Article

Robust Guidance and Precise Spraying of a Four-wheeled Agricultural Robot based on Deep Learning Approach

Chung-Liang Chang * and Hung-Wen Chen

Posted Date: 23 October 2023

doi: 10.20944/preprints202310.1427.v1

Keywords: Agricultural robot; deep learning; precision spraying; PID; fuzzy logic; autonomous guidance.

Article

Robust Guidance and Precise Spraying of a Four-Wheeled Agricultural Robot based on Deep Learning Approach

Chung-Liang Chang * and Hung-Wen Chen

Department of Biomechatronics Engineering, National Pingtung University of Science and Technology, Neipu 91201, Taiwan

* Correspondence: chungliang@mail.npust.edu.tw; Tel.: +886-8-7703202 (ext. 7586)

Abstract: This paper presents a deep learning-based multi-guidance line detection approach, enabling an autonomous four-wheeled agricultural robot to navigate strip farmland. The integration of Proportional-Integral-Derivative (PID) and Fuzzy Logic Control (FLC) systems optimizes the velocity and heading angle of robot, facilitating smooth navigation along identified guidance line. Enhance cornering maneuverability with real-time kinematics (RTK)-assisted Global Navigation Satellite System (GNSS) (RTK-GNSS) positioning. Additionally, the spraying system combined with deep learning can effectively identify weeds and crop nutrient deficiencies to achieve precise spraying. The guidance system prioritizes irrigation lines for navigation, with additional guidance from crop and furrow lines. Trigonometric analysis determines the angular deviation between the identified guidance line and the vertical line of the top view image. Experiments under diverse weather conditions demonstrate the stable navigation of robot at 12.5 cm/s, achieving up to 90% accuracy in weed and nutrient deficiency identification. The spraying accuracy for weeds and deficiencies averages 87% and 73%, respectively, underscoring the system's contribution to sustainable and precision horticulture practices.

Keywords: Agricultural robot; deep learning; precision spraying; PID; fuzzy logic; autonomous guidance

1. Introduction

Amidst the swift surge of smart technologies, their integration into agriculture to combat labor scarcity and an aging workforce is critical. Contemporary practices in crop management, ranging from surveillance to irrigation and fertilization, predominantly rely on manual effort. This dependency not only underscores the necessity for continuous human intervention but also accentuates the risks associated with human error, potentially triggering considerable agricultural setbacks. The diversity in farmland sizes necessitates tailored equipment solutions. Investing in oversized machinery carries implications beyond initial costs, factoring in insurance and depreciation, thereby inadvertently inflating operational expenses. The proven efficacy of automation in performing monotonous tasks has seen its adoption across various sectors, with its applicability in agriculture being equally comprehensive, encompassing activities like weeding and harvesting [1]. Traditional small-scale agricultural robots were designed to navigate based on sensory feedback, apt for structured environments [2]. However, with precision agriculture's evolution, technologies like Real-Time Kinematic (RTK)-assisted Global Navigation Satellite System (GNSS) (RTK-GNSS) and machine vision have assumed central roles in navigation automation. RTK-GNSS has been pivotal in offering precise metrics for positioning, velocity, and timing, aiding in the meticulous choreography of robot trajectories [3,4]. This precision allows for the preprogramming of task sequences in familiar terrains, promoting autonomous fieldwork and refined error mitigation strategies [5].

 doi:10.20944/preprints202310.1427.v1

Moreover, machine vision represents a cost-effective, maintainable, and versatile alternative to human supervision for object recognition and tracking, revolutionizing image processing [6]. The recent trend has seen a proliferation of compact agricultural robots leveraging machine vision, particularly in identifying navigational paths for field operations [7–11], signifying a paradigm shift in agricultural practices. Choi et al. developed a morphology-based approach for extracting guidance lines in rice fields, enabling autonomous weeding robots to operate without damaging crops [7]. Their method initially involved grayscaling images, followed by Otsu thresholding and thinning processes to delineate plant edges. Experimental results demonstrated that the proposed algorithm exhibited robust performance, with guidance lines achieving high precision of less than 1° deviation. Addressing the issue of indistinct navigation lines in cornfields was the focus of Suriyakoon et al. [8]. Due to problems such as exposure variations or crop obstructions, traditional image processing often failed, rendering navigation lines indiscernible. By using the soil in images as guiding points through specialized image processing, the study found that navigation lines could be accurately identified, even when parts of the frame were obscured by corn leaves or shadows, or the image's horizon was tilted.

Bonadiesa et al. proposed two methodologies for autonomous guidance of unmanned vehicles in fields [9]. The first method identified the edges of paths between crops, while the second recognized the lateral crop lines. Post-identification, Hough transform operation were utilized for edge detection, and the median lines between them served as navigation paths, guiding unmanned vehicles through field operations autonomously. A navigation path-fitting method for agricultural robots was introduced by Chen et al., enhancing the traditional Hough transform's computational efficiency and addressing precision concerns related to the least squares method [10]. Following the segmentation of plants and soil via grayscale processing, predictive points were marked through a regression equation, and path fitting was conducted using Hough transform. The results revealed a maximum deviation of less than 0.5° for navigation paths based on the Hough transform, offering significant improvements over the 10.25° average error from the Least Squares method and reducing the most considerable time consumption to 17.92ms, saving 35.20 ms.

Ruan et al. proposed a classification algorithm that leverages YOLO-R and density-based spatial clustering of applications with noise (DBSCAN) to determine the number of crop rows and individual crops within each row [12]. Finally, a least squares method is applied to fit the crop row lines, achieving a crop row recognition rate of at least 89.8%. In the realm of guidance line recognition, two predominant methods are the Hough transform and the least squares method [13]. The Hough transform stands out for its robust interference resistance and its proficiency in extracting edge lines to derive navigation paths. On the other hand, the least squares method offers a mathematical approach to fit navigation paths with high precision. Both methods, however, can be sensitive to changes in illumination, affecting factors like color, brightness, saturation, contrast, reflections, shadows, and noise. Such variability may hinder the accurate extraction of guiding lines. Moreover, fluctuating wind conditions in fields can induce plant motion, resulting in blurred images and potentially misidentifying crop center points [14–17]. With the advancements in high-speed computing technology, employing deep learning for navigation line extraction has gained traction [18,19]. Hu and Huang proposed a method for extracting crop row center lines based on an enhanced Tiny-YoLov4, where MobileNetv3 replaces the original CSPDarknet53 structure in the YoLov4 model [18]. Images within multiple detection frames undergo binarization and mean filtering algorithms to extract crop feature points within the frames, with the final center line fitted using the least squares method. Furthermore, de Silva et al. introduced a U-Net deep learning model capable of detecting crops under favorable weather conditions, good lighting, minimal weeds, and when crops are planted in neat, uninterrupted rows [19]. The Hough transform is then applied to identify guidance lines. These advancements underscore the ongoing innovation in agricultural technology, with deep learning and advanced image processing taking the lead in improving precision and efficiency in the field.

Deep learning methods are also frequently employed in robotics for the identification of weeds and crops. Various research endeavors highlight the effectiveness of these advanced techniques in

 doi:10.20944/preprints202310.1427.v1

precision agriculture. Hu et al. utilized the YOLOv4 model to detect 12 types of weeds in rice fields, with experimental results showcasing a detection accuracy of 97%, a recall rate of 81%, an F1-score of 0.89, and an average detection time of 377ms [20]. Ruigrok et al. focused on the application of a weed detection algorithm for plant spraying [21]. They employed a YOLOv3 model trained for this purpose, achieving accurate detection of 83% of potato plants. Field tests demonstrated effective control of 96% of weeds, with only a 3% error rate in crop termination. Presented by Chang et al. is a weed removal robot that employs artificial intelligence techniques [22]. The YOLOv3 model was used for weed identification, and the results demonstrated a weed recognition rate of 90.7%. The weeding robot achieved an efficiency of 88.6% in removing weeds at a speed of 15cm per minute. Wang et al. applied the YOLOv5 methodology for weed detection, with their model demonstrating an accuracy and recall rate of 95% and 90%, respectively [23]. Chen et al. employed the YoLo-sesame model to identify weeds and crops in sesame fields, achieving an average precision rate (mAP) of 96.1% at a frame rate of 36.8 frames per second (fps) [24]. Umar Farooq et al. introduced a lightweight deep learning model, Tiny-YOLOv4, for weed detection, addressing the trade-off between algorithmic cost-effectiveness and performance [25].

Utilizing the YOLOv3 model for weed detection, as detailed by Ruigrok et al., they trained it on image data from 20 different fields and tested it in five different arable field [26]. The results indicated that increasing the variance in training data while keeping the sample size constant can reduce the generalization error during detection. Numerous heading control methods have been disclosed in the field of agricultural robotics. Suriyakoon et al. proposed a method for autonomous navigation in cornfields [27]. This approach combines image processing with a combination of Proportional-Integral-Derivative (PID) and fuzzy logic control techniques. MATLAB was employed for simulating the robot's movement in the field using a speed differential model. Specifically, PID was used to regulate the wheel speeds of robot, while fuzzy logic methods were applied to control the robot's heading, ensuring its autonomous navigation in the cornfield. Qiu et al. introduced a four-wheel-drive agricultural mobile robot based on Ackermann Steering Principle (ASP) technology [28]. Initially, a control model was employed to estimate the steering angles of the front and rear wheels. Subsequently, the optimal position for the steering center was determined based on the minimal deviation of the inner steering angle. Finally, the linear velocity of robot was generated according to ASP principles. Experimental results demonstrated that the enhanced model was capable of reducing the overall motor energy consumption and minimizing instances of robot slippage during testing. Bonadies et al. developed an unmanned ground vehicle for autonomous navigation in fields using visual guidance [29]. Combining PID with fuzzy logic control enables vehicle to navigate autonomously in a farmland.

