agronomy

MDPI

Article

Farmland Navigation Line Extraction Method Based on RS-LineNet Network and Root Subordination Relationship Optimization

Yanlei Xu, Zhen Lu, Jian Li, Yuting Zhai, Chao Liu, Xinyu Zhang and Yang Zhou *

College of Information and Technology, Jilin Agricultural University, Changchun 130118, China; yanleixu@jlau.edu.cn (Y.X.); 20231240@mails.jlau.edu.cn (Z.L.); lijiandz@jlau.edu.cn (J.L.); zhaiyuting@mails.jlau.edu.cn (Y.Z.); liuchao@mails.jlau.edu.cn (C.L.); zhangxinyu20050509@gmail.com (X.Z.)
* Correspondence: zhouyang@jlau.edu.cn

Abstract

Navigation line extraction is vital for visual navigation with agricultural machinery. The current methods primarily utilize plant canopy detection frames to extract feature points for navigation line fitting. However, this approach is highly susceptible to environmental changes, causing position instability and reduced extraction accuracy. To address this problem, this study aims to develop a robust navigation line extraction method that overcomes canopy-based feature instability. We propose extracting feature points from root detection frames for navigation line fitting. Compared to canopy points, root feature point positions remain more stable under natural interference and less prone to fluctuations. A dataset of corn crop row images under multiple growth environments was collected. Based on YOLOv8n (You Only Look Once version 8, nano model), we proposed the RS-LineNet lightweight model and introduced a root subordination relationship filtering algorithm to further improve detection precision. Compared with the YOLOv8n model, RS-LineNet achieves 4.2% higher precision, 16.2% improved recall, and an 11.8% increase in mean average precision (mAP_{50}), while reducing the model weight and parameters to 32% and 23% of the original. Navigation lines extracted under different environments exhibit an 0.8° average angular error, which is 3.1° lower than canopy-based methods. On Jetson TX2, the frame rate exceeds 12 FPS, meeting practical application requirements.

Keywords: visual navigation; deep learning; root feature point; subordination relationship filtering algorithm; edge deployment devices

check for updates

Academic Editor: Maofang Gao

Received: 25 July 2025
Revised: 21 August 2025
Accepted: 26 August 2025
Published: 28 August 2025

Citation: Xu, Y.; Lu, Z.; Li, J.; Zhai, Y.; Liu, C.; Zhang, X.; Zhou, Y. Farmland Navigation Line Extraction Method Based on RS-LineNet Network and Root Subordination Relationship Optimization. *Agronomy* **2025**, *15*, 2069. https://doi.org/10.3390/agronomy15092069

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the rapid growth of the global population, the demand for food continues to rise, making food-related issues one of the core challenges of economic development [1,2]. Despite significant advances in agricultural technology over the past few decades, agricultural production still largely relies on human labor, which is both costly and labor-intensive. To address this challenge, intelligent agricultural equipment has gradually emerged as a crucial means of enhancing production efficiency. For instance, the adoption of smart farming technologies, such as automated systems for precision irrigation and fertilization, has led to significant reductions in water and fertilizer consumption while improving crop yields [3–6]. Within this realm, the autonomous operation capability of agricultural robots is particularly vital, and navigation technology, as its core component, serves as a key guarantee for achieving efficient and precise operations [7–9].

