0.51
ExG+FAST (threshold=20)	0.73	0.64	0.55	0.18	0.52
ExG+FAST (threshold=30)	0.42	0.46	-0.37	0.26	0.19
ExG+FAST (threshold=40)	0.52	0.39	0.56	0.10	0.39
Zhou [14] (ExG)	-1.09	-0.33	-0.64	-0.08	-0.54
ExR	0.19	0.26	-0.28	0.12	0.07

Table 3 presents the image processing times used to extract the navigation line. The processing time includes image resizing from 1920x1080 pixels to 640x360 pixels. The computation time of the proposed method is much shorter than that of Zhou's method (308 ms) but similar to that of ExR method (51 ms). The processing times of all the growth stages ranged from 53 to 66 ms at a threshold value of 10. At a threshold value of 10, the FAST keypoint detection requires a longer time to process than at the higher threshold value because of the large number of keypoints. On the other hand, at a threshold value of 40, the processing time is shorter in the range of 47 to 48 ms. The average time tends to decrease from 61 to 48 ms when the threshold value is higher.

Table 3 Processing time (milliseconds) of navigation line extraction.

Method	Period I	Period II	Period III	Period IV	Average
ExG+FAST (threshold=10)	66	53	62	61	61
ExG+FAST (threshold=20)	54	49	54	53	53
ExG+FAST (threshold=30)	50	48	57	54	52
ExG+FAST (threshold=40)	47	47	48	48	48
Zhou [14] (ExG)	308	309	316	320	313
ExR	51	48	48	42	47

3.2 Results of different field conditions

Although field conditions differ in illumination, soil color, and vegetation density, these variations can influence the quality of the EXG+FAST output. To examine the consistency of ROI selection, the same boundary line was applied to images obtained under different conditions but within the same viewpoint. The results confirm that a single ROI configuration remains valid across condition changes. The results of the navigation line extraction of EXG+FAST with a threshold value of 10 under various field conditions, such as uneven plant growth, broken rows, very low light intensity, and different weed conditions, are shown in Figure 10. The EXG+FAST method is very accurate under these conditions. Note that very low light intensity is a representation of the lighting conditions on a cloudy day.

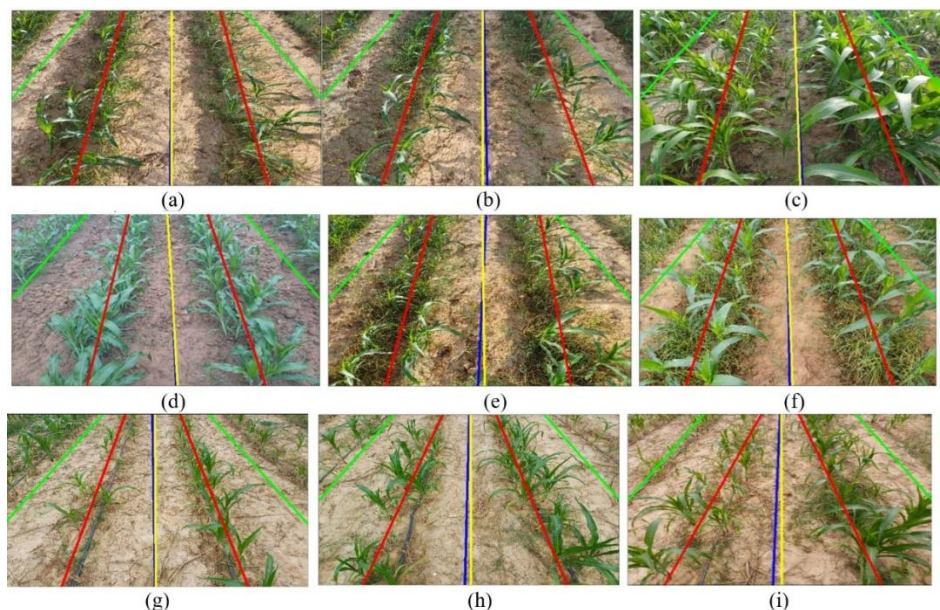


Figure 10 EXG+FAST results: (a) uneven plant growth, (b) broken rows, (c) uneven plant growth, (d) very low light intensity, (e) sparse weeds, (f) medium weeds, (g) sparse crop, (h) broken rows, and (i) uneven plant growth

4. Discussion

For LiDAR-based row-following navigation [16], point cloud data were used to train a neural network to detect crop row lines and generate waypoints for a mobile robot platform. The LiDAR sensor detects crop stems and provides the corresponding point cloud data; therefore, the method is applicable primarily to crops whose stems effectively reflect the laser. It is particularly suitable for later maize growth stages (Periods III and IV), when stem height and diameter are relatively large. However, the cost of LiDAR sensors remains high, especially for outdoor applications.