Tu et al. designed a sliding mode control (SMC) and inverse calculation method based on kinematics, front wheel steering and coordinated steering models to enhance the robustness of the system [30]. Testing in various environments demonstrated the stability and robustness of the integrated controller in tracking trajectories. Zhang et al. proposed a state-feedback navigation control system based on the Ackermann steering model for a single-cylinder diesel track vehicle [31]. The system utilizes the Ackermann steering control model in conjunction with a PWM-regulated steering proportional control approach. Path points are delineated using RTK-GNSS, and cubic spline interpolation is applied for the smoothing of these points. An effective guidance and control system for agricultural robots necessitates the precise execution of tasks such as spraying and weeding. Employing machine vision integrated with deep learning methodologies for the identification of distinct agronomic elements, such as crops, proves to be a promising strategy for navigational guidance.

Nevertheless, a significant proportion of existing methods remain sequestered in the realms of offline simulation or laboratory experimentation, with minimal empirical evidence supporting their applicability in real-world agricultural operations. Additionally, there is a conspicuous deficiency in literature concerning the in-field testing and validation of these integrated methodologies for steering control and task execution. Compounding these challenges is the reliance on singular crop types for training datasets, thereby critically hampering the universality and adaptability of the recognition

 doi:10.20944/preprints202310.1427.v1

models. This study presents a machine vision-based method for the autonomous guidance and field operations of agricultural robots, leveraging deep learning for guidance, control, and operational systems. The proposed system can automatically identify potential navigation paths on field ridges, modulating travel speed and directional angles using PID controllers and fuzzy logic. Adopting the YoLov4-based architecture, the system is tailored for various field recognition tasks such as crop identification, drip irrigation line detection, field ridge recognition, weed detection, and crop nutrient deficiency identification. In terms of field operations, the robot's spraying mechanism is designed to accurately dispense liquid fertilizer and herbicides to crops showing nutrient deficiencies and to target weeds.

The organization of this paper is as follows: In Section 2, the methodology is presented, including the configuration of various modules inside the robot, the motion model and steering mode of the four-wheeled mobile robot, path planning during field operations, visual guidance methods for generating field row lines, and control methods for robot speed and heading. Section 3 discusses the experimental results, including tests on the robot's autonomous guidance, weed and nutrient-deficient crop identification, and performance tests of the spraying system. Finally, Section 4 provides the conclusion, summarizing the key findings of the study.

2. Methodology

In Section 2.1, the design method and motion model of the agricultural robot are presented. Section 2.2 covers path planning and the coordinate conversion method of the robot's positioning system. The field-based row detection method is described in Section 2.3. Finally, Section 2.4 demonstrates the speed and heading control method of the robot.

2.1. Agriculture Robot Design

This section presents a stable guidance and operation method for a four-wheeled agricultural robot. Figure 1 shows the appearance of the robot in a four-wheeled mode. During autonomous operation, a GNSS-based navigation is used for precise positioning, utilizing two set of RTK-GNSS modules (Model: C099-F9P, u-Blox Inc., Switzerland) along with two antennas (Model: VEXXIS Antennas 502, NovAtel Inc., France), which are installed on the front and rear brackets on the top of the robot. An embedded development board (Model: Jetson Xavier NX, NVIDIA Inc., USA) serves as the controller for integrated tasks within the robot's operating system, processing signals from various sensors and enabling autonomous operation per programmed instructions. The development board and peripheral electronic components are housed within a control box. A camera (Model: BRIO 4K Ultra HD, Logitech Inc., Switzerland) is affixed beneath the central bracket of the robot, tasked with capturing images of the farmland, crops, weeds, or water tape lines, among others. Three spray modules are enclosed in a waterproof box attached to the robot's side bracket. Each module's outlet is connected by hoses to nozzles at the rear of the robot's central bracket. The nozzles, all directed towards the ground, align with the left, center, and right sections of the camera's field of view, respectively. Connectivity for data transfer utilizes USB, RS485, and RS-232 protocols, serving as the interface between the robot's operating system and various components such as the camera, two RTK-GNSS receivers, drivers, and other peripherals. This structured cabling and interface management ensure seamless communication and operation coordination, critical for the robot's precision agriculture tasks. The strategic placement of these components, along with the careful integration of data interfaces, enhances the robot’s functional efficiency and responsiveness in a farming environment.

 doi:10.20944/preprints202310.1427.v1

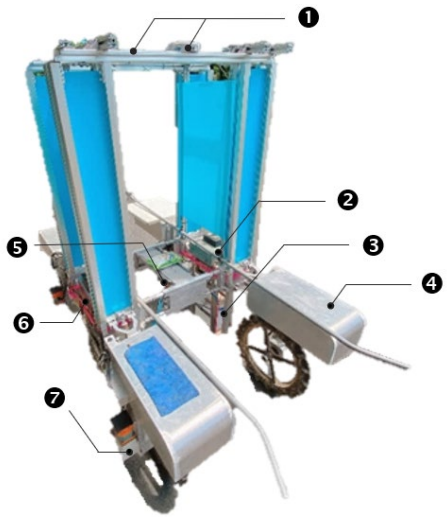


Figure 1. The appearance and functional module configuration of the robot. ❶ Two RTK-GNSS positioning modules; ❷ controller box; ❸ spraying module; ❹ electric cylinders; ❺ webcam; ❻ battery; ❼ DC motors.

Connected to four sets of electric motors (Model: 9B200P-DM, Taiying Technology) and servo motors (Model: CS60-150C8AE, CSIM Inc., Taiwan) are four sets of computer-controlled driving devices (Model: CSBL1400, CSIM Inc., Taiwan). Each electric motor's drive shaft is coupled to a reduction gear set (Model: 9VD360H, TROY Enterprise Co., Ltd., Taiwan) with a 360:1 reduction ratio. The output shaft of the servo motor is connected to a linear electric cylinder (Model: LOE-40-100-C-L5, Lind Inc., China). As the output shaft of the servo motor rotates, it simultaneously drives the linear electric cylinder's rod to move linearly back and forth, as well as the connected parts at the rod's end to rotate. This rotation adjusts the steering angle δ of the wheel (see Figure 2). The relationship between δ and the length of extension or contraction l of the electric cylinder is defined as (1).

$$l = \sqrt{(r - r\cos\delta)^2 + (d + r\sin\delta)^2} \tag{1}$$

Among these parameters, r denotes the length from the origin of the connecting member to the push rod's endpoint, while d represents the horizontal distance between the electric cylinder's front end and the connecting member.

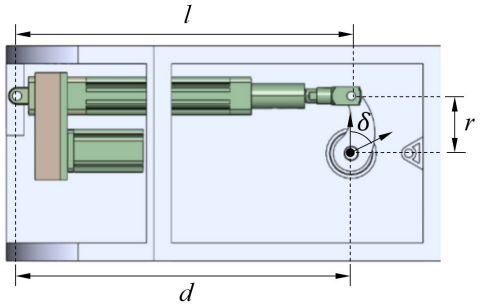


Figure 2. Steering mechanism of robot.

Given the constant and relatively slow travel speed of robot, along with its rigid tires, its motion state at any given moment can be described using a bicycle model, as shown in Figure 3a.

The global X-Y coordinate plane is a fixed horizontal plane upon which the vehicle moves and is used to describe its motion. It is assumed that O, O_f, and O_r represent the vehicle's center of gravity, the center of the rear wheel, and the center of the front wheel, respectively. The distance between the center of the front wheel and the vehicle's center of gravity is L_f, while L_r is the distance between the center of the rear wheel and the vehicle's center of gravity. The slip angle is represented

 doi:10.20944/preprints202310.1427.v1

by α, the heading angle by θ, and the speed at the center of gravity by v, with its component velocities being $\dot{x}$ and $\dot{y}$.

This motion model is based on front-wheel steering, assuming the direction of the rear wheels is parallel to the vehicle body. The state variables of kinematic motion encompass the robot's configuration, represented by ($\dot{x}$, $\dot{y}$, $\dot{\theta}$).

$$\dot{x} = v\cos(\theta+\alpha) \tag{2}$$

$$\dot{y} = v\sin(\theta+\alpha) \tag{3}$$

$$\dot{\theta} = \frac{v\cos\alpha\tan\delta}{L_f+L_r} \tag{4}$$

where $v=\frac{v_f\cos\delta+v_r}{2\cos\alpha}$, $\alpha=\tan^{-1}(\frac{L_r\tan\delta}{L_f+L_r})$. v_f and v_r illustrate the velocity of front and rear wheel.

In this study, during straight-line movement, the robot's front wheel steering angle ranged from −17° to 17°, with the rear wheel steering angle fixed at 0. Moreover, during turning maneuvers, all wheels assumed a steering angle of $|\delta|$= 47°, putting the robot in a state of on-the-spot rotation (Figure 3b).

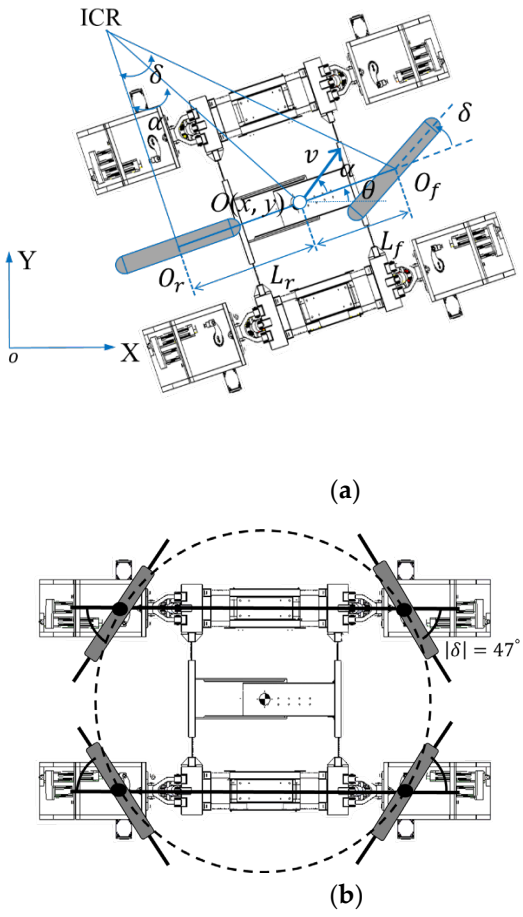


Figure 3. The motion mode of robot. (**a**) linear motion model; (**b**) rotation mode.