Currently, research on navigation technology primarily focuses on two major fields: satellite navigation and visual navigation [10]. Satellite navigation utilizes a Global Positioning System (GPS) to provide positioning and path planning for agricultural machinery. However, in complex farmland environments, GPS signal obstructions can impede real-time performance [11]. To address the limitations of satellite navigation technology, visual navigation technology has gradually emerged as a research hotspot [12]. This technology acquires environmental information in real time through cameras, extracts path features from complex environments, and delivers accurate navigation data. Navigation lines, as important visual cues in the environment, have become a focal point of research. In the field of visual navigation line extraction, many researchers have conducted in-depth explorations using traditional image processing techniques, achieving significant progress. Zhang et al. [13] segmented an image; they employed horizontal bandings and vertical projection strategies to extract the center points of wheat rows. By integrating position clustering with the shortest path method, they identified the feature point set and fitted the crop rows accordingly. Although the average angular error was merely 0.5°, there is a need to enhance time efficiency. Yang et al. [14] divided a segmented image into horizontal bandings, extracted multilevel regions of interest (ROIs) and their micro-ROIs using a step-by-step sliding bounding box technique, and they fitted the navigation line after extracting the feature points. This approach resulted in an average angular error of 1.49° and an average single-frame processing time of 312.3 ms. Zhou et al. [15] proposed a crop row detection algorithm based on adaptive multi-ROI, which divides image bandings and gradually updates feature points within the ROI to ultimately fit crop row lines and navigation lines. The detection accuracy of this algorithm is 95.3%, with an average single-frame processing time of 240.8 ms. In addition, Zhang et al. [16] proposed a method for accurately obtaining feature points and extracting navigation lines during the soybean seedling stage, based on the average coordinates of pixel points in the soybean seedling band. The proposed algorithm achieved an average distance deviation of 7.38 and an average angle deviation of 0.32, with the fitted navigation line reaching an accuracy of 96.77%. When extracting navigation lines, the traditional image processing methods mentioned above do not rely on signal coverage, unlike satellite navigation technology. The accuracy and angular error of these methods can meet the needs of farmland operations, but there is a general problem of long computation times, which makes it difficult to balance accuracy and real-time performance.

In recent years, with the rise of deep learning [17], visual navigation techniques based on convolutional neural networks have been widely adopted for the extraction of navigation lines [18–24]. Wang et al. [18] proposed a method that combines vegetation index and ridge segmentation to extract feature points through horizontal banding, further optimizing navigation line extraction. The detection accuracy of this method reached 95.3%, with a frame rate of 10 frames per second (FPS). Gong et al. [19] implemented corn crop detection by optimizing the YOLOv5s backbone network and introducing an attention mechanism. They fitted the crop row lines using the center point of the detection frame as the feature point, and experiments demonstrated that the fitting error of the method was within 5°, with an average processing time of 53 ms. Diao et al. [20] proposed a novel spatial pyramid structure based on the YOLOv8s model to enhance the detection accuracy of corn plant cores. They utilized the center point of the detection frame for corn plant cores as the feature point to fit the crop row line, achieving an average fitting error of 0.63° and an average processing time of 45 ms. Ju et al. [21] utilized the improved MW-YOLOv5s model to identify rice seedlings and establish a navigation line by fitting straight lines through the center point of the detection frame. The experiment confirmed that the seedling injury rate for this method was 2.8%, with a frame rate of 19.51 FPS. Cao

et al. [22] proposed the YOLOv8n-Trunk model, which generates feature points and fits navigation lines by detecting vine trunks. The network achieves a detection accuracy of 92.7% for trunks, and experiments have demonstrated that the navigation paths derived from the detection results are reliable. Liu et al. [23] detected pineapples using an enhanced YOLOv5 model, which employed the inverse perspective transformation of the detection frames to extract center points for straight-line fitting. This approach effectively improved the extraction accuracy of navigation lines within high-density crop rows, resulting in an average fitting error of 3.54°. For corn crops, Diao et al. [24] proposed a method based on the Swin Transformer-YOLOv8s network, which achieves an average angular error of 0.58° and a processing time of 47 ms.

