



Visual navigation and crop mapping of a phenotyping robot MARS-PhenoBot in simulation

Zhengkun Li^a , Rui Xu^a, Changying Li^{a,*} , Longsheng Fu^b

^a Bio-Sensing, Automation, and Intelligence Laboratory, Department of Agricultural and Biological Engineering, University of Florida, Gainesville, Florida, USA

^b College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling, Shaanxi, China

ARTICLE INFO

Keywords:

Phenotyping robot
Visual navigation
Row following
Deep learning
Visual SLAM
Mapping
Simulation

ABSTRACT

Cultivating high-yield and high-quality crops is important for addressing the growing demand for food and fiber from an increasing population. In selective breeding programs, autonomous robotic systems have shown great potential to replace manual phenotypic trait measurements which are time-consuming and labor-intensive. In this paper, we presented a Robot Operating System (ROS)-based phenotyping robot, MARS (Modular Agricultural Robotic System)-PhenoBot, and demonstrated its visual navigation and field mapping capacities in the Gazebo simulation environment. MARS-PhenoBot was a solar-powered modular robotic platform with a four-wheel steering and four-wheel driving configuration. We developed a navigation strategy that fuses multiple cameras to guide the robot to follow crop rows and transition between them, enabling visual navigation across the entire field without relying on global navigation satellite system (GNSS) signals. Three row-detection algorithms, including thresholding-based, detection-based, and segmentation-based methods, were compared and evaluated in simulated crop fields with discontinuous and continuous crop rows, as well as with and without the presence of weeds. The results demonstrated that the segmentation-based method achieved the lowest average cross-track errors of 2.5 cm for discontinuous scenarios and 0.8 cm for continuous scenarios in row detection. Additionally, a field mapping workflow based on RTAB-MAP (Real-Time Appearance-Based Mapping) and V-SLAM (Visual Simultaneous Localization and Mapping) was developed. The workflow produced the 2D maps identifying crop and weed locations, as well as 3D models represented as point clouds for crop shapes and structures. Using this mapping workflow, the average crop localization error was measured at 6.4 cm, primarily caused by the visual odometry drift. The generated point clouds of crops could support further phenotyping analyses, such as crop height/diameter measurements and leaf counting. The methodology developed in this study could be transferred to real-world robots that are capable of automated robotic phenotyping for in-field crops, providing an effective tool for accelerating selective breeding programs.

1. Introduction

The growing world population, climate change, degradation and loss of available land, and the increasing occurrence of new pests and diseases all threaten the world's food supply [1–3]. It has been reported that rapidly rising demand in global food production requires a doubling of crop production yields by 2050 [4]. Recently, high-throughput and high-resolution phenotyping technology has been emphasized to breed genotypes of high yield and quality. This technology explores how plants respond to environmental and genetic perturbations by identifying and assessing both simple and complex plant traits including, but not limited to, plant height, biomass, flowering time, and grain yield [5]. Currently,

those phenotypic traits are measured manually by field technicians, which is labor-intensive and prone to human measurement errors [6].

Unmanned ground vehicles (UGV) and unmanned aerial vehicles (UAV) are major autonomous platforms that are used for measuring phenotypic traits in the field [7]. Robotic phenotyping systems focus on either collecting images of plants using various cameras (such as RGB cameras, hyperspectral cameras, and thermal cameras) [8–10] or reconstructing the 3D structure of the plants with different range sensors (including stereo cameras, Time-of-Flight of light sensors, LiDAR sensors, and Computed Tomography sensors) [11–14]. Current phenotyping systems allow for data collection at different scales, including at the organ, plant, plot, and field level [5]. UAV is typically applied to

* Corresponding author.

E-mail address: cli2@ufl.edu (C. Li).

biomass estimation and identification of crop diseases at the plot and field level [15]. UGV can measure detailed phenotypic traits at the organ and plant level including, but not limited to, plant height, plant diameter, and plant architecture are [16–18].

Compared to UAVs, UGVs have the advantages of a larger sensor payload and higher sensor resolution to the crops. Some researchers have developed ground phenotyping robots using UGVs, including “BoniRob” [19], “TerraSentia” [20,21], “RoboHortic” [22], Robotanist [13], Vinobot [23], and “PATHoBot” [24], along with more scalable modular platforms such as Thorvald [25], MARS [26], and Amiga [27]. Compared with traditional agricultural tractors, these lightweight phenotyping robots can be autonomous and intelligent, making them less dependent on human labor. Also, they create little or no soil compaction and are less limited by field conditions than heavy tractors. Advanced computer vision algorithms are deployed on these platforms to perform field phenotyping tasks, such as plant counting [28], stem trait estimation [29,30], and leaf trait estimation [31].

Ground phenotyping robots operate in uncontrolled or semi-structured environments, presenting significant challenges for navigation due to varying terrain, plant densities, and environmental conditions [17]. While GNSS (Global Navigation Satellite System) and IMU (Inertial Measurement Unit) provide robust solutions for navigation, they often depend on high-performance RTK-GNSS (Real-Time Kinematic GNSS), which can be cost-prohibitive and limit broader applications [32,33]. Moreover, these systems can be less effective in areas with weak GNSS signals, such as under dense foliage or near tall structures that cause signal obstructions [34,35]. Therefore, visual navigation and LiDAR-based navigation can be used as supplements to GNSS-based navigation in large-scale field applications.

LiDAR-based navigation relies on landmarks that can define crop rows, such as the plant, trunk, and poles in a polytunnel [18,36,37]. To detect the crop row in the point cloud, Random Sample Consensus (RANSAC), Hough Transform (HT), or Density-Based Spatial Clustering of Applications with Noise (DBSCAN) was generally used to overcome the noise influence [38]. Besides, another real-time iterative approach named PEARL had excellent performance in crop line extraction [39]. Other approaches such as Particle Filter (PF) with a laser beam model or Kalman filter (KF) with a line-detection algorithm were applied to robot navigation in the orchard and field [40]. However, relying solely on LiDAR sensors remains challenging for long-term navigation in complex agricultural fields due to dynamic environmental conditions, sensor occlusions, and variations in crop structures [41,42]. Factors such as changing light intensity, plant growth stages, and the presence of weeds or debris can affect the accuracy of point cloud data. Additionally, LiDAR-based methods struggle in environments with sparse landmarks

or repetitive patterns, where distinguishing between rows becomes difficult [43]. Several recent approaches combining vision sensors and LiDAR sensors provided potential improvements to address the challenges [44–46].

Compared to LiDAR, visual navigation using low-cost cameras is becoming increasingly popular, especially for row-planted crops [47]. In general, visual navigation can be divided into three stages: 1) applying machine vision methods to detect and segment vegetables from images; 2) calculating the navigation line from the detected crop row; and 3) controlling the robot's orientation relative to the crop row [48]. Most segmentation algorithms rely on image features such as colors and textures of the crop rows, which are sensitive to image quality and lighting conditions. These algorithms primarily use color-threshold segmentation using different color factors, such as ExR and ExG [49, 50]. As for navigation line detection methods, several classical algorithms including Hough Transform (HT), Linear Regression (LR), and Horizontal Fringes (HF) are widely applied to detect crop rows after plant segmentation [51].

Recent advancements in deep learning have introduced more robust and adaptive methods for crop row segmentation and navigation line detection. Convolutional Neural Networks (CNNs), in particular, have shown significant improvements in segmenting crop rows under varying environmental conditions, such as changing illumination, occlusions, and complex backgrounds [47,52]. These models leverage large datasets and can generalize well across different crop types and field conditions compared to traditional feature-based methods [53–57]. Based on the crop-row structure as the control signal, visual guidance systems were developed to guide robots to follow the crop rows [58–61]. However, due to the camera's limited field of view, the robot often loses sight of the crop rows as it approaches the end of a row. This creates challenges in accurately detecting the row transition and guiding the robot to the next row. The inability to maintain consistent visual cues during this transition can lead to misalignment, reduced efficiency, and potential damage to crops. To address this issue, visual navigation is typically used together with GNSS guidance, for example, fusing the visual navigation and GNSS guidance results or using the visual navigation for row following and switching to GNSS guidance when the robot transitions between rows [62]. Some other researchers also proposed a pure vision-based navigation framework without RTK-GNSS, which allows the robot to follow the crop rows accurately and handle the transition to the next row seamlessly within the same framework [59,63].

Agricultural field mapping is another important goal for phenotyping robots and has become increasingly important to crop monitoring and management. Most of the current field mapping research relies on remote sensing technologies to capture the whole field with a large view,

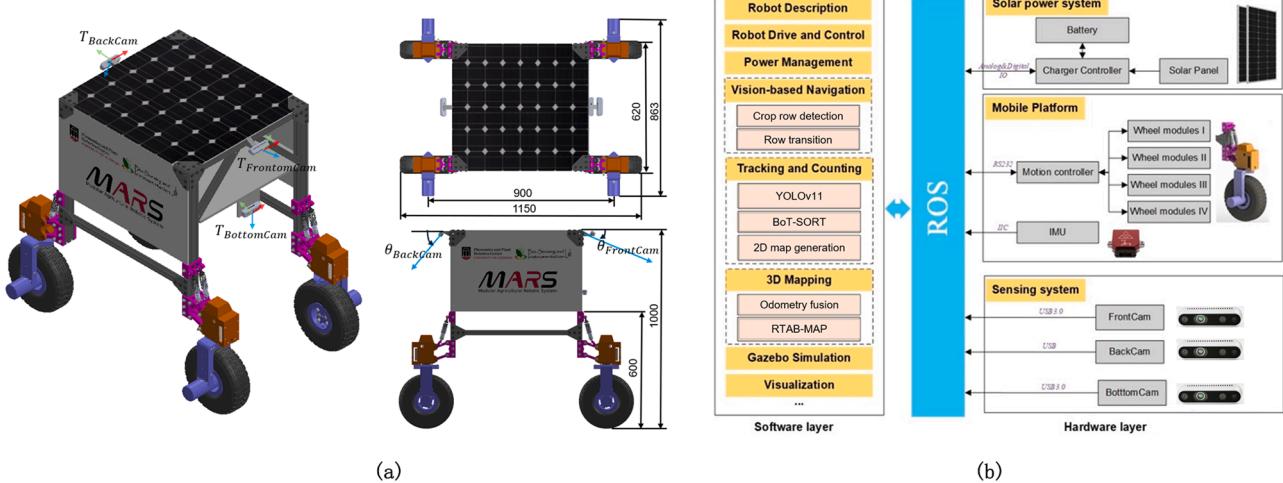


Fig. 1. Overview of phenotyping robot “MARS-PhenoBot”: (a) the mechanical design, (b) ROS based software architecture.