2.2. Path Planning

The robot's path of movement between the ridges and furrows in the field is planned as indicated by the dashed arrows in Figure 4. The left and right wheels straddle the sides of a ridge, with each ridge's head and tail end serving as turning points (block dots). Upon reaching these points, the robot stops and performs a 90-degree on-the-spot rotation. Then, the robot moves to the turning point of

 doi:10.20944/preprints202310.1427.v1

the next ridge and stops. Subsequently, the robot rotates 90 degrees again, this time with its front facing the ridge. Finally, the robot proceeds to move straight along the furrow. The GNSS receiver, enhanced with RTK capabilities, produces navigation data in a format established by the National Marine Electronics Association (NMEA), offering highly accurate longitudinal and latitudinal details [32]. The RTK-GNSS receivers supply precise position data for designated turning points within agricultural terrains. For tracing the robot's path across a two-dimensional plane, coordinates based on the WGS-84 geodetic system, as utilized by GNSS, undergo conversion to a two-dimensional projection within the Taiwan Geodetic Datum 1997 (TWD97) [33].

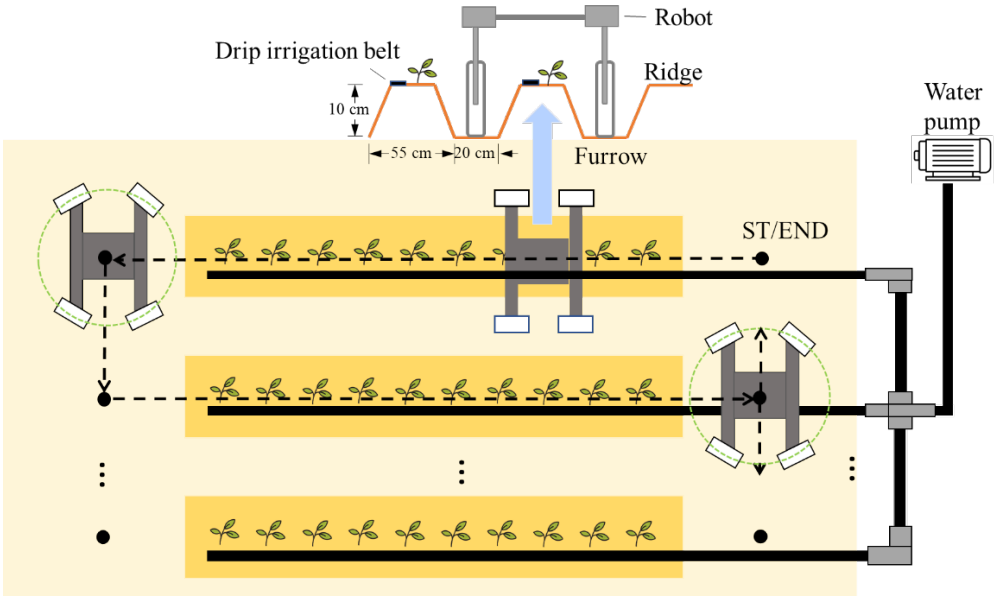
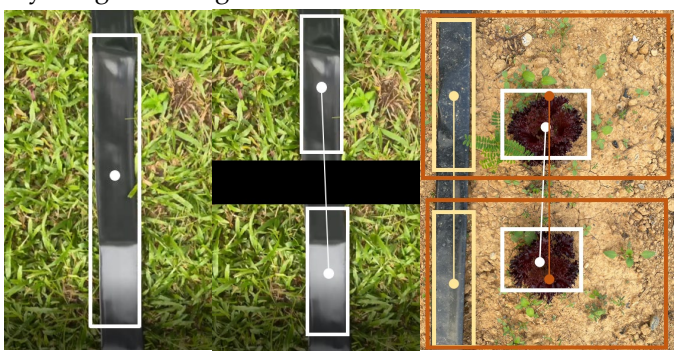


Figure 4. Schematic of the path planning of the robot moving in the field. Black dots represent turning points or start/end points (ST/END), and dotted arrows represent movement paths.

2.3. Row Detection

During fallow periods, farmland may exhibit only furrows or what are referred to as black water strips (interpreted here as drip irrigation lines) interspersed between ridges. In contrast, the planting season may present a mix of crops and furrows without the consistent presence of irrigation lines, or, in some instances, exclusively crops, contingent upon individual agricultural practices. To accommodate these variations and facilitate autonomous guidance line generation, a camera was mounted on the robot to record terrestrial features. Object center points, such as those of irrigation lines (illustrated as white dots in Figure 5a), were discerned through deep learning techniques. However, since a single point does not define a line segment, the images were processed with a black mask, bifurcating them into upper and lower sections and thereby segmenting individual objects. This method enabled the linkage of center points from the bifurcated objects, creating discernible line segments (depicted in Figure 5b). Potential objects identified within the images encompass irrigation lines, crops, and furrows (Figure 5c). Notably, the detection process for crop objects circumvents the necessity for preliminary image masking.



 doi:10.20944/preprints202310.1427.v1

(**a**) (**b**) (**c**)

Figure 5. Schematic representation of single-point detection and line segment formation methods. (**a**) single point; (**b**) single line segment; (**c**) multiple line segments.

It's crucial to maintain the stable capture of object center points during the robot's movement. Challenges such as terrain-induced oscillation or variable external lighting can disrupt the accuracy of object bounding box determination in deep learning applications, causing deviations in center point detection. To address this, multiple prospective object center points are extracted from images for a statistical regression analysis, aiming to identify the most accurate function that fits the distribution of these points while minimizing the sum of squared errors [34]. Given k data pairs, represented as $\{(x_i, y_i),\ i= 1,\ldots,k\}$, the core relation between x_i and y_i is discerned, factoring in an error term to account for any uncertainties or deviations.

$$y_i = ax_i + b + \varepsilon_i \quad (5)$$

Assuming that $\hat{a}$ and $\hat{b}$ stand for the approximations of parameters a and b these are utilized in a subsequent problem, aiming to achieve the most suitable fit for the data points.

$$\min_{a,b} \sum_{i=1}^{k} \hat{\varepsilon}_i^2 \quad (6)$$

If b is assumed to be 0, the ordinary least squares method is used for estimating $\hat{a}$ to minimize (6).

$$\hat{a} = \frac{\sum_{i=1}^{k} x_i y_i}{\sum_{i=1}^{k} x_i^2} \quad (7)$$

While the robot navigates, its object detection system, based on the YOLOv4 deep learning model, identifies various features such as water tape lines, crop rows, or furrows. These serve as guiding markers. The system also recognizes weeds and signs of nutrient deficiency in crops, essential for precision agriculture tasks. YOLOv4, built on Darknet and introduced by Alexey Bochkovskiy et al. in 2020, represents a significant evolution in the YOLO series [35]. It stands out due to its capacity for real-time object detection and localization through a single processing stage, leveraging advancements like deeper CNNs, FPN, and SPP. This feature makes it adept at recognizing small objects, overcoming obstructions, and facilitating multi-scale detection. The model's efficiency stems from its architecture, divided into three core parts: the backbone, the neck, and the detection head. As depicted in Figure 6, the backbone (CSPDarknet53) initiates the process, extracting features from the input image and channeling them through three significant residual blocks (N_1-N_3) composed of specialized layers. Notably, the Cross-Stage Partial Network (CSPNet) within these blocks enhances gradient information flow, reducing computational redundancy and improving real-time applicability—a vital aspect for mobile robotics in precision agriculture [36]. In YOLOv4, the Leaky-ReLU activation function is applied to various convolutional layers and other layers with non-linear activations within the architecture, allowing the model to enhance its non-linear modeling capabilities while maintaining computational efficiency. Through this activation function, the model can better represent input data without compromising computational efficiency.

Introducing the post-convolutional image features into the Neck layer, SSPnet is used to expand the receptive field of the detection model with different-scale max-pooling layers. PANet repetitively extracts features using both top-down and bottom-up approaches. Here, S_i (i = 1, 2, 3, 4, 5) represents the fusion process of different-sized feature maps in the Neck layer. Si performs the processes of convolution, batch normalization, and Leaky-ReLU feature extraction and classification. In S_1, multiple max-pooling layers are used, S_2 and S_3 employ top-down feature fusion methods, while S_4 and S_5 execute feature fusion processes using a bottom-up approach. Finally, the results are output in the YoLo head layer, including YoLo head 1 (76×76), YoLo head 2 (38×38), and YoLo head 3 (19×19). These heads are used for fusion and interaction with feature maps of different scales to detect objects of varying sizes.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 23 October 2023 doi:10.20944/preprints202310.1427.v1

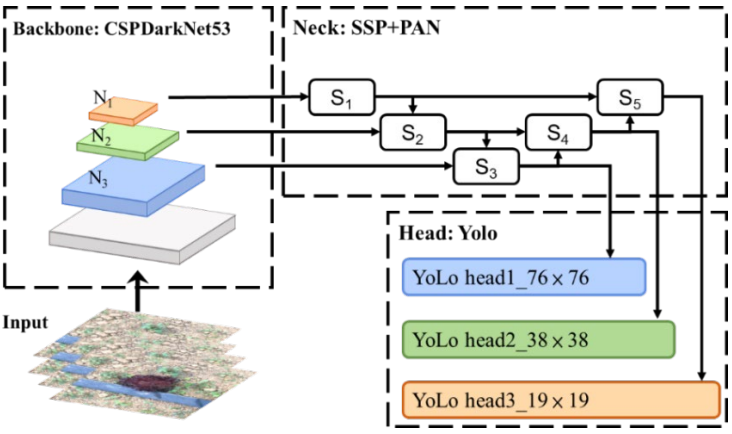


Figure 6. The framework of YoLov4 model.