The above visual navigation line extraction algorithm, when combined with neural networks, generally exhibits superior performance in terms of computing time while meeting the accuracy requirements for practical applications. Currently, this type of algorithm primarily relies on canopy detection frames to extract feature points and to fit crop row lines and navigation lines. However, this approach faces notable limitations. Firstly, the horizontal deviation between the canopy and the root system can lead to farm machinery inadvertently crushing crops. Secondly, natural disturbances, such as wind, may induce fluctuations in feature points, consequently diminishing the fitting accuracy. Furthermore, when the edges of the photographed images display incomplete plants, the deviation between the feature points extracted using the canopy detection frame and the actual feature points becomes substantial, adversely affecting navigation line extraction. The fundamental reason for these issues is that the characteristic points of the canopy are highly susceptible to environmental changes, resulting in their positions being unstable. In contrast, the location of crop roots is relatively fixed and less influenced by the natural environment. Therefore, extracting feature points based on root detection frames can effectively mitigate these problems and is more suitable for the precise positioning of navigation lines in agricultural operations. To highlight the advantage of root-based feature extraction for robust and precise navigation line fitting in dynamic field conditions, Figure 1 provides a comparative visualization of navigation lines extracted from canopy-based and root-based feature points. Under natural disturbances such as westward winds, the canopy-based feature point-fitted navigation line deviates significantly to the left due to the movement of corn leaves, increasing the risk of farm machinery damaging crops. In contrast, the root-based feature point-fitted navigation line remains stable and aligned with the actual crop rows, as root positions are minimally affected by such environmental factors.

In root-based feature point extraction and fitting methods, traditional image processing techniques have achieved certain research progress. For instance, Gong et al. [25] proposed a method for extracting navigation lines from the composite positioning points of corn stems and roots, achieving an accuracy rate of 93.8%. However, this approach suffers from low computational efficiency, rendering it inadequate for real-time applications. Meanwhile, methods incorporating neural networks have also yielded notable results. Zheng et al. [26] utilized an improved YOLOX-Nano model to detect the roots of jujube trees, extracting the bottom center point as the feature point. They then determined the navigation line by combining K-means clustering with geometric relationships, resulting in an average heading deviation of 2.55° (Table 1).

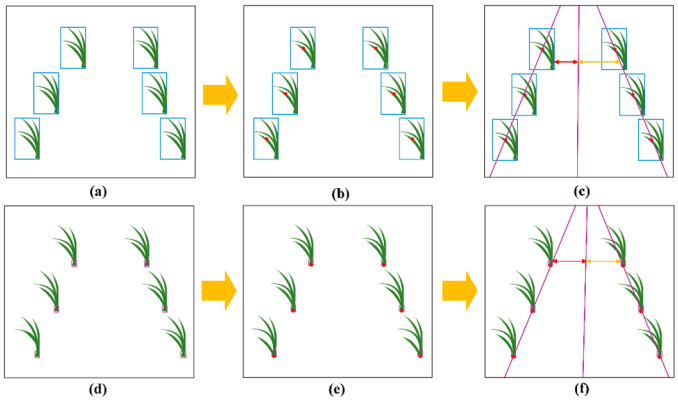


Figure 1. Comparative visualization of canopy-based and root-based navigation line fitting processes under natural disturbances. The red double arrows and the yellow double arrows respectively represent the distances of the extracted navigation lines from the crop ridges on both sides. (**a**) Detection of corn plants; (**b**) extract feature points based on the detection frame of corn plants; (**c**) extract navigation lines based on canopy feature points; (**d**) root detection of corn plants; (**e**) extract feature points based on the root detection frame of corn plants; (**f**) extract navigation lines based on root system feature points.

Table 1. Comparative analysis of representative methods for navigation line extraction.