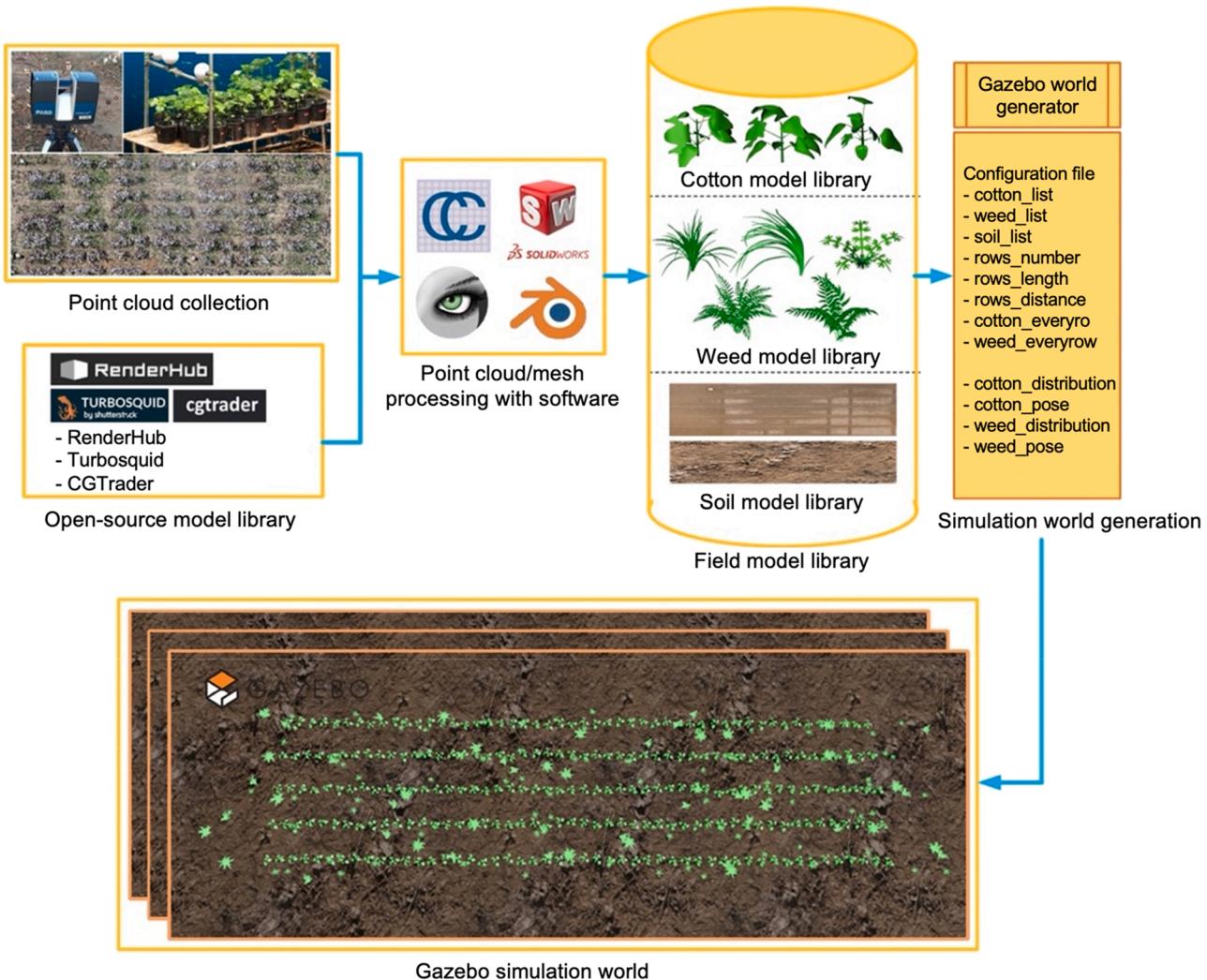


Fig. 2. Gazebo simulation world generation pipeline.

focusing on the measurement or estimation of the integral properties, such as soil health and nutrition, biomass, disease, and stress resistance [64]. For the ground mobile robot, the limited views can only cover its surrounding area. Existing approaches are limited without considering the spatial connections between the current local scene to others. The current approaches are far from meeting the needs of phenotyping tasks targeting long time-span and large-scale areas [65]. Thus, large-scale mapping technologies are necessary for deploying phenotyping tasks in the field. With simultaneous localization and mapping (SLAM) advancement, Visual SLAM technology has the potential to complete a large-scale phenotyping task. It can construct accurately a large-scale scene, track the position of the vision system on the global map, and estimate the motion trajectory. Several V-SLAM frameworks such as ORB-SLAM [66], RTAB-MAP [67] have been used to generate the 3D mapping for agricultural tasks [68–74].

Despite significant advancements in agricultural robotics, existing phenotyping systems often lack integrated workflows that simultaneously address both visual navigation and field mapping in dynamic and realistic environments. Many approaches either focus solely on individual tasks, such as navigation or mapping, or rely on overly simplified setups that do not adequately replicate field conditions. Moreover, the validation of algorithms in simulation environments tailored for agricultural scenarios is underexplored, especially in diverse crop and weed configurations. To address these gaps, we developed the MARS-PhenoBot, a ROS-based phenotyping robot, along with a

comprehensive simulation workflow to realize visual navigation and field mapping simultaneously in the Gazebo simulator.

The specific objectives were to 1) develop a phenotyping robot and simulation workflow for algorithm validation, 2) design and validate a visual navigation algorithm for row following and transition with three row-detection algorithms tested in four different crop scenarios, and 3) develop a field mapping workflow for 2D maps identifying crop and weed locations, as well as 3D models for crop shapes and structures.

2. Materials and methods

2.1. MARS-PhenoBot

MARS-PhenoBot, a lightweight version of the modular agricultural robotic system (MARS) [26], was developed for in-field monitoring and phenotyping (Fig. 1a). It was equipped with four-wheel drive four-wheel steering (4WS4W), allowing easy maneuver in unstructured environments. The frame, constructed from aluminum extrusions, can be adjusted to fit various crop layouts. Independent suspensions are utilized to improve stability on uneven terrain. It was powered by a Li-Po battery and a solar panel, managed by a charge controller and battery management system. Three Intel RealSense D435i cameras (Intel RealSense, California, United States) capture the front, back, and bottom views. The view of the *FrontCam* and *BackCam* were adjusted by the tilt angles $\theta_{FrontCam}$ and $\theta_{BackCam}$. The *BottomCam* is oriented in a nadir position to

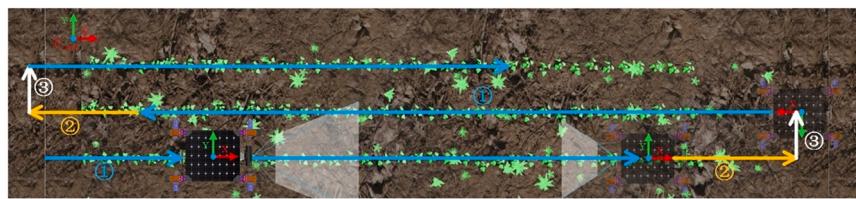


Fig. 3. Visual navigation scheme for a typical row crop field. The robot enters the field and begins row-following using the front camera (①, cyan lines). As it is approaching the end of the row, it continues row-following with the back camera (②, yellow lines). Finally, the robot transitions to the next crop row (③, white lines). This sequence is repeated to navigate through the entire field.

capture detailed observations of the plants. A high-performance on-board computer, NVIDIA Jetson TX2, is used to control the camera and robot.

The control software of MARS-PhenoBot was developed based on the Robot Operating System (ROS), as illustrated in Fig. 1(b). It can control the robot to perform various operations with control algorithms based on the observations and its surrounding environment. The hardware has three subsystems: a solar power system, a mobile platform, and a sensing system. The charge controller of the solar power system is responsible for monitoring the battery state and charging status while receiving control signals from the ROS master to facilitate power management. The mobile platform has a configuration of four-steering and four-driving and is controlled by a customized motion controller [26]. Wheel odometry estimated from the encoder of the motors were fused with an inertial measurement unit (IMU) using a Kalman filter to estimate the robot's location and pose. The *FrontCam* and *BackCam* were used in the visual navigation and *BottomCam* was used for plant mapping.

2.2. Simulation environment

In this study, we used the Gazebo simulator to model the phenotyping robot's operations across various crop fields, enabling comprehensive testing of the developed algorithms. Generating an effective simulation environment involved addressing two primary challenges: creating rich, realistic models and ensuring flexible configurability. High-fidelity models are necessary to maintain accuracy in translating simulation outcomes to real-world scenarios. Meanwhile, an easily configurable generator of simulation world allows us to rapidly adapt and modify the environment to test different conditions, improving the efficiency and versatility of our simulations.

SDF (Simulation description format) models are used in Gazebo simulator to describe objects and environments for robot simulators, visualization, and control. Within the description files, all the elements (or named models) including robots, lights, sensors, and static objects, are specified using various tags. For example, crop plant models can be described with “visual” tags to define the visual attributes, including the pose, geometry, materials, shadows, etc. Our study aimed to develop a Gazebo world generation workflow to generate the world description files by customizing the simulation objects ([Fig. 2](#)).

Generally, a crop field is consisting of plants and soil. The plants (including crops and weeds) are defined by their type, quantity, and distribution rules. To enhance the diversity of the simulation world, we developed a 3D model library. There are two primary methods for acquiring 3D models: (1) creating mesh models from scanned high-resolution point clouds and (2) utilizing models from online 3D model repositories such as RenderHub, TurboSquid, and CGTrader. In our previous works [75], a 3D LiDAR scanner (FARO Technology, USA) was used to collect high-resolution point clouds from crop field and greenhouse. Then, the mesh models of single plants and soil could be created using CloudCompare or MeshLab software, and the mesh models could be further modified with 3D modeling software such as SolidWorks and Blender.

After generating sufficient models, we developed a Python script to

Algorithm 1

Field navigation scheme

automate the creation of Gazebo world files based on user-defined configuration files. The configuration files describe the parameters of the field, including model lists and the distribution rules for plants. To better simulate real-world field conditions, random noise was introduced to the positioning of plants. Furthermore, the pose of each plant, including both crop plants and weeds, was randomized to enhance the diversity and realism of the simulated scenes.

2.3. Visual navigation algorithm

We developed a visual navigation algorithm specifically designed for row crops, utilizing multiple cameras to enable autonomous navigation without relying on an explicit field map. The algorithm extracts visual features from camera inputs and incorporates prior knowledge of the crop row structure to guide the robot effectively through the field. The navigation process begins with a row-following module that identifies and tracks crop rows using advanced image processing techniques. While navigating along a row, the robot continuously monitors and adjusts its position and heading to stay aligned. Upon reaching the end of a row, the algorithm uses field layout data and the robot's heading information to transition to the next row, ensuring efficient coverage of the entire field.

2.3.1. Visual navigation workflow

Crops are typically planted in parallel, straight rows, which provides a structured layout that facilitates autonomous navigation for a mobile robot using only two cameras (Fig. 3). The field is divided into multiple, repetitive segments, each consisting of three navigation stages: (1) row

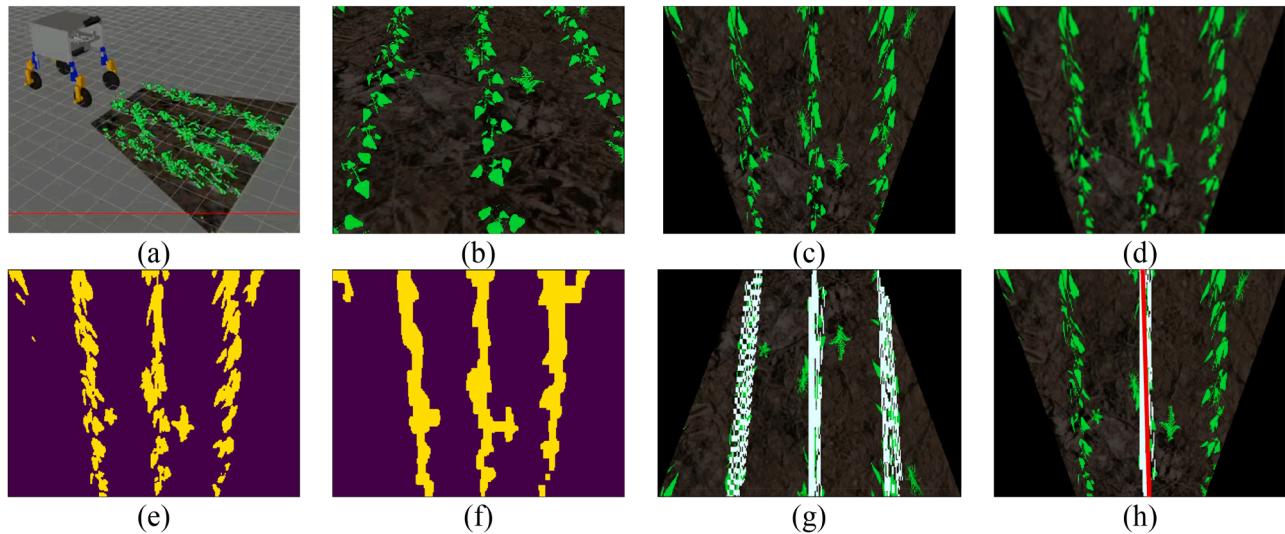


Fig. 4. The sequence of thresholding-based method for crop row detection process. (a) Gazebo Simulation, (b) Original Image, (c) Homography Transform, (d) Blur, (e)HSV Threshold, (f) Morphological Processing, (g) Probabilistic Hough Transform, (h) Crop Row Detection.

following with the front camera (cyan lines), (2) row following with the rear camera (yellow lines), and (3) row transition based on row distance (white lines). This sequence of behaviors is repeated for each row segment, allowing the robot to systematically navigate through all rows in the field.