Within the head layers, the main operation is the detection of features at different scales. Bounding Box Regression is utilized to adjust the predicted boxes' positions to match the actual target boxes. Finally, the Generalized Intersection over Union (GIOU) method is employed to assess the overlap between predicted boxes and real target boxes [37]. Predicted frame ($\boldsymbol{\varphi}'$) and real target frame ($\boldsymbol{\varphi}$) are illustrated in Figure 7, with $\bar{\boldsymbol{\varphi}}$ denoting the minimum outer bounding frame encompassing both.

$$L_{GIOU}=\frac{1-IOU(\varphi,\varphi')+|\bar{\varphi}-\varphi\cup\varphi'|}{|\bar{\varphi}|} \quad (8)$$

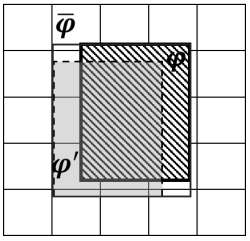


Figure 7. Illustration of the prediction frame and the real target frame. $\boldsymbol{\varphi}'$ is the prediction frame, $\boldsymbol{\varphi}$ is the real target frame (ground truth), and $\bar{\boldsymbol{\varphi}}$ is the minimum peripheral frame between $\boldsymbol{\varphi}'$ and $\boldsymbol{\varphi}$.

Finally, the total loss (P_{Loss}), comprising object classification loss (LOC), confidence loss (LOF), and object prediction position loss (LOCI), is defined by (25).

$$P_{Loss}=\epsilon_1 LOC+\epsilon_2 LOF+\epsilon_3 LOCI \quad (9)$$

where ϵ_1, ϵ_2, and ϵ_3 represent the balancing coefficients; LOC and LOF are defined using the cross-entropy method [38]; LOCI is predicted based on the Complete Intersection Over Union (CIOU) algorithm, which calculates the positional loss between the bounding box and the real target box. Finally, defining the results of predicted categories as either identical or different from the real results, assuming TP represents samples where the model predicts positive and is also actually positive, FP for samples where the model predicts positive but is actually negative, TN for samples where the model predicts negative and is also actually negative, and FN for samples where the model predicts negative but is actually positive. According to the above definitions, Precision, Recall, and F1-Score are defined to assess the detection performance of the identification model.

Precision: It represents the proportion of actual positive samples among all the samples predicted as positive by the model.

$$Precision=\frac{TP}{TP+FP} \quad (10)$$

Recall: It depicts the proportion of successfully predicted positive samples among all the actual positive samples.

$$Recall=\frac{TP}{TP+FN} \quad (11)$$

 doi:10.20944/preprints202310.1427.v1

F1-Score: It is the harmonic mean of precision and recall, used to provide a comprehensive evaluation of the performance of model.

$$F1\text{-}Score = 2\times\frac{Precision\times Recall}{Precision+Recall} \quad (12)$$

2.4. Guidance and Heading Control

2.4.1. Velocity Measurement

The PID control method is commonly applied in the velocity control systems of various robots. Facing the often unstable dynamic characteristics of motion systems, this control method maintains a robot's stable operation process by adjusting three parameters: Proportional Gain (K_p), Integral Gain (K_i), and Derivative Gain (K_d).

K_p establishes a proportional relationship between the controller output and the input system error signal. When the proportional gain increases, the output under the same error also increases, and vice versa. However, when only using K_p, the system may exhibit a steady-state error. To eliminate this offset, K_i is introduced, which works by integrating the error over time to accelerate the system's ability to reach the target state. As time progresses, K_i also increases, meaning that even with smaller errors, it will grow due to the passage of time until the steady-state error reaches zero.

On the other hand, K_d adjusts based on the rate of change of the error signal relative to time. In this study, four sets of PID controllers are used to control the rotation speed of the four motors of a four-wheeled robot, each dedicated to one motor. This approach ensures precise adjustments, responding to the specific needs and conditions of each wheel, enhancing the robot's overall stability and performance in varying operational contexts.

In this research, four sets of PID controllers are deployed to regulate the rotational speed of the four individual motors within a four-wheeled robot.

$$u(t)=K_p e(t)+K_i\int_0^t e(t)\,dt+K_d\frac{d}{dt}e(t) \quad (13)$$

where $e(t)$ depicts the error between the desired velocity and estimated velocity at time t. Certainly, a trial-and-error method is initially employed to determine the K_p value that brings the system to a marginally stable state. This process yields a proportional gain parameter denoted as K_{pc}. K_{pc}, in combination with a proportional coefficient, is then used to set the K_i value. Additionally, the system's time period (T_c) is measured and employed to obtain the K_d parameter. PID parameters are defined in Table 1.

Table 1. Definition of PID parameters.

Type	K_p	K_i	K_d
PID	$0.3\cdot K_{pc}$	$0.2\cdot K_{pc}$	$0.06\cdot K_{pc}\cdot T_c$

2.4.2. Heading Angle Measurement

The fuzzy logic control method is utilized to adjust the steering angle of the robot's front two wheels, as depicted in Figure 8. This system comprises fuzzification, an inference engine, defuzzification, and a knowledge base [39].

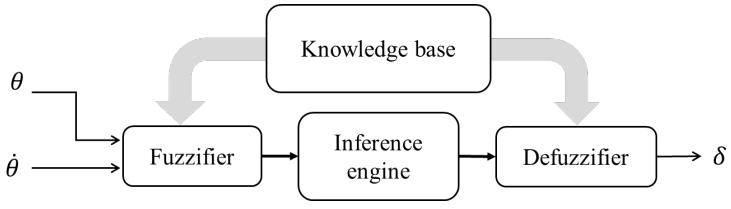
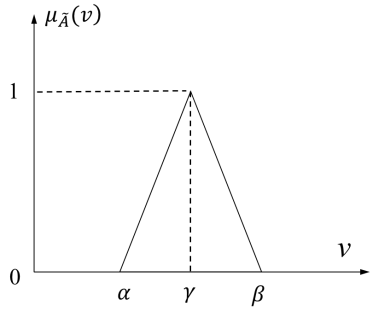


Figure 8. Block diagram of fuzzy logic controller.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 23 October 2023 doi:10.20944/preprints202310.1427.v1

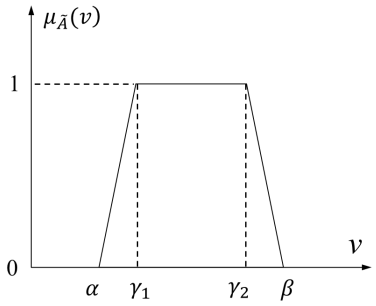
- Fuzzification process;

The role of fuzzification is to map input crisp values "v" to a fuzzy set, defining a linguistic term $\widetilde{A}$ to represent this set, viewed as a membership function. The most common functions are triangular and trapezoidal membership functions. As shown in Figure 9a, the triangular function and its mathematical representation consist of three parameters. The lower boundaries on the left and right are represented by α and β, respectively, while γ denotes the peak of the triangle. When the input crisp value " v " falls between α and γ, its degree of membership $\mu_{\widetilde{A}}(v)$ is non-zero; it is 1 when " v " equals γ, and zero when " v " is less than α or greater than β. It can be inferred that the closer " v " is to γ, the higher its degree of membership. Similarly, Figure 9b depicts the trapezoidal membership function and its mathematical representation, consisting of four parameters: α, γ_1, γ_2, and β.



$$\mu_{\widetilde{A}}(v)=\begin{cases}0, & \text{if } v<\alpha\\ \dfrac{v-\alpha}{\gamma-\alpha}, & \text{if } v\in[\alpha,\gamma]\\ 1, & \text{if } v\in[\gamma]\\ \dfrac{\beta-v}{\beta-\gamma}, & \text{if } v\in[\gamma,\beta]\\ 0, & \text{if } v>\beta\end{cases}$$

(**a**)



$$\mu_{\widetilde{A}}(v)=\begin{cases}0, & \text{if } v<\alpha\\ \dfrac{v-\alpha}{\gamma_1-\alpha}, & \text{if } v\in[\alpha,\gamma_1]\\ 1, & \text{if } v\in[\gamma_1,\gamma_2]\\ \dfrac{\beta-v}{\beta-\gamma_2}, & \text{if } v\in[\gamma_2,\beta]\\ 0, & \text{if } v>\beta\end{cases}$$

(**b**)

Figure 9. Membership function and mathematical expressions. (**a**) Triangular membership function; (**b**) ladder membership function.

The heading angle (θ) and the rate of change of the heading angle ($\dot{\theta}$) are the two input variables for the fuzzy controller. The heading angle corresponds to three fuzzy linguistic terms: LO (Left Offset), M (Middle), and RO (Right Offset). Additionally, three fuzzy linguistic terms represent the rate of change of the heading angle: Negative (N), Zero (Z), and Positive (P). The output variable is the steering angle correction amount (δ), defined by the fuzzy linguistic terms Left (L), Mid (M), and Right (R). Table 2 compiles the input and output variable values corresponding to the fuzzy logic statements and the parameter values of their membership functions for the heading angle controller.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 23 October 2023 doi:10.20944/preprints202310.1427.v1

Table 2. Parameters of membership function of input and output variable in fuzzy logic controller.