Navigation line extraction for a cabbage-harvesting robot [20] employed the HSV color model to detect cabbage, soil, and plastic mulch. Although the method was successfully validated under varying lighting conditions for mature, heading-stage cabbages, it has not been tested across different growth stages, soil color variations, or weed conditions, all of which may affect the accuracy of crop row detection. Furthermore, the row structure of cabbage differs substantially from that of maize, which exhibits more scattered foliage.

Maize zone classification in [21] required a large dataset to train the YOLO model for fertilization and navigation control. The front-view of the interrow space was captured under diverse field conditions, including variations in illumination, growth stages, and weed density. The method is relatively straightforward to implement but necessitates CUDA support for real-time processing. Although YOLO achieved superior performance compared to LiDAR-based approaches, the data acquisition process is time-consuming and labor. Additionally, ExG+FAST can be implemented on low-cost CPUs or embedded platforms, such as Raspberry Pi and Jetson Nano. Owing to the perspective characteristics of front-view images, the proposed method is less suitable for early crop growth stages. Furthermore, the method does not require additional data collection. However, its performance remains limited under conditions of dense weed infestation, which can interfere with accurate navigation line extraction.

5. Conclusions

This work proposed FAST keypoint detection on excess green images for navigation line extraction of the AGFD robot. The keypoints were distributed throughout both the crop and soil in the gray image but only on the crop row in the excess green image. On the basis of this characteristic, the keypoints of the left and right rows are fitted into lines via the least squares method. These two lines are used for determining the navigation line. Moreover, a strategy for automatic initialization and adaptive boundary line generation is proposed, in which the initial region of interest (ROI) is detected using YOLO-based soil segmentation. This approach eliminates the need for manual ROI specification and enhances system adaptability across diverse field conditions. Four maize growth stages were tested. The results revealed that FAST keypoint detection on excess green images outperforms the benchmarks for both the accuracy of navigation line extraction and shorter processing times. The accuracy of the proposed method under various field conditions is presented. The optimal threshold is growth-stage dependent, and the method is fast and robust.

The performance of the method is limited under conditions of severe weed infestation. Future work should focus on improving weed discrimination within the interrow space to enhance overall accuracy. Furthermore, an automatic thresholding approach should be developed to improve the applicability of the proposed method in real field operations.

6. Acknowledgements

This research was funded by the Faculty of Engineering, King Mongkut's University of Technology North Bangkok. Contract no. ENG-67-008.