Our field navigation approach ([Algorithm 1](#)) utilizes image streams from both front and rear cameras (FrontCam and BackCam) along with the yaw angle from an IMU sensor (Line 1). We developed a customized row-following controller based on a PID controller and Kalman filter. This controller uses feedback from the crop row detection algorithm to directly regulate the robot's angular velocity. Detailed descriptions of the row-following controller and the crop row detection algorithm can be found in [Sections 2.3.2 and 2.3.3](#), respectively. The navigation scheme also incorporates field layout information, such as the number of rows, to determine when the task is complete.

In the navigation control loop (Line 2), the customized crop row detection algorithm is initially applied to images from the front camera (FrontCam) (Line 4). If a crop row is detected, the robot enters Stage ① and follows the row using the line-following controller (Line 16). When the robot gets close to the end of the row (Stage ②), the FrontCam, angled forward, can no longer detect the crops. At this point, the rear camera (BackCam) replaces the FrontCam to detect the remaining crops and provide navigation references (Lines 6 and 14), keeping the robot within the crop row until it is no longer detected by the BackCam. The robot then transitions to Stage ③, where it moves to the next crop row. During this transition stage, the robot performs a 180° rotation (Line 8), using real-time yaw data from the IMU to ensure precise turning.

Subsequently, the robot moves perpendicular to the crop row (Line 10) until the *FrontCam* detects the next row. Visual-inertial odometry estimates the transition distance, ensuring that the movement matches the row spacing.

Throughout the entire field, the robot repeatedly executes the control loop (Lines 3–16) for each crop row until all rows have been traversed.

2.3.2. Crop row detection

To navigate along a crop row, we extracted the line along which the crops are arranged and then calculated the offset error for the line-following controller. Three types of crop row detection algorithms were developed for comparison.

Thresholding-based method: [Fig. 4](#) illustrates an example of an image processing sequence working on the captured image. As shown in [Fig. 4a](#), the view of the *FrontCam* covers multiple crop rows and several weeds in the Gazebo simulation. As the *FrontCam* was mounted on the central line of the robot, the central crop row in the image is the one to follow. We performed a homography transformation for the original image ([Fig. 4b](#)) to obtain an orthonormal perspective to ensure the rows were parallel in the image ([Fig. 4c](#)). Then, the blurring process was performed to remove salt and pepper noise to improve the signal-to-noise ratio ([Fig. 4d](#)) to suppress noise and preserve key information like contours and edges. The HSV (hue, saturation, value) threshold segmentation algorithm was used to segment the plants (cotton plants and weeds) from the soil background in the HSV color space ([Fig. 4e](#)). The resulting image was subjected to open and closed operations in

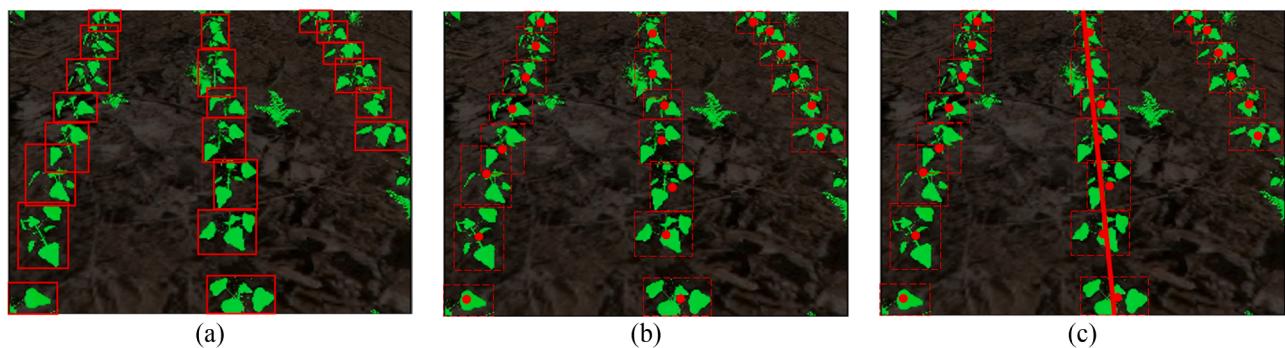


Fig. 5. The sequence of detector-based row detection. (a) individual crop detection, (b) central point extraction. (c) crop line extraction based on Hough transform.

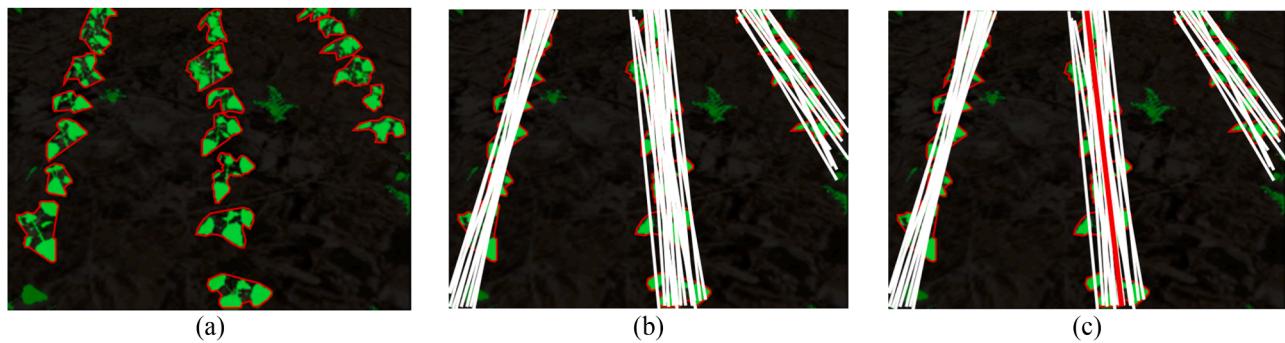


Fig. 6. The sequence of crop row detection process. (a) Individual crop segmentation, (b) Hough transform to find potential rows, (c) select the central line as the following row.

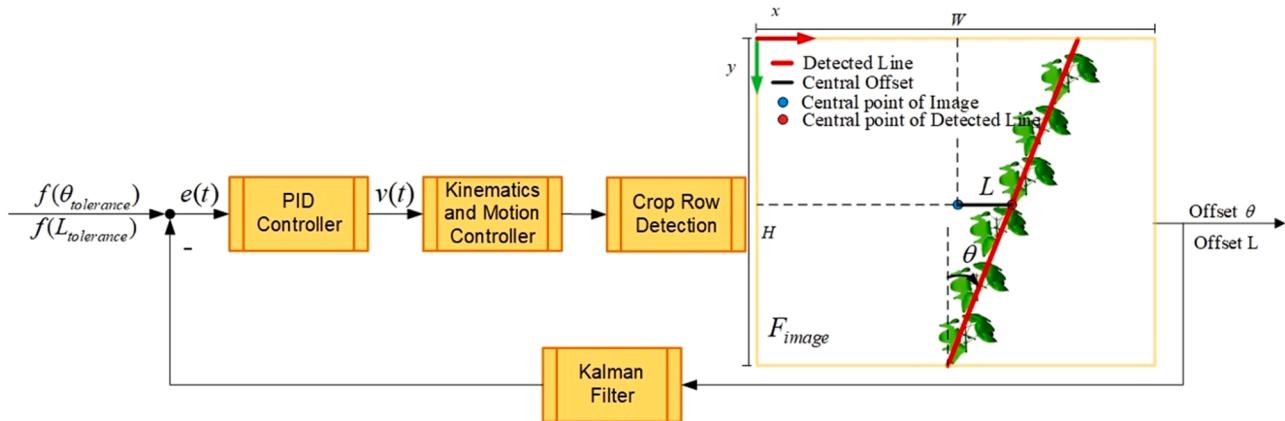


Fig. 7. The proposed row following controller. It uses a PID controller to control the robot's velocity to eliminate the offset which is calculated by an image processing approach. f is the tolerance function; $e(t)$ is the error; $v(t)$ is the controlled velocity.

morphological filtering to extend the pixels of the same crop rows, which integrated the separate plants of the same row with an integral object (Fig. 4f).

Next, probabilistic Hough transform [76] was used to find all straight lines in the processed image. Probabilistic Hough transform can filter the nonconforming detected line with the pre-defined parameters: the detected resolution in pixels and angle, the minimum number of intersecting points to detect a line, the minimum length of the detected line, and the maximum distance allowed to connect points in the same line.

Since multiple lines can be detected for a single crop row (Fig. 4g), the final line to follow must be extracted from the detected lines. A 2D line can be defined by an intersection point and a vertical angle. To determine the following line, the central point of the line set corresponding to the middle row (lines within the 0.35–0.65 width range of the image) was calculated as the intersection point. The vertical angle was determined by averaging the vertical angles of all the detected lines in the set (Fig. 4h).

Detection-based method: The traditional method, which relies on color differences to distinguish crops from soil, struggles to differentiate between weeds and crops. To address this limitation, deep learning methods provide a more robust solution for crop row detection. Specifically, a state-of-the-art YOLOv11 model was employed as the crop detector, trained using both simulation and real-world data. The trained detector identifies crops with high accuracy, and the central points of the detected crops are calculated. These central points are then processed using the Hough Transform to identify the crop row structure. Finally, the central line of the detected crop row is selected as the following line for navigation (Fig. 5).

Segmentation-based method: An alternative deep learning approach for crop row extraction utilizes a segmentation model (Fig. 6).

The core concept is to replace traditional thresholding-based segmentation methods with a lightweight deep learning segmentation model. YOLOv11 was selected as the crop segmentation model due to its efficiency and accuracy. Compared to detection-based methods, segmentation offers pixel-level predictions, enabling more precise crop row extraction. This approach not only enhances accuracy but also supports the segmentation of both individual plants and continuous rows, making it more applicable to real-world scenarios where such variations are common.

2.3.3. Row following controller

The crop row detection process uses outputs from the crop row detection algorithm, specifically, the angular offset (θ) and the horizontal offset (L) of the detected crop row—as feedback to adjust the robot's movement (Fig. 7). The angular offset (θ) represents the angle between the detected crop line and the vertical axis, while the horizontal offset (L) is determined by the difference between the image's central point and the central point of the detected crop line. These offset values are fed into a PID controller (proportional, integral, derivative), which adjusts the robot's linear and angular velocities to maintain accurate row-following.

In real-world applications, the detected offset is not always accurate, as it depends on image quality and the robustness of the crop row detection algorithm. An inaccurate offset can cause the robot to oscillate along the crop row. To mitigate this, two techniques are incorporated into the feedback control: a Kalman filter for smoothing the offset signal and offset tolerance regulation to constrain velocity adjustments. The Kalman filter estimates the error and generates measurements closer to the actual values, effectively reducing noise in the offset signal. Meanwhile, the offset tolerance method ensures that the robot adjusts its

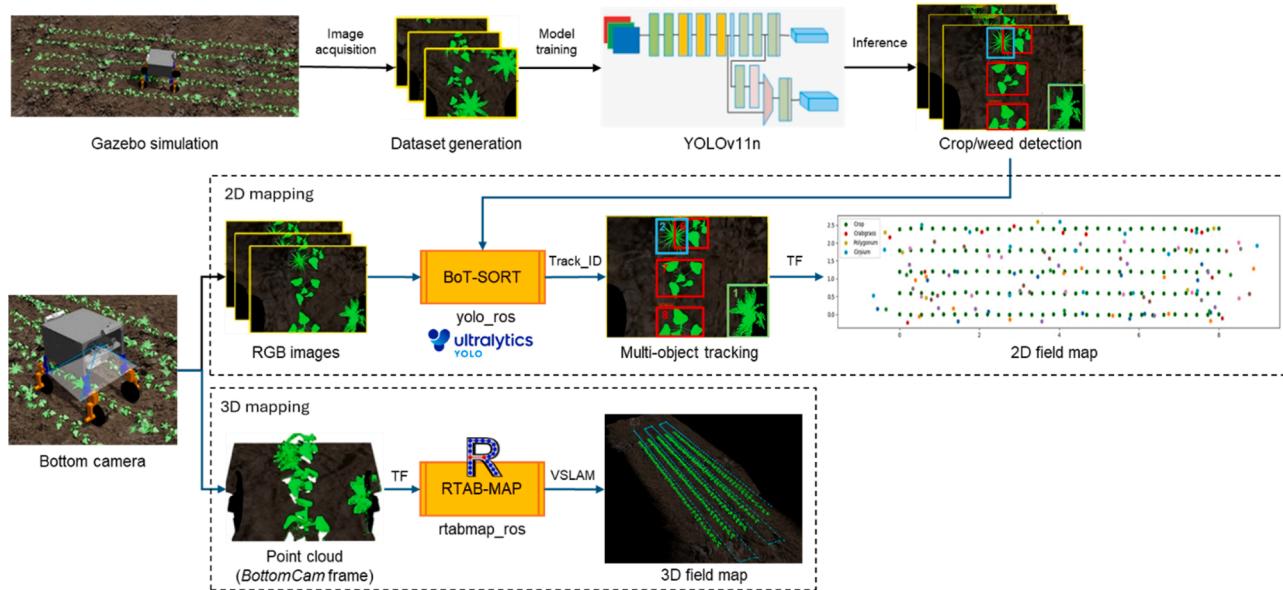


Fig. 8. The diagram of field mapping workflow.

velocity only when the offset exceeds a defined threshold. This approach keeps the offset within a reasonable range, improving stability and reducing unnecessary corrections.