Input Variable						Output Variable		
Heading Angle (θ)			Rate of Change of the Heading Angle ($\dot{\theta}$)			Steering Angle (δ)		
Crisp Interval		Linguistic Labels	Crisp Interval		Linguistic Labels	Crisp Interval		Linguistic Labels
Triangular $[\alpha, \gamma, \beta]$	Ladder $[\alpha, \gamma_1, \gamma_2, \beta]$		Triangular $[\alpha, \gamma, \beta]$	Ladder $[\alpha, \gamma_1, \gamma_2, \beta]$		Triangular $[\alpha, \gamma, \beta]$	Ladder $[\alpha, \gamma_1, \gamma_2, \beta]$	
◎	[-100, -100, -20, 0]	LO	◎	[-100, -100, -25, 0]	N	◎	[-17, -17, -7, 0]	L
[-10, 0, 10]	◎	M	[-20, 0, 20]	◎	Z	[-5, 0, 5]	◎	M
◎	[0, 20, 100, 100]	RO	◎	[0, 25, 100, 100]	P	◎	[0, 7, 17, 17]	R

- Knowledge base;

The knowledge base consists of rules primarily using [IF—THEN] statements to describe relationships between input and output variables. When multiple fuzzy input variables are involved, [IF —AND—THEN] statements are used. For the heading angle controller, nine rules are defined based on two input variables and one output variable:

Rule 1: IF (θ is LO) AND ($\dot{\theta}$ is N) THEN (δ is L)

Rule 2: IF (θ is LO) AND ($\dot{\theta}$ is Z) THEN (δ is M)

Rule 3: IF (θ is LO) AND ($\dot{\theta}$ is P) THEN (δ is R)

Rule 4: IF (θ is M) AND ($\dot{\theta}$ is N) THEN (δ is L)

Rule 5: IF (θ is M) AND ($\dot{\theta}$ is Z) THEN (δ is M)

Rule 6: IF (θ is M) AND ($\dot{\theta}$ is P) THEN (δ is R)

Rule 7: IF (θ is RO) AND ($\dot{\theta}$ is N) THEN (δ is L)

Rule 8: IF (θ is RO) AND ($\dot{\theta}$ is Z) THEN (δ is M)

Rule 9: IF (θ is RO) AND ($\dot{\theta}$ is P) THEN (δ is R)

For example, in Rule 1, when θ belongs to the left offset and $\dot{\theta}$ is a negative rate of change, the wheels turn left.

- Fuzzy inference and decision;

The Mamdani model, also known as the "max-min composition method," is used for inference. Its principle is to select the minimum membership degree corresponding to the fuzzy set of the antecedent condition in the fired rules and assign it to the corresponding fuzzy set in the consequent condition, based on the input values. Finally, the output fuzzy sets of all the fired rules are combined using a union operation, which is the maximum value method. This section may be divided by subheadings. It should provide a concise and precise description of the experimental results, their interpretation, as well as the experimental conclusions that can be drawn. Figure 10 illustrates the concept of Mamdani inference using two rules. The horizontal axis represents the physical domain, and the vertical axis represents a real interval ranging from 1 to 0, which corresponds to the membership degrees in fuzzy theory. "v " and "w " represent the crisp values of the input variables. $\mu_{\tilde{A}}$, $\mu_{\tilde{B}}$, $\mu_{\tilde{C}}$ represent the fuzzy sets corresponding to the input variables in rule p or rule q. $\mu_{\tilde{z}}$ represents the union of the output fuzzy sets of the fired rules.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 23 October 2023 doi:10.20944/preprints202310.1427.v1

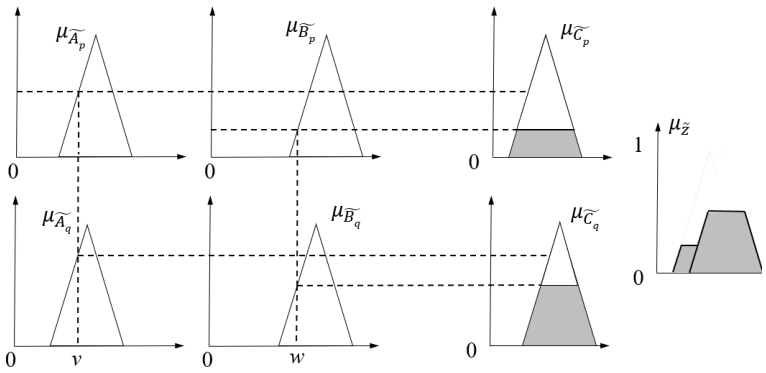


Figure 10. Mamdani model with max-min composition.

For example, let's consider the case when θ = 7.34 and $\dot{\theta}$ = 29. As shown in Figure 11, in the nine rules, the yellow areas represent the fuzzy sets corresponding to the two input values with membership degrees greater than 0. On the right half, the inference result using the min operation includes the regions with membership degree 0 and lower membership degrees in yellow. The max operation rule involves selecting the maximum degree of membership among the consequent conditions of various rules and outputting it as the result.

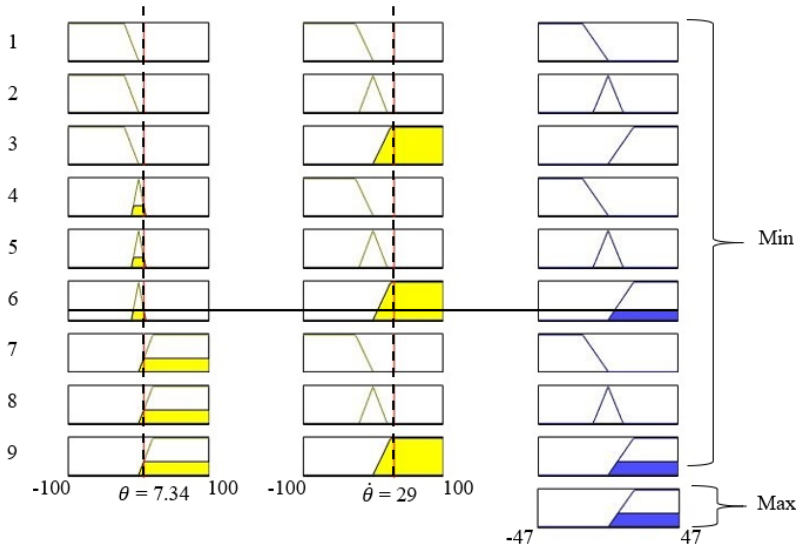


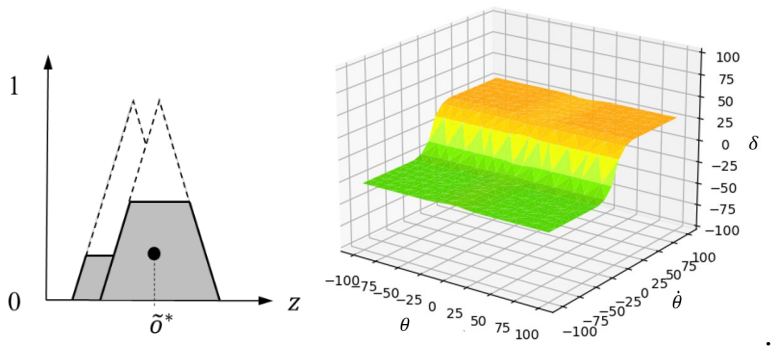
Figure 11. The inference engine using Mamdani model.

- Defuzzification.

After the comprehensive inference, the result consists of a series of fuzzy sets with different degrees of membership. Therefore, it is necessary to convert these values into crisp numbers using a defuzzification process. Common methods for defuzzification include the centroid method, the max membership method, the height method, and the area method. The centroid method is the most commonly used and is preferred in this study as the defuzzification method, as shown in (14), where $\tilde{o}^*$ represents the defuzzified crisp value. This method is used to calculate the centroid of the shaded area in the inference result shown in Figure 12.

$$\tilde{o}^* = \frac{\int_z z\mu_{\tilde{z}} dz}{\int_z \mu_{\tilde{z}} dz} \tag{14}$$

 doi:10.20944/preprints202310.1427.v1



(**a**) (**b**)

Figure 12. (**a**) Defuzzification (center of gravity method); (**b**) 3-D surface view of fuzzy rules for FLC.

3. Results and Discussion

3.1. Environmental Conditions and Experimental Parameters

The experimental site is located in a demonstration field situated in a rural area with coordinates at longitude 120.60659° and latitude 22.64606°. The field spans approximately 13 meters in length, with individual row widths measuring 80 centimeters. The robot's physical dimensions are 2.4 meters in length and 1.06 meters in width, with a ground clearance height of 1.1 meters. Due to spatial constraints, the robot operates within the space between two rows. The crop of interest is Rosalind Red Leaf Lettuce (HV-067, Taiwan Agro Seed Co., Ltd.). Two deep learning models were developed and employed for row detection and the detection of weeds and symptoms of nutrient deficiency in crops. Image samples were captured by the operator at random times each day in the experimental field. These images encompassed weeds, both healthy and abnormal crops, irrigation lines, and ridges. The collected samples were divided into a 7:2:1 ratio for the training, validation, and testing sets. Images in the training set were manually annotated using the open-source image annotation tool, LabelImg, to mark objects within the images. The abnormal growth of Rosalinda lettuce crops is characterized by symptoms such as yellowing and wrinkling of leaf edges, as depicted in Figure 13. These symptoms indicate a deficiency in essential elements such as nitrogen and potassium.

The hyperparameters for both YoLov4 models are configured as follows: batch size set to 64, subdivisions set to 32, image size of 416×416 (width×height), decay rate of 0.0005, momentum of 0.949, learning rate set to 0.001, and a maximum number of batches set to 10,000.

(**a**) (**b**)

Figure 13. Leaf appearance due to nutritional deficiencies. (**a**) Sample 1; (**b**) sample 2.