7. References

- [1] Kuan YN, Goh KM, Lim LL. Systematic review on machine learning and computer vision in precision agriculture: applications, trends, and emerging techniques. *Eng Appl Artif Intell*. 2025;148:110401.
- [2] Zhu ZX, Chen J, Yoshida T, Torisu R, Song ZH, Mao ER. Path tracking control of autonomous agricultural mobile robots. *J Zhejiang Univ - Sci A*. 2007;8:1596-603.
- [3] Burgos-Artiz XP, Ribeiro A, Guijarro M, Pajares G. Real-time image processing for crop/weed discrimination in maize fields. *Comput Electron Agric*. 2011;75(2):337-46.
- [4] Ji R, Qi L. Crop-row detection algorithm based on Random Hough Transformation. *Math Comput Model*. 2011;54(3-4):1016-20.
- [5] Rovira-Más F, Zhang Q, Reid JF, Will JD. Hough-transform-based vision algorithm for crop row detection of an automated agricultural vehicle. *Proc Inst Mech Eng D: J Automob Eng*. 2005;219(8): 999-1010.
- [6] Bakker T, Wouters H, van Asselt K, Bontsema J, Tang L, Müller J, et al. A vision based row detection system for sugar beet. *Comput Electron Agric*. 2008;60(1):87-95.
- [7] Jiang G, Wang Z, Liu H. Automatic detection of crop rows based on multi-ROIs. *Expert Systems With Applications*. 2015;42(5): 2429-41.
- [8] Montalvo M, Pajares G, Guerrero JM, Romeo J, Guijarro M, Ribeiro A, et al. Automatic detection of crop rows in maize fields with high weeds pressure. *Expert Syst Appl*. 2012;39(15):11889-97.
- [9] Bengochea-Guevara JM, Conesa-Muñoz J, Andújar D, Ribeiro A. Merge fuzzy visual servoing and GPS-Based planning to obtain a proper navigation behavior for a small crop-inspection robot. *Sensors*. 2016;16(3): 276.
- [10] Liu L, Mei T, Niu R, Wang J, Liu Y, Chu S. RBF-Based monocular vision navigation for small vehicles in narrow space below maize canopy. *Appl Sci*. 2016;6(6):182.
- [11] Yang S, Mei S, Zhang Y. Detection of maize navigation centerline based on machine vision. *IFAC-Pap*. 2018;51(17):570-5.
- [12] Suriyakoon S, Ruangpayoongsak N. Leading point based interrow robot guidance in corn fields. *Proceeding of International Conference on Control and Robotics Engineering (ICCRE)*; 2017 Apr 1-3; Bangkok, Thailand. USA: IEEE; 2017.
- [13] Zhang X, Li X, Zhang B, Zhou J, Tian G, Xiong Y, et al. Automated robust crop-row detection in maize fields based on position clustering algorithm and shortest path method. *Comput Electron Agric*. 2018;154:165-75.
- [14] Zhou Y, Yang Y, Zhang B, Wen X, Yue X, Chen L. Autonomous detection of crop rows based on adaptive multi-ROI in maize fields. *nt J Agric Biol En*. 2021;14(4):217-25.

- [15] Li X, Su J, Yue Z, Duan F. Adaptive Multi-ROI agricultural robot navigation line extraction based on image semantic segmentation. *Sensors*. 2022;22(20):7707.
- [16] Affonso F, Tommaselli FAG, Capezzuto G, Gasparino MV, Chowdhary G, Becker M. CROW: a self-supervised crop row navigation algorithm for agricultural fields. *J Intell Robot Syst*. 2025;111:28.
- [17] Liang X, Chen B, Wei C, Zhang X. Inter-row navigation line detection for cotton with broken rows. *Plant Methods*. 2022;18:90.
- [18] Zheng K, Zhao X, Han C, He Y, Zhai C, Zhao C. Design and experiment of an automatic row-oriented spraying system based on machine vision for early-stage maize crops. *Agriculture*. 2023;13(3):691.
- [19] Wei J, Zhang M, Wu C, Ma Q, Wang W, Wan C. Accurate crop row recognition of maize at the seedling stage using lightweight network. *Int J Agric Biol Eng*. 2024;17(1):189-98.
- [20] Zhang J, Ma Y, Fan Y, Su N, He X, Zhang H. Real-Time navigation line extraction for cabbage harvesting robots in complex farmland environments: a multi-mask fusion and dynamic row-tracking approach. *Int Core J Eng*. 2025;11(6):291-304.
- [21] Ruangpayoongsak N, Rangdeang T, Puakhom C. Vision-based maize field zone classification for control of robot automatically dispensing granular fertilizer. *Agr Nat Resour*. 2025;59(6):590605.
- [22] Guijarro M, Pajares G, Riomoros I, Herrera PJ, Burgos-Artizzu XP, Ribeiro A. Automatic segmentation of relevant textures in agricultural images. *Comput Electron Agric*. 2011;75(1):75-83.
- [23] Rosten E, Drummond T. Machine learning for high-speed corner detection. In: Leonardis A, Bischof H, Pinz A, editors. *ECCV 2006. Lecture Notes in Computer Science*, vol. 3951. Berlin: Springer; 2006. p. 430-43.
- [24] Oliveira H, Vangasse A, Soares L, Oliveira A, Ferreira B, Leite G. An adaptable mobile robot platform with vision-based perception for precision agriculture. *Proceeding of 2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE); 2023 Oct 9-11; Salvador, Brazil. USA: IEEE; 2023. p. 466-71.*
- [25] Roboflow. Row object detection dataset [Internet]. 2025 [cited 2025 Dec 26]. Available from: <https://universe.roboflow.com/tps-klzw4/row-6hskc-qp5g7>.