Once the control error is calculated, a PID controller is employed to determine the control velocity. In this work, the robot is set to a constant linear velocity, while only the turning velocity is actively controlled using a proportional (P) controller. This design simplifies the control system while maintaining effective navigation along the crop row.

2.4. Field mapping workflow

Our proposed field mapping pipeline combined crop detection-based tracking and SfM-based (Structure-from-Motion) mapping approach, which can generate the 2D crop/weed distribution map and the 3D reconstruction of the field map (Fig. 8).

In the 2D mapping workflow, *BottomCam* was driven with the *realsense-ros* driver and registered the RGB image and point cloud. *BottomCam* possessed a smaller view and can observe more details information with few disturbances in the background. The registered RGB image was input to the *yolo_ros* node to generate the 2D bounding box of detected plants. A customized lightweight YOLOv11-nano detector was trained with the image dataset collected from the Gazebo simulation world. A tracking node based on BoT-SORT [77] architecture described

the detected bounding box data and published the tracking box result. The map generation node combined the registered point clouds data, transform tree, and the tracking box results to generate the 2D distribution map.

In the 3D mapping workflow, an open-source visual SLAM (Simultaneous Localization and Mapping) library, RTAB-Map (Real-Time Appearance-Based mapping) [67], was deployed to generate an accurate 3D map for the entire field. The *BottomCam* was used for both 2D and 3D mapping because it captured detailed crop and weed structures beneath the robot, enhancing the spatial resolution and accuracy of the reconstructed field map. The RTAB-Map system processed point cloud data from the *BottomCam* in conjunction with odometry information to build the 3D map. An extended Kalman filter was employed to fuse multiple data sources, including wheel odometry, IMU, and visual odometry derived from the *BottomCam*. This fusion improved localization accuracy and ensured consistency in mapping. The generated 3D map not only provided a detailed representation of the field but also enabled the extraction of individual crop models for further evaluation, such as analyzing crop morphology and spatial distribution.

2.5. Experiment setup

The Gazebo worlds were generated using our custom Gazebo world

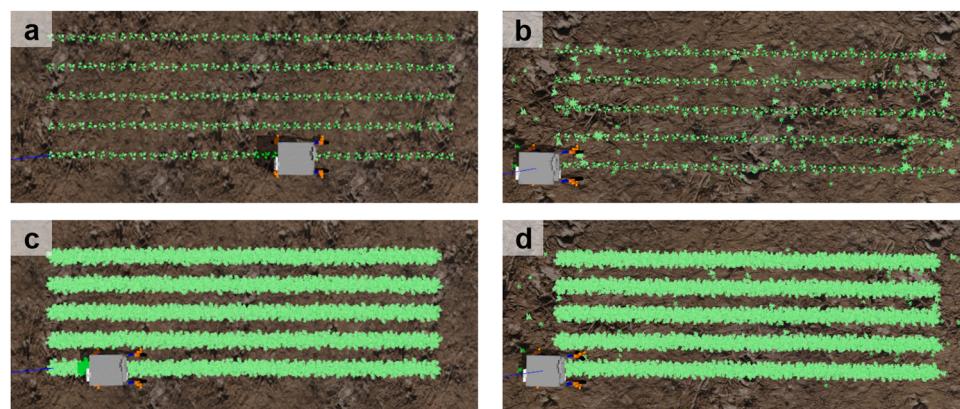


Fig. 9. Cases of simulation world using Gazebo generation pipeline. (a) and (b) are the simulated fields where the plants are separated to simulate the cotton seedling fields; (c) and (d) are the fields with continuous canopy to mimic the peanut field; (b) and (d) contains three types of weeds as the noise.

generator (Section 2.2). Four distinct simulation fields were created using a variety of 3D models, including representations of cotton and peanut plants, weeds, and soil (Fig. 9). Each field was configured with five crop rows, each 8 meters long and spaced 0.6 meters apart. Within a single crop row, 40 cotton or peanut plants were distributed approximately evenly, maintaining an average spacing of 20 cm. To simulate natural variability, Gaussian noise with a standard deviation of 0.05 meters was applied to the plant spacing, creating a realistic distribution that reflects actual field conditions. Crop and weed models were randomly selected from a predefined list, and their orientations were randomized to enhance the diversity of the simulation environment. Additionally, three common types of weeds—crabgrass, polygonum, and Cirsium—were incorporated. These weeds were randomly distributed within 0.3×0.3 m grids across the field regions.

The MARS-PhenoBot was described using Unified Robot Description Format (URDF) and integrated into Gazebo simulation environments. The tilt angles of the front and rear cameras ($\theta_{FrontCam}$ and $\theta_{BackCam}$) were set to 60° and 45° , respectively. During simulations, the robot's linear velocity was kept constant at 0.2 m/s. For the crop-row following controller, the offset tolerance parameters were set to $\theta_{tolerance} = 8^\circ$ and $L_{tolerance} = 8$ pixels. Additionally, the PID controller was configured in a proportional-only (P-control) mode, with a proportional coefficient of 0.1. With these parameters, the robot was able to execute both the navigation and field mapping pipelines simultaneously within the Gazebo simulator. All data generated by the ROS system was recorded to rosbag files for subsequent analysis.

For the four fields, three row extraction algorithms were applied to follow crop rows and guide the robot's navigation throughout the field. The visual navigation workflow was evaluated based on the resulting navigation path trajectories. As for the field mapping algorithm, single plants were manually extracted from the point cloud of the whole field for quality evaluation comparing with their 3D model.

2.6. Evaluation metrics

Metrics of YOLO Detection/segmentation: Considering the tradeoff between the detection accuracy and inference speed in a limited computational unit, we choose the lightweight YOLOv11n and YOLOv11n-seg model [78] as the detection and segmentation model to detect and segment the crops and weeds. Transfer learning was used to increase learning efficiency and generalization as the training dataset was small. In this study, the YOLO model was initialized by weights pre-trained on MS COCO dataset (<https://cocodataset.org/>) and then fine-tuned on our training set.

To train the plant detector, a total of 527 images from the three cameras were randomly sampled from four different scenarios. Crops and weeds in the images from scenarios (a) and (b) were annotated with bounding boxes and masks to create the detection and segmentation dataset. For scenarios (c) and (d), only the mask of the entire row was annotated and included in the segmentation dataset. These images were annotated using the online annotation tool Roboflow and augmented threefold through rotations, flips, shears, and exposure adjustments to increase diversity. The dataset was then divided into training set (70%) and validation set (30%).

The performance of YOLO models were evaluated with COCO metrics [79], specifically, including Precision, Recall, mAP50, and mAP95:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (1)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (2)$$

$$\text{IoU} = \frac{\text{area of overlap}}{\text{area of union}} \quad (3)$$

$$AP = \sum_{k=0}^{N-1} P(k)[r(k) - r(k+1)] \quad (4)$$

where TP, FP, and FN are the numbers of true positives, false positives, and false negatives, respectively. $P(k)$ is the precision at IoU threshold k , $r(k)$ is the recall at IoU threshold k , and N is the number of thresholds. In the COCO metrics, the standard IoU threshold used to define whether a detection is a true positive is 0.5. For mAP50 and mAP95, the calculation is an average of the Average Precision (AP) for different classes at a specific IoU threshold (50% for mAP50 and 95% for mAP95).

Metrics of visual navigation: Performance of visual navigation: In the simulation tests, the robot demonstrated its capability to navigate through the entire field using only visual crop features, without depending on external inputs such as GPS or landmarks. The cotton field was modeled as a 2D plane, where the crop arrangement was represented by x- and y-coordinates of the cotton plants. The locations and the headings of the robot were dynamically updated from the Gazebo simulator, providing the trajectories for evaluating the robot's navigation performance. To assess trajectory accuracy, cross-track error and heading error were measured to compare the robot's visual navigation trajectory with the ideal trajectory.

1) Cross-track error: calculated by determining the robot's vertical deviation from predefined reference rows, using the closest reference row for each y-coordinate of the robot's path.

2) Heading error quantified deviations in the robot's orientation relative to the ideal trajectory during the line following stage.

Metrics of 2D mapping: The 2D mapping error refers to the discrepancy between the detected and tracked plant locations in the generated 2D distribution map and their actual ground-truth positions. It is evaluated by comparing the coordinates of the tracked plants with the ground-truth positions provided by the simulation environment. The error is measured both along the crop row (x-direction) and perpendicular to the crop row (y-direction), as well as the overall Euclidean distance error between corresponding plant locations.

Metrics of 3D mapping: After the robot completed its scan of the entire field, the RTAB-MAP framework was used to generate a complete 3D point cloud of the field. To focus on individual crops, the point clouds of specific plants were segmented from the overall 3D mapping results using CloudCompare software. The evaluation of the alignment quality and accuracy of the segmented point cloud involved comparing it to a ground truth point cloud, which was derived from the original mesh model of the plant through uniform sampling.

The assessment metrics included several key measurements:

- 1) ICP Overlap: This metric quantifies the overlap between two point clouds after alignment using the Iterative Closest Point (ICP) algorithm. It indicates the percentage of the source point cloud effectively matched to the target point cloud.
- 2) ICP RMSE: The root mean square error (RMSE) measures the average point-to-point distance error after ICP alignment. This metric provides insight into the accuracy of the alignment process.
- 3) Chamfer Distance: The Chamfer Distance evaluates the average nearest-neighbor distance between two sets of points, serving as a metric for assessing the overall similarity between the point clouds.
- 4) Volume Difference: This metric calculates the difference in volume between two 3D objects. The volume is expressed in the same units as the original data, allowing for precise comparison of object dimensions.