3.2. Initial Test

The training iterations for the two detection models of different categories were stopped after reaching 10,000 and 6,000 iterations, with P_{Loss} values of 1.0719 and 14.8856, respectively. After training, the models were tested, and the precision, recall, and F1-Score results are presented in Table

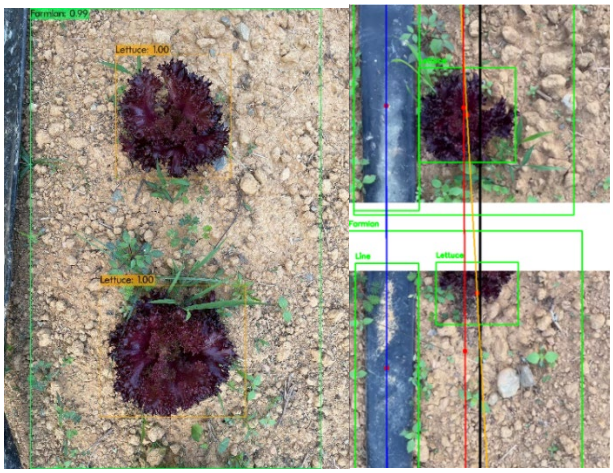
 doi:10.20944/preprints202310.1427.v1

3. It can be observed that the average precision for all three object categories exceeds 98%, with a recall rate of 99%, and an F1-Score of at least 96%. Regarding nutrient deficiency and weed detection, precision for both features is over 90%, with a recall rate of over 81%, and an F1-Score of at least 85%.

Table 3. The performance of the YOLOv4 model in detecting objects of different categories.

Type	Average precision (%)	Recall (%)	F1-Score (%)
Black drip irrigation belt	99.2	99.0	96.0
Crop	99.1	99.0	96.3
Ridge	98.5	99.0	96.1
Crop with nutritional deficiencies	90.0	81.3	85.4
Weed	91.2	84.2	88.5

As the robot moves between furrows, the deployed YoLov4 model in the robotic operation system continuously and instantly performs object detection to capture expected objects in the images, as shown in Figure 14a. Orange boxes represent crops, and green boxes represent furrows. The selected objects are then used to generate multiple guidance lines, as described in Section 2.3. For example, Figure 14b illustrates crop lines (orange lines), ridge lines (red lines), and drip irrigation lines (blue lines). The normal black line represents the vertical line in the center of image.



(**a**) (**b**)

Figure 14. The results of object detection on the ridge and reference line detection. (**a**) Displays the detection results for crops (orange boxes) and furrows (green boxes); (**b**) shows the central vertical line of the image (black), irrigation lines (blue), ridge lines (red), and crop lines (orange). The green boxes indicate object bounding boxes for crops, drip irrigation belt, or ridge.

3.3. Experimental Results

Presented in this section are the performance of the robot executing autonomous navigation techniques and the results of field operations tests, including weed detection, crop identification, recognition of nutrient-deficient crops, and spraying operations.

3.3.1. Autonomous Guidance

The experiments were conducted in the summer of 2023. The experimental time periods were divided into three segments: from 9 AM to 11 AM, from 12 PM to 2 PM, and from 3 PM to 5 PM. Within each time segment, the robot executed ten rounds of circumnavigation along a predefined route. While in motion, the robot continuously maintained its velocity and corrected its heading angle through the utilization of four PID controllers and a fuzzy logic-based steering control algorithm embedded within the guidance system. Objects falling into three distinct categories within the ridges were identified and subsequently used to generate guidance lines. These guidance lines were employed to compute the average heading angle of the robot. Two RTK-GNSS receivers were

 doi:10.20944/preprints202310.1427.v1

employed to record the robot position, separated by 1.5 meters. The latitude and longitude of location obtained from the two receivers are each converted into two TWD97-based positions. Observing the robot movement trajectory in Figure 15, it can be found that the robot can move from the starting point (ST) to the end point (END) smoothly in different segments. Since different weather conditions will affect the positioning performance of RTK-GNSS receiver, especially if there are large trees around the experimental site, the movement trajectories recorded at some times have high positioning errors. Using these positioning data to assist the robot in correcting its heading angle is less reliable.

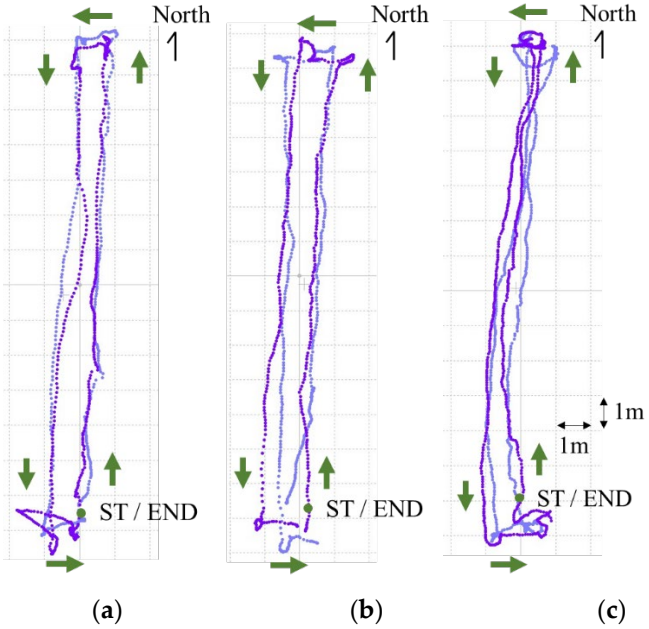


Figure 15. The robot movement trajectories (dark purple dot and light blue dot) received by two sets of receivers at different time segments. (**a**) 9 AM to 11 AM (weather: sunny and cloudy); (**b**) 12 PM to 2 PM (weather: sunny); (**c**) 3 PM to 5 PM (weather: cloudy). The size of the square is 1x1 m (length x width). The green dot represents the starting point and is also the end point.

The change in velocity of wheel and heading angles of robot are shown in Figures 16 and 17 respectively. The control parameters for the four sets of motors, K_p, K_i, and K_d, were all set to 0.5, 0.1, and 0.6, respectively. In Figure 16, it can be observed that the robot successfully completed two ridge crossings (orange area). Specifically, when the robot moved in a straight line (blue area), the desired motor speeds for all four wheels were maintained at approximately 1.95±0.03 rpm. Additionally, the rotational velocity of four wheels remained at around 2.80±0.03 rpm during in-place rotations.

While using the PID method for velocity control, momentary speed overshooting occurs (gray circles in Figure 16a) during motor speed changes. Figure 17 displays the variations in heading angles resulting from different category guidance lines. It highlights that guidance lines formed by the center points of crop objects exhibit the largest heading angle variations compared to other categories, while guidance lines formed by irrigation lines show the smallest heading angle variations.

 doi:10.20944/preprints202310.1427.v1

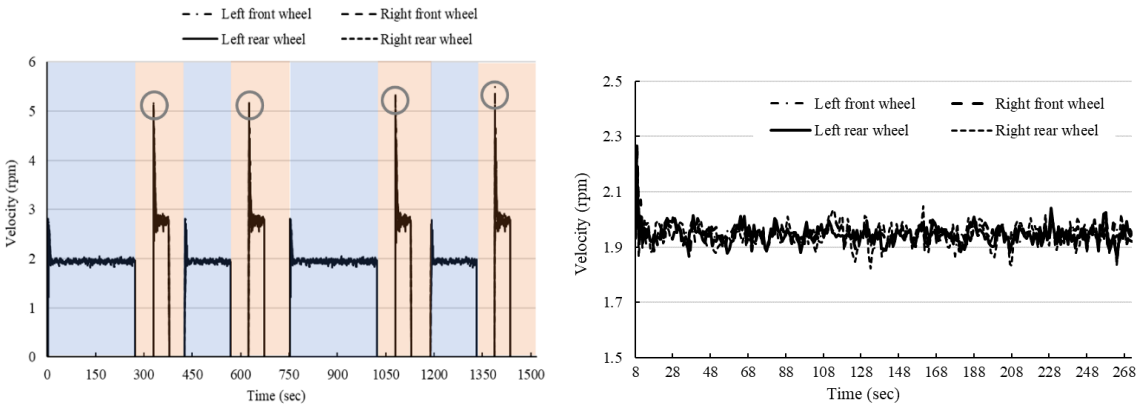


Figure 16. Velocity control with PID (morning segments). (**a**) Changes in the rotational velocity of the four wheels (moving in a straight line (blue area)), changing furrow (orange area); (**b**) changes in the velocity of the four wheels during straight-line movement (enlargement of Figure 16(**a**) (8 seconds to 268 seconds)).

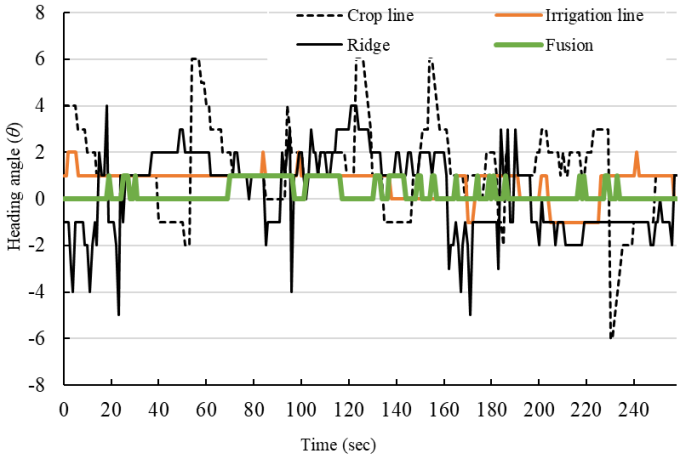


Figure 17. Changes in heading angle on different types of guidance lines (morning segment). The angular variation in the mean heading angle is approximately 1 degree (green line).

Similar results are also presented under different weather conditions, as shown in Table 5. From the table, it can be observed that, among the three types of guidance lines, the smallest variation in heading angle is obtained when using drip irrigation lines. It is followed by ridge lines, and the largest variation is exhibited by crop lines. Comparing the slope changes presented in Figure 18, it can be observed that the slopes for irrigation lines (blue) and ridge lines (red) are similar, but crop lines (orange) exhibit significantly larger differences from the other lines.

Table 5. The average variation range of heading angle under different weather conditions, in degrees (°).

Type of Guidance Line	Sunny	Sunny and Cloudy	Cloudy
Irrigation line	±3	±2	±2
Crop line	±6	±6	±6
Ridge line	±6	±3	±4
Fusion	±2	±2	±1

 doi:10.20944/preprints202310.1427.v1

Figure 18. The crops at the top, middle and bottom of the image are not parallel to the center vertical line of the ridge. Irrigation line (blue color); crop line (orange color); ridge line (red color).