Reference	Dataset Size	Method	Average Angular Error (°)	Average Processing Time (ms)
Zhang et al. [13]	300	Traditional image processing techniques	0.5	/
Yang et al. [14]	1000	Traditional image processing techniques	1.49	312.3
Zhou et al. [15]	2890	Traditional image processing techniques	1.63	240.8
Zhang et al. [16]	/	Traditional image processing techniques	0.32	290
Gong et al. [25]	400	Traditional image processing techniques	Within 5	298
Wang et al. [18]	2000	Deep learning	/	100
Gong et al. [19]	7000	Deep learning	Within 5	53
Diao et al. [20]	12,000	Deep learning	0.63	45
Ju et al. [21]	36,000	Deep learning	/	51.3
Cao et al. [22]	1300	Deep learning	/	11.1
Liu et al. [23]	2664	Deep learning	3.54	57.6
Diao et al. [24]	10,000	Deep learning	0.58	47
Zheng et al. [26]	2084	Deep learning	2.55	/

Although the methods for extracting crop row lines and navigation lines based on root detection frames have achieved certain progress, their application in corn crops has not yet been fully explored. Corn is a major crop in Northeast China, with a cultivation area exceeding 14 million hectares and contributing over 30% of the national maize output. Its large-scale and highly mechanized production demands precise navigation technology to enhance operational efficiency and reduce crop damage. Compared to jujube trees, the target size of corn roots is smaller, and their color closely resembles that of the soil, which

increases the difficulty of detection. In response to the aforementioned issue, this paper proposes an improved solution for the YOLOv8n model (You Only Look Once version 8, nano model), focusing on the precise detection of corn plants and their roots. By integrating a hierarchical filtering strategy with the least squares method, the accurate extraction of navigation lines is achieved based on the model's prediction results. This method is referred to as RS-LineNet.

2. Materials and Methods

2.1. Workflow of the Proposed Navigation Line Extraction Method

The overall workflow of the proposed method is as follows:

1. Data acquisition and model training—corn crop row images are collected under multiple environmental conditions and annotated, followed by training and prediction using the RS-LineNet network.
2. Root detection optimization—based on the prediction results, a subordination relationship filtering algorithm is proposed to analyze the spatial correlation between corn plants and root detection frames, removing isolated misdetections that do not correspond to actual root locations.
3. Feature point extraction and navigation line fitting—feature points corresponding to the same crop ridge are then clustered using a clustering algorithm [27]. Subsequently, the crop row lines are fitted using the least squares method, which allows for the extraction of the navigation line.

The overall flow of the method is illustrated in Figure 2.