3. Results and Discussion

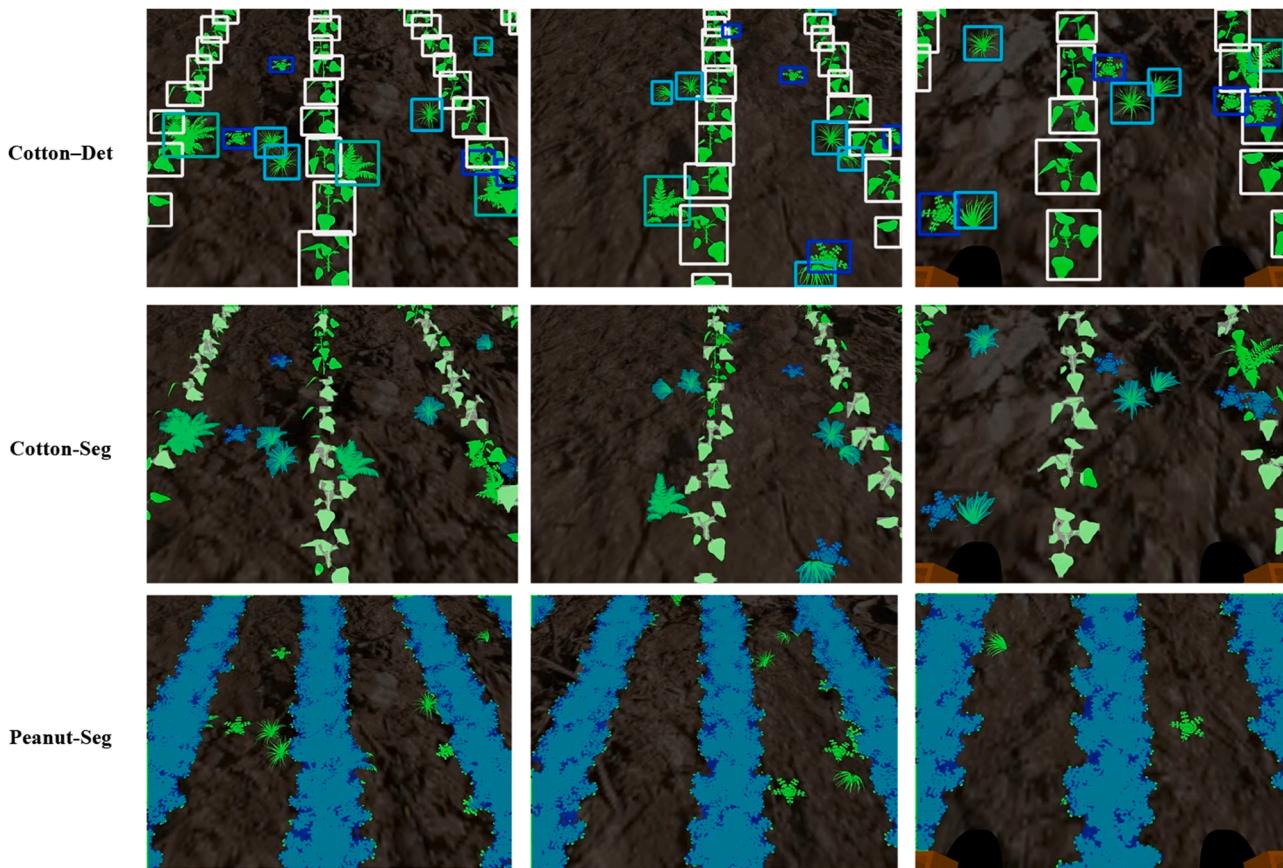
3.1. Visual navigation evaluation

YOLO detector/segmentation: The YOLOv11 model demonstrated high accuracy in both detection and segmentation tasks for cotton and

Table 1

Performance of YOLO model for crop/weeds detection and segmentation.

Field	Method	Class	Performance			
			Precision (%)	Recall (%)	mAP50 (%)	mAP95 (%)
Cotton	Detection	Crop	81.9	97.6	97.9	85.4
		Crabgrass	78.7	87.5	87.0	59.1
		Polygonum	100	89.5	98.7	71.1
	Segmentation	Cirsium	100	83.4	87.7	72.4
		Crop	94.5	94.1	98.1	72.9
		Crabgrass	86.3	82.6	89.7	56.6
Peanut	Segmentation	Polygonum	94.1	95.6	96.3	78.6
		Cirsium	92.6	98.2	97.2	75.3
		Crop	99.7	96.8	99.2	84.1

**Fig. 10.** Illustration of the YOLO detection and segmentation results. From the top to bottom: detection in the cotton field, segmentation in the cotton field, segmentation in the peanut field.

peanut crops (Table 1). For crop detection and segmentation, the model achieved over 97% mAP50, showcasing its ability to precisely identify and segment individual crops and crop rows. This level of accuracy highlights the robustness of the trained detectors, which exhibit strong confidence in reliably detecting individual plants or segmenting crop rows in diverse agricultural scenarios (Fig. 10).

The three types of weeds, including Crabgrass, Polygonum, and Cirsium, served as sources of interference in the detection and segmentation process. The performance of the YOLOv11 model varied across different weed types due to their unique characteristics. Polygonum, with its larger size and well-defined boundaries, was easier to distinguish, allowing the model to achieve higher detection and segmentation accuracy. In contrast, Crabgrass posed significant challenges due to its smaller size and frequent overlap with crops. These characteristics often obscured the entire shape of the weed, making it difficult for the detector to reliably identify and segment it.

Navigation comparison: Overall, all three visual navigation algorithms successfully enable the robot to follow the crop rows and transit to the next one in four simulated fields (Fig. 11). Among the three methods, the robot can track crop rows with a cross-track error of less than 6 cm and a heading error of less than 1 degree for most of the path. Considering the geometric shape and volume of cotton plants, the cross-track errors are almost negligible when the robot's heading is stable. At the end of the row, the robot was able to transition to the next within an average maneuvering space of 2 m. Thus, our proposed visual navigation pipeline allows the robot to enter the next row using a limited maneuvering space, which is often a critical requirement while navigating in a field. It was observed that errors increased when the robot enters to new rows but decreased once the row following controller takes effect. The trajectory error of the row end was larger than that of the other paths.

Among the three algorithms, the segmentation-based method consistently demonstrates the lowest cross-track errors and heading

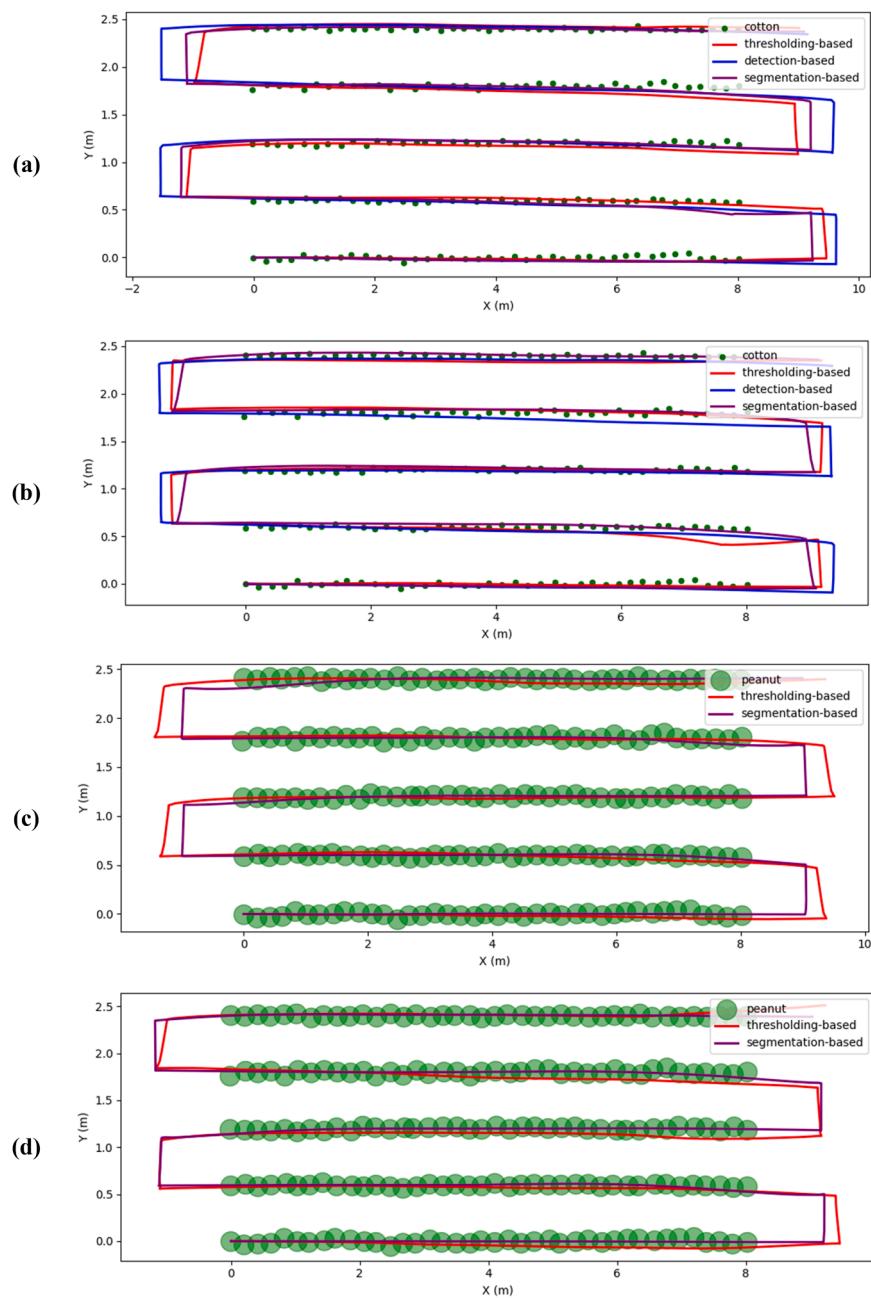


Fig. 11. Performance evaluation of visual navigation in the simulation field. (a) cotton field without weeds, (b) cotton field with weeds, (c) peanut field without weeds, (d) peanut field with weeds.

errors across all scenarios, indicating its robustness and superior accuracy (Fig. 12). The detection-based algorithm performs moderately, generally yielding higher errors compared to segmentation-based methods. The thresholding-based algorithm, however, shows the highest errors in most scenarios, particularly in fields with weeds, suggesting its vulnerability to environmental noise and complexity.

When comparing performance across field conditions, weed presence increased errors for all algorithms. In cotton fields, segmentation-based methods achieved the lowest mean cross-track errors (~ 2.40 cm without weeds and ~ 2.61 cm with weeds), while thresholding-based and detection-based methods showed larger mean errors. In peanut fields, segmentation-based methods excelled even more prominently, with mean errors as low as ~ 0.93 cm (without weeds) and ~ 0.77 cm (with weeds), significantly outperforming the thresholding-based algorithm (~ 4.01 cm with weeds). These results highlight that the

segmentation-based algorithm is the most reliable approach for cross-track error minimization, particularly in challenging field conditions with weeds.

Navigation error analysis: The proposed visual navigation pipeline was developed based on prior knowledge that the crop rows were arranged in multiple parallel lines. It exploits the row structure inherent in the crop fields to guide the robot along the crop row without the need for explicit localization systems, GNSS, or a map of the environment. The simulation experiments demonstrate the feasibility of using only local observations from the on-boarding cameras to navigate. Our pipeline allows the robot to successfully navigate across the entire crop fields row by row and cover the entire field.

However, there are still some limitations and further improvements for the visual navigation pipeline when applying it in real world. First, although the deep-learning-based segmentation method demonstrated

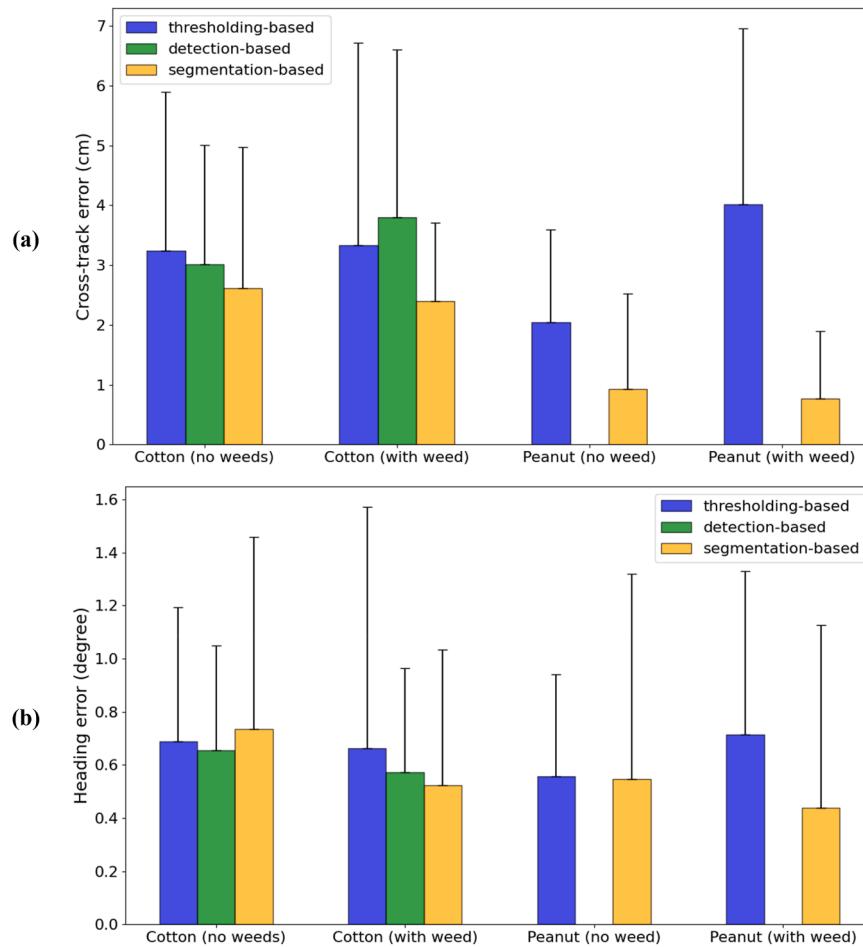


Fig. 12. Comparison of cross-track errors (a) and heading errors (b) between three algorithms (thresholding-based, detection-based, and segmentation-based) across two crops (cotton and peanut) under different field conditions (with and without weeds). The error bars represent the standard deviation.