On the other hand, under sunny and cloudy conditions with an image processing rate of 7 fps, different movement speeds of the robot impact the recognition rate of guidance lines. As shown in Figure 19, it can be observed that the line detection system is able to recognize irrigation lines and achieve a recognition rate of 98% when the robot's movement speed is 12.5 cm/s. However, as the robot's movement speed increases, the recognition rate of irrigation lines gradually decreases. Specifically, at a movement speed of 24 cm/s, the recognition rate has dropped to below 80%, and when the robot's movement speed reaches 35 cm/s, the recognition rate has decreased to below 55%.

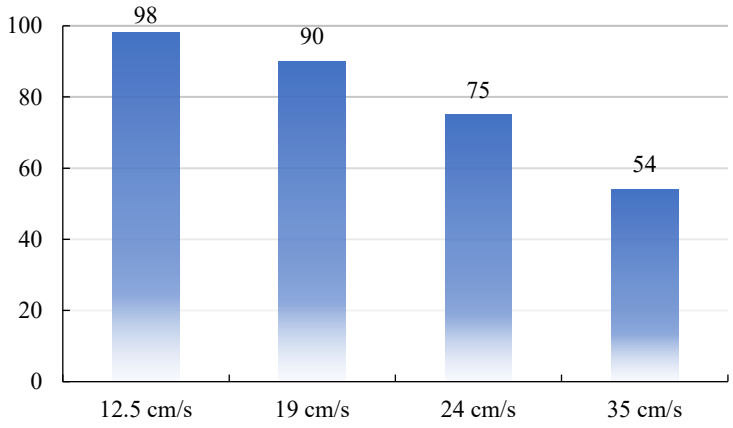


Figure 19. The performance comparison of the irrigation lines recognition at different movement speeds for robot.

3.3.2. Identification of Weed and Crop Nutrient Deficiency Symptoms

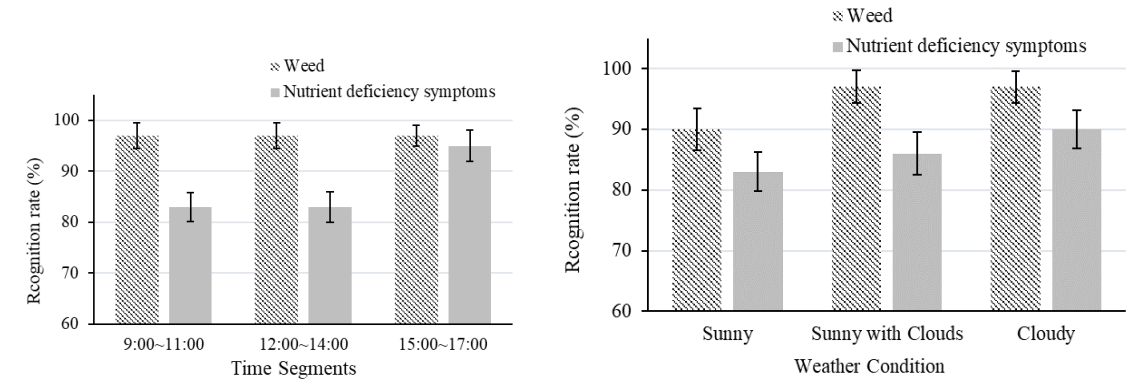
The recognition performance of the trained model for weeds and crop nutrient deficiency symptoms was evaluated under different time periods and varying weather conditions. Weeds were randomly distributed across the ridge in the field, and crops exhibiting growth abnormalities were preprocessed to have distinct appearances from normally growing crops. Figures 20a,b depict the recognition results of weeds and crop nutrient deficiency symptoms, with confidence scores indicated within green or purple boxes. As shown in Figure 21a, when the robot moved at a speed of

 doi:10.20944/preprints202310.1427.v1

12.5 cm/s, the average recognition rate for weeds reached 99% across different time segments, while the average recognition rate for crop nutrient deficiency symptoms was 88%. Under different weather conditions, as illustrated in Figure 21b, the average recognition rate for weeds was 97%, and for crop nutrient deficiency symptoms, it was 86%.

(**a**) (**b**)

Figure 20. The feature recognition results for weeds and crop nutrient deficiency symptoms on the ridge. (**a**) Weeds (in green boxes); (**b**) nutrient deficiency symptoms (in purple boxes).



(**a**) (**b**)

Figure 21. Recognition rates of weeds and crop nutrient deficiencies in different time segments (**a**) and under different weather conditions (**b**).

3.3.3. Spray Test

Spraying operations were conducted in the morning (9:00◎11:00), noon (12:00◎14:00), and afternoon (15:00◎17:00), with approximately a 5:1 ratio of healthy crops to crops with growth abnormalities. The robot had three sets of nozzles: the central nozzle primarily sprayed crops with growth abnormalities, while the left and right nozzles focused on weed control. Correct spraying was determined by visually inspecting whether colored water had contaminated the crops, as shown in Figure 22. Failure to spray the target or the absence of color contamination was considered an error. Each spraying experiment was repeated ten times during each time segment, with the crops being thoroughly cleaned of any color contamination from the previous trial. The robot moves at a speed of 12.5 cm/sec. The average spraying accuracy for the three time periods is shown in Figure 23. Weed spraying achieved the highest accuracy at 92%, while spraying crops with nutrient deficiencies reached a peak accuracy of 80%. However, during the noon period (12:00 to 14:00), the accuracy dropped to below 70%.

 doi:10.20944/preprints202310.1427.v1

Figure 22. The result of spraying performed by the spray module (brown color).

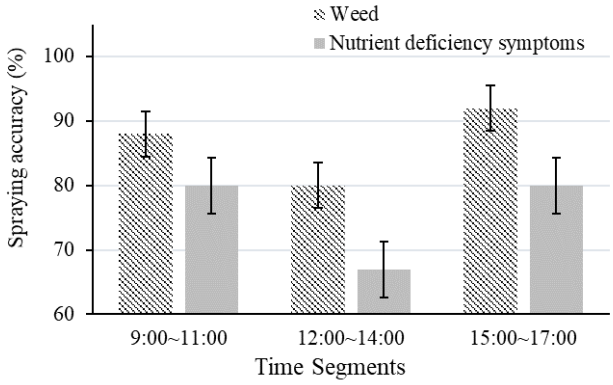


Figure 23. Comparison of spraying performance at different time segments.

3.3.4. Discussion

Based on the comprehensive analysis of our experimental results, we extract the following key points:

Firstly, the four-wheel independent steering design allows the robot to have a gyration radius of 0 when performing turning operations, which can increase the effective farming area and increase crop yield per unit area. The two sets of RTK-GNSS receivers used to obtain the robot's position and movement trajectory can not only observe the robot's movement posture, but also help the robot perform ridge-changing operations. This is limited by weather conditions and the presence of large-scale objects around the experimental site. When satellite signals are blocked, positioning errors may occasionally be higher than 50 centimeters. When this phenomenon occurs, it is not conducive to use this method to correct the heading angle of the robot.

Secondly, the robot heavily relies on deep learning technology for guidance. It is most effective in forming guide lines when the drip irrigation belts on the ridge are used as reference points, followed by the ridges, and lastly, crops. In practice, misalignment of crops in the ridge can lead to the failure to form a reliable guide line or result in an unreliable one. This phenomenon is closely associated with the experience and planting practices of farmers. To mitigate this and avoid excessive steering corrections, we take an average of the heading angles obtained from the three types of guidance lines.

In field experiments, the heading angle control method based on fuzzy logic is combined with four PID controllers to maintain a stable speed when the robot moves, making the robot move smoother. This fuzzy logic inference method can also be used to adjust the range of the base value of the membership function to output the steering angle based on the ridges of different widths to prevent the robot from damaging the ridges. It is worth noting that the operation time of the electric cylinder used for steering control is synchronized with the output update rate of the fuzzy controller. However, the delay in actuation time often leads to a mismatch with the robot's movement speed. In this study, when the robot is moving at the slowest speed of 12.5 cm/s, the guidance system will

 doi:10.20944/preprints202310.1427.v1

engage in heading angle control only when the output steering angle value from the fuzzy controller exceeds ±3 degrees.

Next, in the experiment evaluating the travel speed of the robot in the field, we observed that when the image processing speed was 7 fps, the guidance line could not be accurately identified by the guidance system in the field when travel speed of robot exceeded 20 cm/s. This was due to the fact that when the robot was moving at high speeds, the statistical data for the distribution of the center point of targets could not be obtained immediately, leading to a failure in fitting the guidance lines. In the future, it is possible to improve the computing performance of the guidance system or apply a real-time multi-target tracking framework to solve this problem [39].

Regarding the performance of weed and crop nutrient deficiency identification, when a sufficient number of samples were used for model training, and there was a clear color distinction between weeds and crops, weed identification accuracy exceeded 92%. However, the identification accuracy for crops with nutrient deficiencies ranged from 83% to 99%, with a significant variability. This result can be attributed to the strong lighting conditions causing this type of crop to be mistakenly classified as weeds or not correctly labeled. In the spraying test operations, the accuracy of weed spraying reached 87%, while the accuracy of nutrient deficiency spraying was approximately 73%. These results were lower than both weed identification accuracy and crop nutrient deficiency identification accuracy, suggesting a possible delay in the spraying process. Based on our initial assessment, it was concluded that motor activation delays occasionally occur within the spray module. Further in-depth analysis and research are still needed to fully understand these findings.