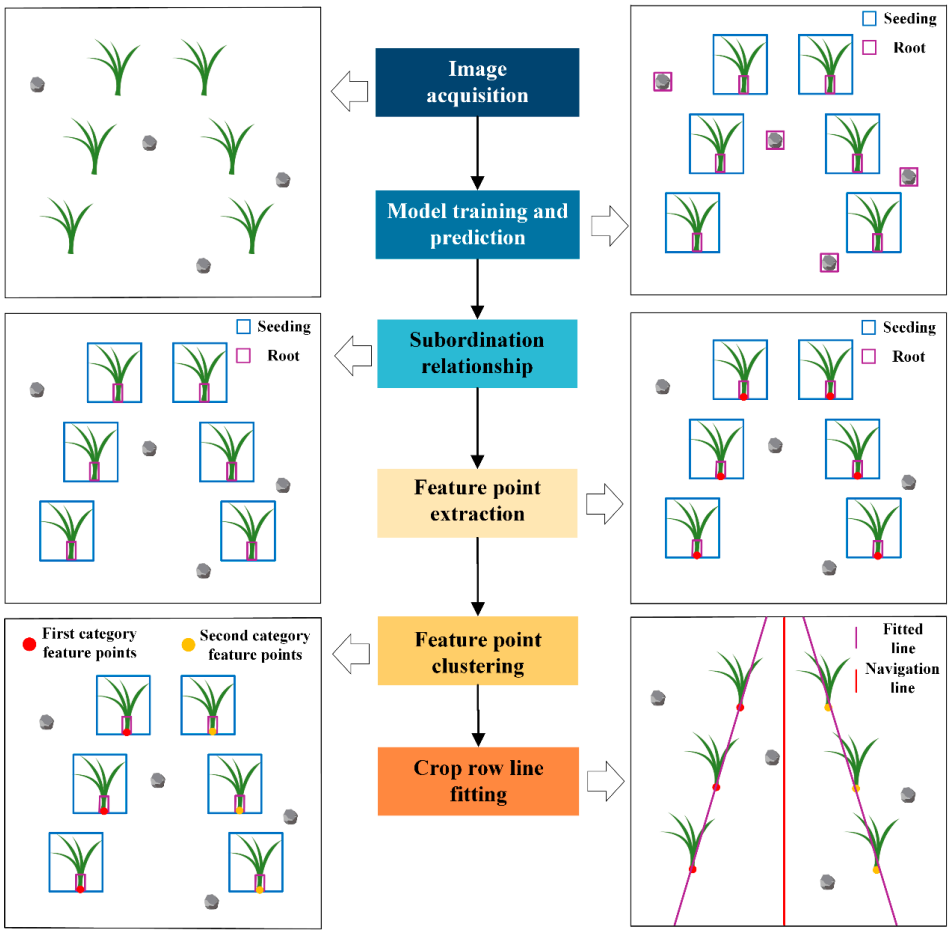


Figure 2. The farmland navigation line extraction flow based on RS-LineNet and subordination relationship optimization.

2.2. Dataset Establishment

All images of corn crop rows in this paper were captured using a smartphone (Redmi Note 11 Pro; Xiaomi Inc., Beijing, China) in the experimental research field of Jilin Agricultural University, located in Changchun, Jilin Province (125.41° E, 43.81° N). Photography of corn crop rows under various growth environments was conducted from 25 May to 15 June 2024. The different growth environments include normal growth, weed symbiosis, adhesive growth, and seedling-missing growth, among other conditions. A total of 1422 original images were collected, and the dataset was divided into training, validation, and testing sets in a ratio of 8:1:1. In this study, we utilized the labeling tool Labelimg (version 1.8.6) to annotate the corn plants and their roots. Due to the frequent occurrence of adhesion and mutual occlusion between the leaves during the growth period of corn, it was often more challenging to accurately label some independent plants. Therefore, in some scenes, multiple corn plants were labeled as whole during the annotation process. To increase the diversity of the data and more effectively capture the key features of corn plants and their roots, data augmentation techniques such as horizontal flipping, noise addition, and motion blur were applied to the images in the dataset. The enhanced dataset contains 5682 images. As shown in Figure 3, the dataset images and labeling examples are presented, with red circles highlighting representative issues, including weed symbiosis, adhesive growth, seedling-missing growth, and overall annotations of leaf adhesion.