robustness in typical crop fields where crops are grown separately or overlap within a row, the pipeline struggles under scenarios with sparse or dense weed pressure. The presence of weeds can confuse the segmentation model, leading to incorrect crop row detection and navigation errors. Additionally, the deep learning model requires an extensive and diverse dataset for training to account for the wide variability in field conditions, such as different crop types, growth stages, lighting conditions, and occlusions caused by overlapping vegetation or field machinery. The collection and annotation of such datasets can be time-consuming and labor-intensive, posing a challenge to scalability. Second, the reliance on visual cues alone can be problematic in adverse weather conditions, such as heavy rain, fog, or direct sunlight, which can degrade image quality and affect the accuracy of the segmentation and navigation components. Integrating additional sensor modalities, such as LiDAR or multispectral cameras, could help improve performance under challenging environmental conditions by providing complementary information. Third, the current pipeline assumes that crop rows are arranged in a consistent and parallel structure, which may not always be the case. In real-world fields, irregular planting patterns, curved rows, or damaged areas can disrupt the navigation process. Enhancements such as incorporating spatial awareness through SLAM techniques or learning-based algorithms to adapt to irregularities could improve the robustness of the pipeline.

Moreover, transitions between rows remain a challenge. While the robot successfully navigates along rows, detecting the end of a row and transitioning to the next one without additional localization systems or a pre-mapped field plan can lead to inefficiencies or missed coverage. Developing methods for seamless row transition, such as using

predictive modeling or reinforcement learning, could help address this limitation. Finally, real-time processing remains an area for improvement. The computational demands of deep learning-based segmentation and navigation algorithms may limit the speed and efficiency of the robot, particularly in large-scale fields. Optimizing the pipeline for edge devices, leveraging lightweight neural network architectures, and employing hardware accelerators can enhance real-time performance.

3.2. Field mapping evaluation

2D Mapping: Our proposed 2D distribution maps rely on the tracking outcome and the coordinate transform of the registered point cloud. SORT is an open-source detection-based tracking architecture which uses Kalman filter and Hungarian algorithm to link the bounding boxes and tracks (Bewley, Ge et al. 2016). We directly use the original architecture and deploy our customized YOLOv11-nano detector to generate the bounding boxes. We tested the 2D distribution map generation approach in the simulated cotton fields. We only considered the cotton plants as the tracking objects.

The results showed that all the tracked plants were successfully matched with the corresponding ground-truth, which means our YOLOv11 detector can successfully detect the cotton plants with 100% accuracy and the tracker also can track 200 cotton plants successfully with 100% success rate (Fig. 13a). The error of the plant location is within a reasonable range considering the size of cotton plants. The mean location errors of the plants along the crop row (x error) was 3.90 cm with a standard deviation of 3.21 cm and the error perpendicular to crop rows (y error) was 5.09 cm with a standard deviation of 3.06 cm

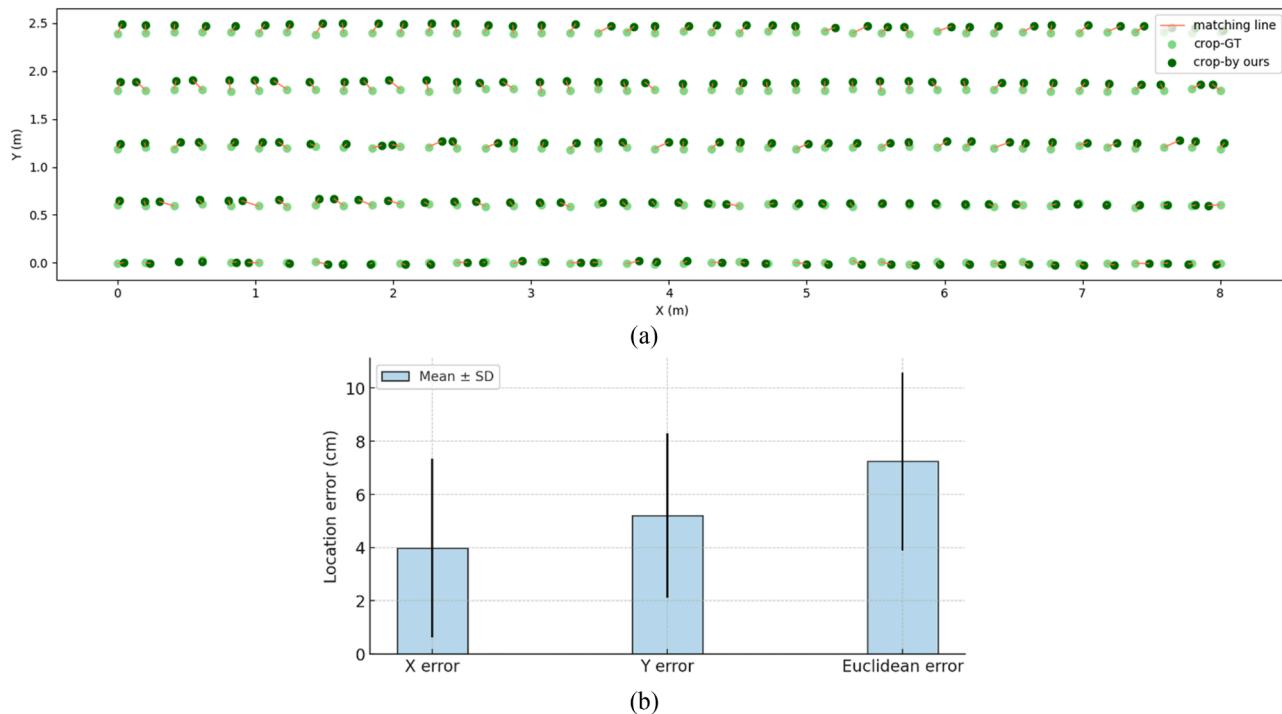


Fig. 13. Performance evaluation of 2D cotton crop mapping (without weeds) using the proposed tracking-based workflow: (a) 2D crop distribution map, (b) boxplot of location error. Dark-green dots present the tracked cotton plants, light-green dots present the ground truth, and red lines are the matching lines between corresponding plants.

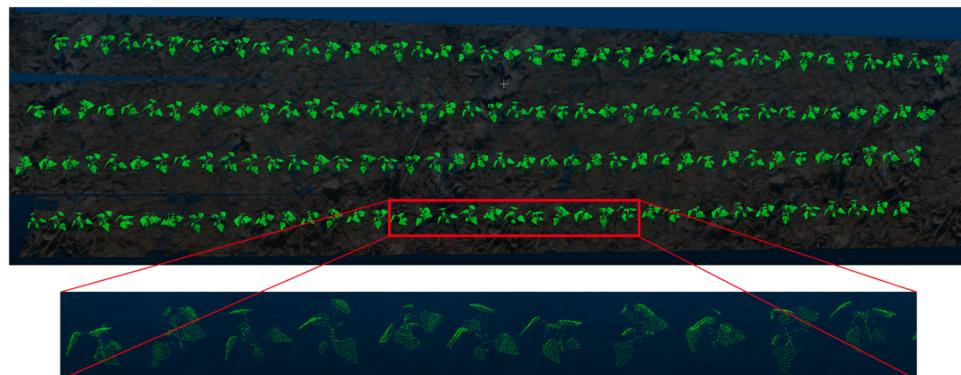


Fig. 14. Demonstration of 3D point cloud using field mapping workflow.

(Fig. 13b). Most of the Euclidean distance between the corresponding crops was less than 9 cm.

3D mapping: Fig. 14 illustrates the 3D map of the cotton field generated by our proposed mapping pipeline. To decrease the overlap point clouds of the same plant, we cropped the image to only include plants of the current followed row. Discrete point clouds were filtered with the CloudCompare software. It was observed that the high-quality point clouds of the field and plants were generated. The leaves and stems of a plant can be easily recognized, which means 3D mapping results are accessible to use for further phenotypic traits analysis.

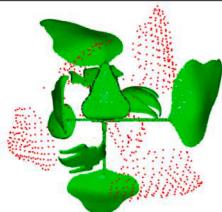
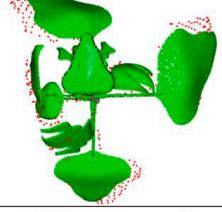
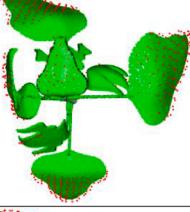
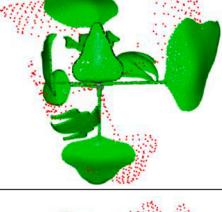
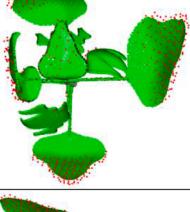
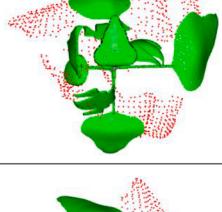
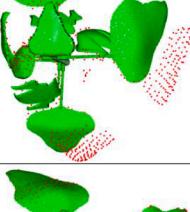
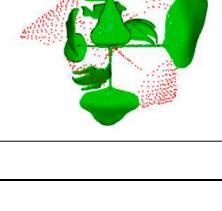
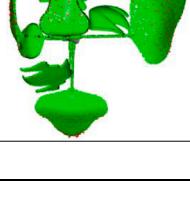
The results demonstrated that the 3D mapping pipeline can generate reliable point cloud, as indicated by high ICP overlap, low RMSE, small Chamfer Distance, and minimal Volume Difference (Table 2). For the majority of the examples with an ICP overlap of 100%, the ICP RMSE values were generally low, ranging from 5.97 to 8.603 mm in well-aligned cases, indicating accurate point-to-point alignment. Chamfer Distances were also consistently small, varying between 10.7 mm and 13.6 mm, demonstrating good overall similarity between the scanned

and ground truth point clouds. Volume differences were minimal, with values between 151,000 mm³ and 262,000 mm³, suggesting the segmented models closely match the structural features of the ground truth. However, the 3D mapping pipeline faced challenges in producing accurate point clouds when odometry drift occurred. In one instance, the pipeline exhibited alignment issues, reflected in a lower ICP overlap of 83.69%, a higher ICP RMSE of 3.83 mm, and increased Chamfer Distance and Volume Difference values of 18.7 mm and 2,574,000 mm³, respectively. This indicates that accurate robot localization is required to get accurate 3D mapping.

Several limitations persist in the proposed mapping pipeline. Accurate 3D mapping heavily relies on low-drift odometry, which remains challenging to achieve despite fusing data from multiple sensors, including the RGB-D camera, IMU, and wheel encoder. This drift results in location errors for all point clouds. For instance, in the top-left corner of the 3D map, significant drift caused a noticeable breach in the mapped area. While RTAB-Map offers loop closure detection to correct odometry drift, the robot's continuous exploration of new areas limited

Table 2

Examples of the point cloud quality evaluation between the ground truth (green mesh model) and our pipeline (red point cloud).