4. Conclusions

Agricultural robots rely on high-precision positioning systems for autonomous operations in the farmland. However, they face challenges due to weather conditions and the lack of network connectivity, which have hindered the widespread adoption of these advanced robots. The proposed system for robust guidance and precise spraying based on deep learning in this study enables four-wheeled robots to autonomously conduct spraying operations in strip fields. This approach has been proven effective in generating a guidance line from common features typically found on ridges. The generated guidance line serves as a reference for the robot to track and follow. By integrating multiple guidance lines, the robot can obtain reliable heading angles. Fuzzy logic inference is employed for computing appropriate steering angle, while PID controller is utilized to maintain a consistent robot velocity, ensuring smoother robot movement. This method is particularly suitable for ridged bed farming with drip irrigation belts (tubing) or neatly arranged crops and can be extended for precise spraying operations in large agricultural areas.

Author Contributions: Conceptualization, C.-L.C.; methodology, C.-L.C.; software, H.-W.C.; verification, H.-W.C.; data management, H.-W.C.; writing—manuscript preparation, C.-L.C. and H.-W.C.; writing—review and editing, C.-L.C.; visualization, C.-L.C.; supervision, C.-L.C.; project management, C.-L.C.; fund acquisition, C.-L.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Science and Technology Council, grant number MOST 111-2221-E-020-024; NSTC 112-2221-E-020-013.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Spykman, O.; Gabriel, A.; Ptacek, M.; Gandorfer, M. Farmers' perspectives on field crop robots—Evidence from Bavaria, Germany. *Comput. Electron. Agric.* **2021**, *186*, 106176.
2. Wu, J.; Jin, Z.; Liu, A.; Yu, L.; Yang, F. A survey of learning-based control of robotic visual servoing systems. *J. Franklin Inst.* **2022**, *359*, 556–577.

3. Kato, Y.; Morioka, K. Autonomous robot navigation system without grid maps based on double deep Q-Network and RTK-GNSS localization in outdoor environments. 2019 IEEE/SICE Int. Symp. Syst. Integration (SII), Paris, France, 346–351.
4. Galati, R.; Mantriota, G.; Reina, G. RoboNav: An affordable yet highly accurate navigation system for autonomous agricultural robots. *Robotics* **2022**, *11*, 99.
5. Chien, J.C.; Chang, C.L.; Yu, C.C. Automated guided robot with backstepping sliding mode control and its path planning in strip farming. *Int. J. iRobotics* **2022**, *5*, 16–23.
6. Tian, H.; Wang, T.; Liu, Y.; Qiao, X.; Li, Y. Computer vision technology in agricultural automation—A review. *Inf. Process. Agric.* **2020**, *7*, 1–19.
7. Choi, K.H.; Han, S.K.; Han, S.H.; Park, K.-H.; Kim, K.-S.; Kim, S. Morphology-based guidance line extraction for an autonomous weeding robot in paddy fields. *Comput. Electron. Agric.* **2015**, *113*, 266–274.
8. Suriyakoon, S.; Ruangpayoongsak, N. Leading point-based interrow robot guidance in corn fields. 2017 2nd Int. Conf. Control Robotics Eng. (ICCRE), Bangkok, Thailand, 8–12.
9. Bonadiesa, S.; Gadsden, S.A. An overview of autonomous crop row navigation strategies for unmanned ground vehicles. *Eng. Agric. Environ. Food* **2019**, *12*, 24–31.
10. Chen, J.; Qiang, H.; Wu, J.; Xu, G.; Wang, Z. Navigation path extraction for greenhouse cucumber-picking robots using the prediction-point Hough transform. *Comput. Electron. Agric.* **2021**, *180*, 105911.
11. Shi, J.; Bai, Y.; Diao, Z.; Zhou, J.; Yao, X.; Zhang, B. Row detection-based navigation and guidance for agricultural robots and autonomous vehicles in row-crop fields: Methods and applications. *Agronomy* **2023**, *13*, 1780.
12. Ruan, Z.; Chang, P.; Cui, S.; Luo, J.; Gao, R.; Su, Z. A precise crop row detection algorithm in complex farmland for unmanned agricultural machines. *Biosyst. Eng.* **2023**, *232*, 1–12.
13. Zhang, S.; Wang, Y.; Zhu, Z.; Li, Z.; Du, Y.; Mao, E. Tractor path tracking control based on binocular vision. *Inf. Process. Agric.* **2018**, *5*, 422–432.
14. Mavridou, E.; Vrochidou, E.; Papakostas, G.A.; Pachidis, T.; Kaburlasos, V.G. Machine vision systems in precision agriculture for crop farming. *J. Imaging* **2019**, *5*, 89.
15. Gu, Y.; Li, Z.; Zhang, Z.; Li, J.; Chen, L. Path tracking control of field information-collecting robot based on improved convolutional neural network algorithm. *Sensors* **2020**, *20*, 797.
16. Suriyakoon, S.; Ruangpayoongsak, N. Leading point-based interrow robot guidance in corn fields. 2017 2nd Int. Conf. Control Robotics Eng. (ICCRE), Bangkok, Thailand, 8–12.
17. Pajares, G., García-Santillán, I., Campos, Y., Montalvo, M., Guerrero, J.M., Emmi, L., Romeo, J., Guijarro, M., & González-de-Santos, P., Machine-vision systems selection for agricultural vehicles: A guide. *J. Imaging* **2016**, *2*, 34.
18. RegenerateHu, Y.; Huang, H. Extraction method for centerlines of crop row based on improved lightweight Yolov4. 2021 6th International Symposium on Computer and Information Processing Technology (ISCIPT), Changsha, China, 127◎132, 2021.
19. de Silva, R.; Cielniak, G.; Gao, J. (2021). Towards agricultural autonomy: Crop row detection under varying field conditions using deep learning. arXiv preprint arXiv:2109.08247.
20. Hu, D.; Ma, C.; Tian, Z.; Shen, G.; Li, L. Rice Weed detection method on YOLOv4 convolutional neural network. 2021 International Conference on Artificial Intelligence, big data and algorithms (CAIBDA), 41◎45.
21. Ruigrok, T.; van Henten, E.; Booij, J.; van Boheemen, K.; Kootstra, G. Application-specific evaluation of a weed-detection algorithm for plant-specific Spraying. *Sensors* **2020**, *20*, 7262.
22. Chang, C.L.; Xie, B.X.; Chung, S.C. Mechanical control with a deep learning method for precise weeding on a farm. *Agriculture* **2021**, *11*, 1049.
23. Wang, Q.; Cheng, M.; Huang, S.; Cai, Z.; Zhang, J.; Yuan, H. A deep learning approach incorporating YOLO v5 and attention mechanisms for field real-time detection of the invasive weed Solanum rostratum Dunal seedlings. *Comput. Electron. Agric.* **2022**, *199*, 107194.
24. Chen, J.; Wang, H.; Zhang, H.; Luo, T.; Wei, D.; Long, T.; Wang, Z. Weed detection in sesame fields using a YOLO model with an enhanced attention mechanism and feature fusion. *Comput. Electron. Agric.* **2022**, *202*, 107412.
25. Farooq, U.; Rehman, A.; Khanam, T.; Amtullah, A.; Bou-Rabee, M.A.; Tariq, M. Lightweight deep learning model for weed detection for IoT devices. In Proceedings of the 2022 2nd International Conference on Emerging Frontiers in Electrical and Electronic Technologies (ICEFEET), Patna, India, 2022; pp. 1–5.
26. Ruigrok, T.; van Henten, E.J.; Kootstra, G. Improved generalization of a plant-detection model for precision weed control. *Comput. Electron. Agric.* **2023**, *204*, 107554.
27. Suriyakoon, S.; Ruangpayoongsak, N. Leading point based interrow robot guidance in corn fields. In Proceedings of the 2017 2nd International Conference on Control and Robotics Engineering (ICCRE), Bangkok, Thailand, 8–12.

 doi:10.20944/preprints202310.1427.v1

28. Qiu, Q.; Fan, Z.; Meng, Z.; Zhang, Q.; Cong, Y.; Li, B.; Wang, N.; Zhao, C. Extended Ackerman steering principle for the co-ordinated movement control of a four wheel drive agricultural mobile robot. *Comput. Electron. Agric*. **2018**, *152*, 40–50.
29. Bonadiesa, S.; Gadsden, S.A. An overview of autonomous crop row navigation strategies for unmanned ground vehicles. *Eng. Agric. Environ. Food* **2019**, *12*, 24–31.
30. Tu, X.; Gai, J.; Tang, L. Robust navigation control of a 4WD/4WS agricultural robotic vehicle. *Comput. Electron. Agric*. **2019**, *164*, 104892.
31. Zhang, L.; Zhang, R.; Li, L.; Ding, C.; Zhang, D.; Chen, L. Research on virtual Ackerman steering model based navigation system for tracked vehicles. *Comput. Electron. Agric*. **2022**, *192*, 106615.
32. Bennett, P. The NMEA FAQ (Fragen und Antworten zu NMEA), Ver. 6.1, Sep. 1997.
33. Shih, P.T.-Y. TWD97 and WGS84, datum or map projection? *J. Cadastral Survey* **2020**, *39*, 1–12.
34. Chang, C.L.; Chen, H.W. Straight-line generation approach using deep learning for mobile robot guidance in lettuce fields. In Proceedings of the 2023 9th International Conference on Applied System Innovation (ICASI), Chiba, Japan.
35. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y. M. YOLOv4: Optimal speed and accuracy of object detection. arXiv:2004.10934.
36. Wang, C.Y.; Liao, H.Y. M.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Work-shops (CVPRW), Seattle, WA, USA, 1571–1580.
37. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU loss: Faster and better learning for bounding box regression. AAAI Technical Track: Vision **2020**, *34*, 12993–13000.
38. Murphy, K.P. Machine learning: A probabilistic perspective, MIT press, 2012.
39. Lee, C.C. Fuzzy logic in control system: Fuzzy logic controller. I. *IEEE Trans. Syst. Man Cybern*. **1990**, *20*, 404–418.
40. Meimetis, D., Daramouskas, I., Perikos, I.; Hatzilygeroudis, I. Real-time multiple object tracking using deep learning methods. *Neural Comput. & Applic*. **2023**, *35*, 89–118.