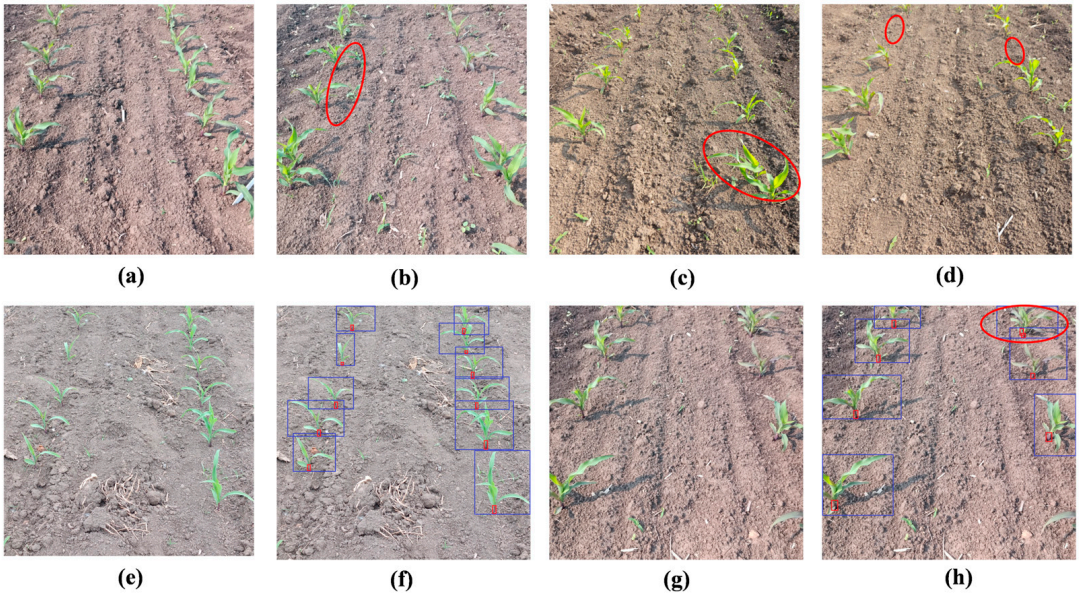


Figure 3. Example display of dataset image and annotation. The blue boxes indicate the labeled seedling frames, while the red boxes represent the labeled root frames. (**a**) Normal growth; (**b**) weed symbiosis, with the red circle highlighting dense weed growth in the vicinity of the corn; (**c**) adhesive growth, with the red circle indicating closely attached corn plants; (**d**) seedling-missing growth, with red circles marking missing plants in the row; (**e**) no leaf adhesion in the original image; (**f**) no overall annotation; (**g**) leaf adhesion in the original image; (**h**) overall annotation, with the red circle indicating the region of overall annotation.

2.3. RS-LineNet Model Establishment

In a complex farmland environment, the roots of corn plants are categorized as micro-targets due to their small dimensions and color similarity to the soil, resulting in relatively low detection accuracy. To address this issue, this paper optimizes the structure of the neck and Head in the YOLOv8n model and prunes the optimized model to construct the RS-LineNet model. This approach aims to enhance the model's capability of detecting the

roots of micro-sized corn plants in complex farmland environments while achieving model lightweight. The RS-LineNet model structure proposed in this paper is shown in Figure 4.