Point clouds	ICP registration	ICP overlap	ICP RMSE (mm)	Chamfer Distance (mm)	Volume Difference (mm ³)
		100%	5.97	010.7	151000
		100%	7.06	13.6	262000
		100%	8.603	13.4	2729000
		83.69%	13.838	18.7	2574000
		100%	2.013	8.4	1701000

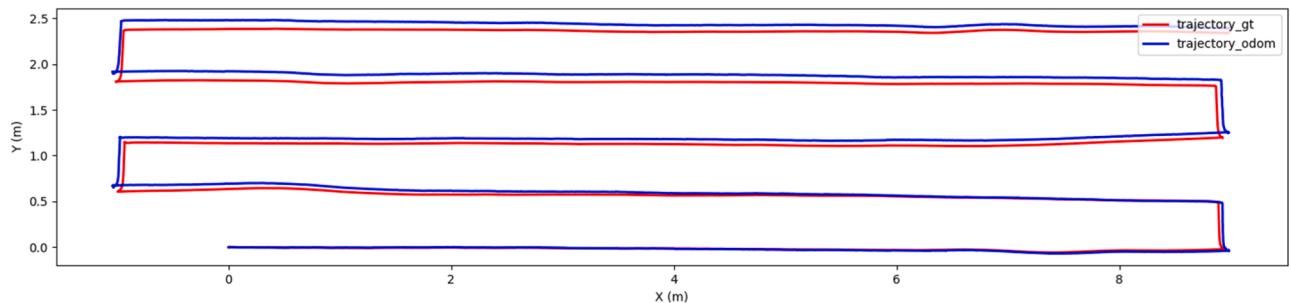


Fig. 15. Analysis of odometry drift along the navigation trajectory. The red line indicates the ground-truth trajectory of the robot, which is based on the simulator's real-time update. The blue one indicates the trajectory from robot's odometry.

the availability of overlapping regions needed to trigger loop closure. Additionally, the simulation environment lacked sufficient discriminative features due to the limited variety of models. The restricted field of view further exacerbated this issue, as the small crop coverage failed to

provide feature similarities above the critical threshold for accurate alignment. Therefore, further development of offline processing technologies is required to calibrate and refine the mapping outcomes.

Mapping Error Analysis: Without absolute global localization (e.g.,



Fig. 16. The illustration of frame sequence of a tracked plant (id 78) with bounding boxes and corresponding central point.

GNSS), relying solely on local data for accurate odometry and location updates is challenging. In our field mapping pipeline, both the 2D distribution map and 3D map depend on precise odometry estimation. This is particularly critical for 3D mapping, where drift exceeding centimeter-level accuracy can result in low-quality plant point clouds.

The primary source of drift occurs during the transition stages, where the robot experiences significant changes in velocity and heading, making odometry highly sensitive to noise (Fig. 15). Once drift begins, its errors accumulate over time. For instance, a slight pose offset during the first transition stage can result in an accumulated location error of 0.05 meters by the second stage, increasing to a maximum of 0.12 meters by the fourth transition stage. Without loop closure techniques, odometry drift cannot be corrected. A potential improvement involves integrating global localization data to enhance odometry estimation. For example, fusing GNSS data with local odometry could significantly reduce drift and improve overall mapping accuracy.

In the 2D mapping pipeline, another source of error in plant location arises from inaccuracies in determining the plant's central location due to partial appearances in the image (Fig. 16). When a plant enters or exits the camera's field of view, only a portion of it may be visible. If this partial plant is detected and treated as a whole, the central point of the tracking bounding box is miscalculated, leading to an incorrect plant center. However, since a plant can appear in multiple image frames, these erroneous plant centers can be filtered out by analyzing the frame sequence and bounding box size.

4. Conclusions

This study introduced MARS-PhenoBot, a Robot Operating System (ROS)-based phenotyping robot, and demonstrated its reliable visual navigation performance in simulation. The proposed visual navigation pipeline leveraged the layout of crop fields to guide the robot along crop rows without relying on explicit localization systems, such as GNSS or environmental maps. It used the field layout to transition from one crop row to another. Experimental results showed that the segmentation-based deep learning approach provided the most robust row detection in both discontinuous and continuous row fields. Additionally, the field mapping approach for generating 2D and 3D maps was evaluated. The 2D distribution map achieved a local error of less than 9 cm for most plants, demonstrating its practicality for crop mapping. However, the 3D mapping quality was impacted by odometry drift, highlighting an area for further improvement. The methodology developed in this study is scalable and holds promise for deployment on real agricultural robots to automate phenotyping tasks in crop fields.

Ethical statement

This study involved no human participants or animals. All experiments were performed in accordance with institutional and national guidelines for agricultural research. The authors ensured proper disposal of biological materials to minimize environmental impact.

CRediT authorship contribution statement

Zhengkun Li: Conceptualization, Data curation, Investigation, Methodology, Project administration, Validation, Visualization, Writing – original draft, Writing – review & editing. **Rui Xu:** Data curation, Investigation, Supervision, Visualization, Writing – review & editing. **Changying Li:** Conceptualization, Project administration, Resources,

Supervision, Writing – review & editing. **Longsheng Fu:** Conceptualization, Formal analysis, Project administration, Resources.

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.

Acknowledgements

This study was partially supported by USDA NIFA (Award No: 2023-67021-40646), the Cotton Incorporated, and the Hatch Project (FLA-ABE-006451).

Data availability

Data will be made available on request.

References

- [1] M. Junaid, A. Gokce, Global agricultural losses and their causes, *Bull. Biol. Allied Sci. Res.* 2024 (1) (2024) 66.
- [2] L. Kumar, N. Chhogyal, T. Gopalakrishnan, M.K. Hasan, S.L. Jayasinghe, C. S. Kariyawasam, B.K. Kogo, S. Ratnayake, Climate change and future of agri-food production. *Future foods*, Elsevier, 2022, pp. 49–79.
- [3] M.G. Muluneh, Impact of climate change on biodiversity and food security: a global perspective—A review article, *Agric. Food Secur.* 10 (1) (2021) 1–25.
- [4] D. Tilman, C. Balzer, J. Hill, B.L. Befort, Global food demand and the sustainable intensification of agriculture, *Proc. Natl. Acad. Sci. USA* 108 (50) (2011) 20260–20264, <https://doi.org/10.1073/pnas.1116437108>.
- [5] N. Shakoor, S. Lee, T.C. Mockler, High throughput phenotyping to accelerate crop breeding and monitoring of diseases in the field, *Curr. Opin. Plant Biol.* 38 (2017) 184–192, <https://doi.org/10.1016/j.pbi.2017.05.006>.
- [6] A.K. Mahlein, Plant disease detection by imaging sensors - parallels and specific demands for precision agriculture and Plant phenotyping, *Plant Dis.* 100 (2) (2016) 241–251, <https://doi.org/10.1094/PDIS-03-15-0340-FE>.
- [7] R. Qiu, S. Wei, M. Zhang, H. Li, H. Sun, G. Liu, M. Li, Sensors for measuring plant phenotyping: A review, *Int. J. Agric. Biol. Eng.* 11 (2) (2018) 1–17, <https://doi.org/10.25165/j.ijabe.20181102.2969>.
- [8] B. Benet, C. Dubos, F. Maupas, G. Malatesta, R. Lenain, Development of autonomous robotic platforms for sugar beet crop phenotyping using artificial vision, in: *AGENG Conference 2018*, 2018.
- [9] Y. Jiang, C. Li, J.S. Robertson, S. Sun, R. Xu, A.H. Paterson, GPhenoVision: A ground mobile system with multi-modal imaging for field-based high throughput phenotyping of cotton, *Sci. Rep.* 8 (1) (2018), <https://doi.org/10.1038/s41598-018-19142-2>.
- [10] R. Ludovisi, F. Tauro, R. Salvati, S. Khouri, G. Mugnozza Scarascia, A. Harfouche, UAV-based thermal imaging for high-throughput field phenotyping of black poplar response to drought, *Front. Plant Sci.* 8 (2017) 1681, <https://doi.org/10.3389/fpls.2017.01681>.
- [11] Y. Bao, D.S. Shah, L. Tang, 3D Perception-based collision-free robotic leaf probing for automated indoor plant phenotyping, *Trans. ASABE* 61 (3) (2018) 859–872, <https://doi.org/10.13031/trans.12653>.
- [12] Y. Bao, L. Tang, M.W. Breitzman, M.G. Salas Fernandez, P.S. Schnable, Field-based robotic phenotyping of sorghum plant architecture using stereo vision, *J. Field. Robot.* 36 (2) (2019) 397–415, <https://doi.org/10.1002/rob.21830>.
- [13] T. Mueller-Sim, M. Jenkins, J. Abel, G. Kantor, The Robotanist: a ground-based agricultural robot for high-throughput crop phenotyping, in: *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017.
- [14] N. Virlet, K. Sabermanesh, P. Sadeghi-Tehrani, M.J. Hawkesford, Field scanalyzer: an automated robotic field phenotyping platform for detailed crop monitoring, *Funct. Plant Biol.* 44 (1) (2016) 143–153, <https://doi.org/10.1071/FP16163>.
- [15] L. Feng, S.S. Chen, C. Zhang, Y.C. Zhang, Y. He, A comprehensive review on recent applications of unmanned aerial vehicle remote sensing with various sensors for high-throughput plant phenotyping, *Comput. Electron. Agric.* 182 (19) (2021) 106033, <https://doi.org/10.1016/j.compag.2021.106033>.
- [16] L. He, W. Fang, G. Zhao, Z. Wu, L. Fu, R. Li, Y. Majeed, J. Dhupia, Fruit yield prediction and estimation in orchards: A state-of-the-art comprehensive review for both direct and indirect methods, *Comput. Electron. Agric.* 195 (2022) 106812.