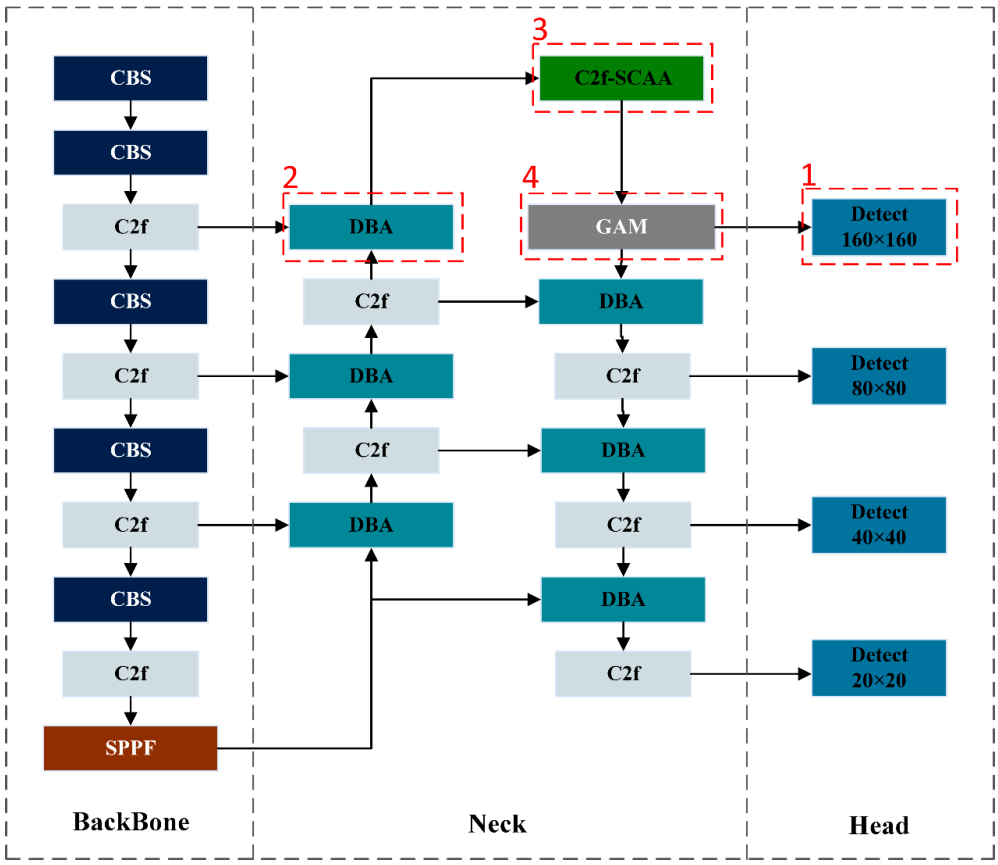


Figure 4. The architecture diagram of the RS-LineNet model. The red dashed boxes highlight: 1. Micro-target detection head module; 2. DBA module; 3. C2f-SCAA module; 4. GAM module.

In the head section, the original detection head structure primarily targets medium and large-sized targets, making it challenging to capture the effective features of the corn plant roots. This paper introduces an additional micro-target detection head module, which is seamlessly integrated into the existing detection head, as shown in Position 1 of Figure 4. This enhanced detection head structure improves the extraction of high-resolution features and significantly boosts the model's ability to capture details of micro-targets, thereby establishing a solid foundation for the optimization of subsequent modules. Furthermore, to enhance the overall detection efficiency and accuracy of the model, this paper replaces the loss function from CIoU [28] to PIoU2 [29]. The PIoU2 loss function optimizes the anchor box regression path and gradient adjustment strategy, which not only accelerates the model's convergence speed but further enhances the accuracy of the detection frame.

In the neck section, this paper proposes a lightweight boundary aggregation module (DBA) based on the selective boundary aggregation module (SBA) [30], as depicted in position 2 of Figure 4. By effectively combining shallow, detailed information with deep semantic information, the DBA significantly enhances the model's capability to accurately depict and locate the target contour. To address the relatively high computational cost associated with the SBA, the DBA employs Deep Separable Convolution (DWConv) in place of standard convolution (Conv). This modification reduces computational complexity while preserving detection accuracy, enabling more efficient edge feature processing and aggregation. Furthermore, to enhance the detection accuracy and robustness of the model in complex scenes, this paper improves the CSP Bottleneck with 2 Convolutions (C2f) feature

fusion module within the YOLOv8n architecture. The Bottleneck in C2f presents challenges, including insufficient global context modeling and redundant receptive field interference, which restrict its ability to perceive micro-targets. To mitigate these issues, we propose the CS_CAA module. By substituting the Bottleneck structure with two tandem CS_CAA modules, we derive the C2f_SCAA module, as shown in position 3 of Figure 4. This module employs context fusion and attention mechanisms to enhance multi-scale feature perception and inter-channel information interaction, significantly reducing redundant information interference and improving detection accuracy for micro-targets. To address the decline in the performance of corn plant root detection caused by factors such as variations in light, weed interference, and similar coloration, this paper introduces and optimizes the Global Attention Mechanism (GAM) [31] module, as depicted in position 4 of Figure 4. This module integrates channel and spatial attention, thereby enhancing the model's capacity to express significant features and improving its adaptability and stability in complex environments. To address the issue of excessive redundant features in the output stage of GAM, the optimized GAM module incorporates a 1×1 pointwise convolution, enabling efficient feature channel recombination and compact processing. This approach reduces redundant features and enhances the model's feature integration capability, thereby improving its flexibility and adaptability.

After the structural optimization of the model, to reduce the network parameter occupancy and decrease the model complexity, this paper introduces the Layer-Adaptive Sparsity For The Magnitude-Based Pruning (LAMP) [32] algorithm based on vector magnitude. The algorithm dynamically adjusts the pruning ratio by quantifying the significance of the parameters in the network layer. This approach retains the key feature expression capability while effectively reducing redundant parameters, thereby ensuring the stability of detection accuracy and enhancing the real-time performance of the model's detection.