- [17] Z. Rui, Z. Zhang, M. Zhang, A. Azizi, C. Igathinathane, H. Cen, S. Vougioukas, H. Li, J. Zhang, Y. Jiang, High-throughput proximal ground crop phenotyping systems—A comprehensive review, *Comput. Electron. Agric.* 224 (2024) 109108.
- [18] R. Xu, C. Li, A review of high-throughput field phenotyping systems: focusing on ground robots, *Plant Phenomics*. 2022 (2022).
- [19] A. Ruckelshausen, P. Biber, M. Dorna, H. Gremmes, R. Klose, A. Linz, R. Rahe, R. Resch, M. Thiel, D. Trautz, BoniRob: an autonomous field robot platform for individual plant phenotyping, *Precision agriculture'09*, Wageningen Academic, 2009, pp. 841–847.
- [20] DeBruin, J., Aref, T., Tolosa, S. T., Hensley, R., Underwood, H., McGuire, M., Soman, C., Nyström, G., Parkinson, E., & Li, C. (2024). Breaking the field phenotyping bottleneck in maize with autonomous robots.
- [21] Sivakumar, A. N., Magistri, F., Gasparino, M. V., Behley, J., Stachniss, C., & Chowdhary, G. (2024). AdaCropFollow: self-supervised online adaptation for visual under-canopy navigation. arXiv preprint arXiv:2410.12411.
- [22] S. Cubero, E. Marco-Noales, N. Aleixos, S. Barbe, J. Blasco, RobHortic: A field robot to detect pests and diseases in horticultural crops by proximal sensing, *Agriculture* 10 (7) (2020) 276, <https://doi.org/10.3390/agriculture10070276>.
- [23] A. Shafeiekhani, S. Kadam, F.B. Fritschl, G.N. DeSouza, Vinobot and Vinoculer: two robotic platforms for high-throughput field phenotyping, *Sensors* 17 (1) (2017) 214, <https://doi.org/10.3390/s17010214>.
- [24] Smitt, C., Halstead, M., Zaenker, T., Bennewitz, M., & McCool, C. J. a. p. a. (2020). PATHoBot: A robot for glasshouse crop phenotyping and intervention. <http://arxiv.org/abs/2010.16272>.
- [25] L. Grimstad, K. Skattum, E. Solberg, G. Loureiro, P.J. From, Thorvald II configuration for wheat phenotyping, in: *Proceedings of the IROS Workshop on Agri-Food Robotics: Learning from Industry*, 2017.
- [26] R. Xu, C. Li, A modular agricultural robotic system (MARS) for precision farming: concept and implementation, *J. Field. Robot.* (2022).
- [27] Farm-ng, The Amiga Robot, Retrieved Dec 20 from, <https://farm-ng.com/>, 2024.
- [28] Z. Zhang, E. Kayacan, B. Thompson, G. Chowdhary, High precision control and deep learning-based corn stand counting algorithms for agricultural robot, *Autonomous Robot.* 44 (7) (2020) 1289–1302, <https://doi.org/10.1007/s10514-020-09915-y>.
- [29] L. Xiang, L. Tang, J. Gai, L. Wang, PhenoStereo: a high-throughput stereo vision system for field-based plant phenotyping - with an application in sorghum stem diameter estimation</i>>, in: 2020 ASAEB Annual International Virtual Meeting, July 13–15, 2020, 2020, <https://elibrary.asabe.org/azdez.asp?JID=5&AID=51546&t=2&CID=virt2020&downPDF=Y&directPDF=Y>.
- [30] S.N. Young, E. Kayacan, J.M. Peschel, Design and field evaluation of a ground robot for high-throughput phenotyping of energy sorghum, *Precis. Agric.* 20 (4) (2018) 697–722, <https://doi.org/10.1007/s11119-018-9601-6>.
- [31] S. Vijayarangan, P. Sodhi, P. Kini, J. Bourne, S. Du, H. Sun, B. Poczos, D. Apostolopoulos, D. Wettergreen, High-throughput robotic phenotyping of energy sorghum crops, *Field and Service Robotics*, 2018.
- [32] Y. Bai, B. Zhang, N. Xu, J. Zhou, J. Shi, Z. Diao, Vision-based navigation and guidance for agricultural autonomous vehicles and robots: a review, *Comput. Electron. Agric.* 205 (2023) 107584.
- [33] Z. Li, C. Li, Robotic vegetable production. *Encyclopedia of Smart Agriculture Technologies*, Springer, 2023, pp. 1–12.
- [34] M. Perez-Ruiz, J. Martínez-Guante, S.K. Upadhyaya, High-precision GNSS for agricultural operations. *GPS and GNSS Technology in Geosciences*, Elsevier, 2021, pp. 299–335.
- [35] T. Purfürst, Evaluation of static autonomous GNSS positioning accuracy using single-, dual-, and tri-frequency smartphones in forest canopy environments, *Sensors* 22 (3) (2022) 1289. [https://mdpi-res.com/d.attachment/sensors/s-22-01289/article_deploy/sensors-22-01289-v2.pdf?version=1644408496](https://mdpi-res.com/d.attachment/sensors/sensor-s-22-01289/article_deploy/sensors-22-01289-v2.pdf?version=1644408496).
- [36] H. Nehme, C. Aubry, T. Solatges, X. Savatier, R. Rossi, R. Boutteau, Lidar-based structure tracking for agricultural robots: application to autonomous navigation in vineyards, *J. Intell. Robot. Syst.* 103 (4) (2021) 61.
- [37] G. Rivera, R. Porras, R. Florencio, J.P. Sánchez-Solís, LiDAR applications in precision agriculture for cultivating crops: a review of recent advances, *Comput. Electron. Agric.* 207 (2023) 107737.
- [38] N.T. Dang, N.T. Luy, LiDAR-based online navigation algorithm for an autonomous agricultural robot, *J. Control Eng. Appl. Info.* 24 (2) (2022) 90–100.
- [39] F.B. Malavazi, R. Guyonneau, J.-B. Fasquel, S. Lagrange, F. Mercier, LiDAR-only based navigation algorithm for an autonomous agricultural robot, *Comput. Electron. Agric.* 154 (2018) 71–79.
- [40] P.M. Blok, K. van Boheemen, F.K. van Evert, J. Ijsselmuideren, G.H. Kim, Robot navigation in orchards with localization based on particle filter and Kalman filter [Article], *Comput. Electron. Agric.* 157 (2019) 261–269, <https://doi.org/10.1016/j.compag.2018.12.046>.
- [41] S.M. Farhan, J. Yin, Z. Chen, M.S. Memon, A comprehensive review of LiDAR applications in crop management for precision agriculture, *Sensors* 24 (16) (2024) 5409.
- [42] J. Qu, Z. Qiu, L. Li, K. Guo, D. Li, Map construction and positioning method for LiDAR SLAM-based navigation of an agricultural field inspection robot, *Agronomy* 14 (10) (2024) 2365.
- [43] M.R. Karim, M.N. Reza, H. Jin, M.A. Haque, K.-H. Lee, J. Sung, S.-O. Chung, Application of LiDAR sensors for crop and working environment recognition in agriculture: A review, *Remote Sens* 16 (24) (2024) 4623.
- [44] C. Ban, L. Wang, R. Chi, T. Su, Y. Ma, A camera-LiDAR-IMU fusion method for real-time extraction of navigation line between maize field rows, *Comput. Electron. Agric.* 223 (2024) 109114.
- [45] C. Han, W. Wu, X. Luo, J. Li, Visual navigation and obstacle avoidance control for agricultural robots via LiDAR and camera, *Remote Sens.* 15 (22) (2023) 5402.
- [46] Z. Zhao, Y. Zhang, L. Long, Z. Lu, J. Shi, Efficient and adaptive lidar–visual–inertial odometry for agricultural unmanned ground vehicle, *Int. J. Adv. Robot. Syst.* 19 (2) (2022), 17298806221094925.
- [47] S. Zhang, Y. Liu, K. Xiong, Y. Tian, Y. Du, Z. Zhu, M. Du, Z. Zhai, A review of vision-based crop row detection method: focusing on field ground autonomous navigation operations, *Comput. Electron. Agric.* 222 (2024) 109086.
- [48] T. Wang, B. Chen, Z. Zhang, H. Li, M. Zhang, Applications of machine vision in agricultural robot navigation: a review, *Comput. Electron. Agric.* 198 (2022) 107085.
- [49] A. English, P. Ross, D. Ball, P. Corke, Ieee, Vision based guidance for robot navigation in agriculture, in: *IEEE International Conference on Robotics and Automation ICRA [2014 iee international conference on robotics and automation (icra)]*. IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, PEOPLES R CHINA, 2014.
- [50] W. Winterhalter, F.V. Fleckenstein, C. Dornhege, W. Burgard, Crop row detection on tiny plants with the pattern hough transform, *IEEe Robot. Autom. Lett.* 3 (4) (2018) 3394–3401.
- [51] J. Shi, Y. Bai, Z. Diao, J. Zhou, X. Yao, B. Zhang, Row detection BASED navigation and guidance for agricultural robots and autonomous vehicles in row-crop fields: methods and applications, *Agronomy* 13 (7) (2023) 1780.
- [52] L. Lei, Q. Yang, L. Yang, T. Shen, R. Wang, C. Fu, Deep learning implementation of image segmentation in agricultural applications: a comprehensive review, *Artif. Intell. Rev.* 57 (6) (2024) 149.
- [53] M. Bakken, V.R. Ponnambalam, R.J. Moore, J.G.O. Gjevestad, P.J. From, Robot-supervised learning of crop row segmentation, in: *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021.
- [54] R. De Silva, G. Cielniak, G. Wang, J. Gao, Deep learning-based crop row detection for infiel navigation of agri-robots, *J. Field. Robot.* (2023).
- [55] J. Wei, M. Zhang, C. Wu, Q. Ma, W. Wang, C. Wan, Accurate crop row recognition of maize at the seedling stage using lightweight network, *Int. J. Agric. Biol. Eng.* 17 (1) (2024) 189–198.
- [56] H. Yuan, Y. Liang, Z. Li, Development of autonomous navigation system based on neural network and visual servoing for row-crop tracking in vegetable greenhouses, *Smart Agric. Technol.* 9 (2024) 100572.
- [57] Z. Zhang, L. Huang, B.-H. Tang, W. Le, M. Wang, J. Cheng, Q. Wu, MATNet: multiattention Transformer network for cropland semantic segmentation in remote sensing images, *Int. J. Digit. Earth.* 17 (1) (2024) 2392845.
- [58] Ahmadi, A., Halsted, M., & McCool, C. J. a. p. a. (2021). Towards autonomous crop-agnostic visual navigation in arable fields.
- [59] A. Ahmadi, L. Nardi, N. Chebrolu, C. Stachniss, Visual servoing-based navigation for monitoring row-crop fields, in: *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020.
- [60] J.B. Li, R.G. Zhu, B.Q. Chen, Image detection and verification of visual navigation route during cotton field management period, *Int. J. Agric. Biol. Eng.* 11 (6) (2018) 159–165, <https://doi.org/10.25165/j.ijabe.20181106.3976>.
- [61] E. Liu, J. Monica, K. Gold, L. Cadle-Davidson, D. Combs, Y. Jiang, Vision-based vineyard navigation solution with automatic annotation, in: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023.
- [62] C. Mwitwa, G.C. Rains, The integration of GPS and visual navigation for autonomous navigation of an Ackerman steering mobile robot in cotton fields, *Front. Robot. AI.* 11 (2024) 1359887.
- [63] de Silva, R., Cielniak, G., & Gao, J. (2023). Crop row switching for vision-based navigation: A comprehensive approach for efficient Crop field navigation. arXiv preprint arXiv:2309.11989.
- [64] A. Atefi, Y.F. Ge, S. Pitla, J. Schnable, Robotic technologies for high-throughput plant phenotyping: contemporary reviews and future perspectives, *Front. Plant Sci.* 12 (2021) 611940, <https://doi.org/10.3389/fpls.2021.611940>.
- [65] M.Y. Chen, Y.C. Tang, X.J. Zou, Z.F. Huang, H. Zhou, S.Y. Chen, 3D global mapping of large-scale unstructured orchard integrating eye-in-hand stereo vision and SLAM [Article], *Comput. Electron. Agric.* 187 (16) (2021) 106237, <https://doi.org/10.1016/j.compag.2021.106237>.
- [66] C. Campos, R. Elvira, J.J.G. Rodríguez, J.M. Montiel, J.D. Tardós, Orb-slam3: an accurate open-source library for visual, visual-inertial, and multimap slam, *Ieee Trans. Robot.* 37 (6) (2021) 1874–1890.
- [67] M. Labb  , F. Michaud, RTAB-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation, *J. Field. Robot.* 36 (2) (2019) 416–446.
- [68] J. Cremona, J. Civera, E. Kofman, T. Pire, GNSS-stereo-inertial SLAM for arable farming, *J. Field. Robot.* 41 (7) (2024) 2215–2225.
- [69] H. Ding, B. Zhang, J. Zhou, Y. Yan, G. Tian, B. Gu, Recent developments and applications of simultaneous localization and mapping in agriculture, *J. Field. Robot.* 39 (6) (2022) 956–983.
- [70] W.B. Dong, P. Roy, V. Isler, Semantic mapping for orchard environments by merging two-sides reconstructions of tree rows [Article], *J. Field. Robot.* 37 (1) (2020) 97–121, <https://doi.org/10.1002/rob.21876>.
- [71] Y. Fan, Z. Feng, A. Mannan, T.U. Khan, C. Shen, S. Saeed, Estimating tree position, diameter at breast height, and tree height in real-time using a mobile phone with RGB-D SLAM, *Remote Sens.* 10 (11) (2018) 1845.
- [72] N. Habibie, A.M. Nugraha, A.Z. Anshori, M.A. Ma'sum, W. Jatmiko, Fruit mapping mobile robot on simulated agricultural area in Gazebo simulator using simultaneous localization and mapping (SLAM), in: *2017 International Symposium on Micro-NanoMechatronics and Human Science (MHS)*, 2017.
- [73] F.K. Khanzada, E. Delavarci, W. Jeong, Y.S. Cho, J. Kwon, Comparative study on simulated outdoor navigation for agricultural robots, *Sensors* 24 (8) (2024) 2487.

- [74] A.K. Nellithimaru, G.A. Kantor, ROLS: robust object-level SLAM for grape counting, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2019.
- [75] S.P. Sun, C.Y. Li, P.W. Chee, A.H. Paterson, C. Meng, J.Y. Zhang, P. Ma, J. S. Robertson, J. Adhikari, High resolution 3D terrestrial LiDAR for cotton plant main stalk and node detection [Article], Comput. Electron. Agric. 187 (10) (2021) 106276, <https://doi.org/10.1016/j.compag.2021.106276>.
- [76] J. Matas, C. Galambos, J. Kittler, Robust detection of lines using the progressive probabilistic hough transform, Comput. Vision Image Understanding 78 (1) (2000) 119–137.
- [77] Aharon, N., Orfaig, R., & Bobrovsky, B.-Z. (2022). BoT-SORT: robust associations multi-pedestrian tracking. arXiv preprint arXiv:2206.14651.
- [78] Jocher, G., Chaurasia, A., & Qiu, J. (2024). Ultralytics YOLO (Version 8.3.13) [Computer software]. Retrieved October 2024 from.
- [79] M. Everingham, L. Van Gool, C.K. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, Int. J. Comput. Vis. 88 (2010) 303–338.