2.3.1. Micro-Target Detection Head Module-Improving the Fundamental Detection Capability of the Model

To enhance the YOLOv8n model's ability to detect micro-targets at the roots of corn plants, this paper introduces an additional detector head module, specifically designed for micro-targets detection, building upon the original detector head structure (as shown in Position 1 of Figure 4). The original YOLOv8n detection head primarily focuses on medium- to large-sized targets. When confronted with micro-targets, the model exhibits low detection performance due to insufficient feature resolution and limited detail capture capabilities. To address this issue, the newly added micro-target detection head enhances the extraction of shallow detail information by introducing a high-resolution feature layer and achieves multi-scale information fusion by combining it with deep semantic features. As a result, the model's ability to capture boundary structures and local features of micro-targets is significantly improved. Furthermore, this module optimizes the multi-scale detection mechanism, resulting in more accurate representations of targets of varying sizes within the feature layer. In complex farmland scenes, even when confronted with challenges such as changes in lighting, weed interference, and similar coloration, this module demonstrates strong robustness and detection accuracy. This effectiveness lays a solid foundation for subsequent module optimization and comprehensively enhances the overall performance of the model in micro-target detection tasks.

2.3.2. DBA Module-Improve Model Positioning Accuracy

To enhance the YOLOv8n model's capability to capture both edge and detail features when detecting the micro-targets of corn plant roots, this paper proposes the DBA based on the SBA. The DBA is applied to the neck of YOLOv8n, significantly improving the model's ability to depict and localize target contours accurately by fusing shallow detail

information with deep semantic information. To address the high computational cost of the SBA, the DBA employs DWConv in place of Conv, which effectively reduces computational complexity while maintaining detection accuracy, thereby enabling more efficient edge feature processing and aggregation. The structure of the DBA module is shown in Figure 5.

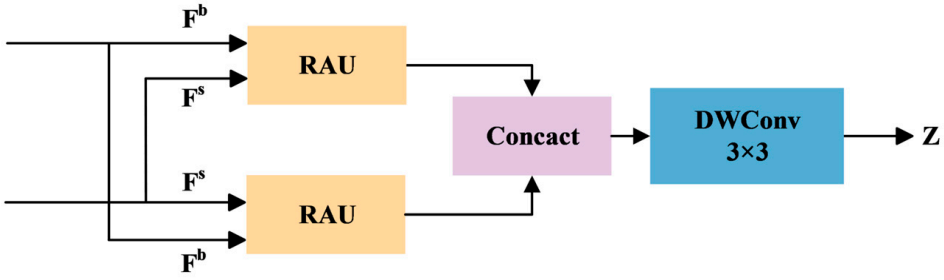


Figure 5. The architecture of the improved lightweight boundary aggregation DBA module.

In the DBA module, F^b represents shallow, detailed information, while F^s denotes deep semantic information. These two types of information are processed through two recalibration attention units (RAUs). RAU modules in distinct manners to compensate for the missing semantic information in the shallow features and the missing detailed information in the deep features. Subsequently, the output feature maps of the two RAU modules are concatenated through a channel connection operation (Concact). Finally, the final output of the module is obtained via a 3 × 3 DWConv. This aggregation strategy achieves a robust fusion of different features and refinement of coarse features. The DBA can be expressed as Equation (1) [33]:

$$Z = DWC_{3\times3}(Concat(PAU(F^s, F^b), PAU(F^b, F^s))) \quad (1)$$

where Z represents the final output of the DBA module, $DWC_{3\times3}$ represents the DWConv with a convolution kernel size of 3 × 3, and $PAU(\cdot,\cdot)$ is the block function of RAU, where the structure of *PAU* is shown in Figure 6:

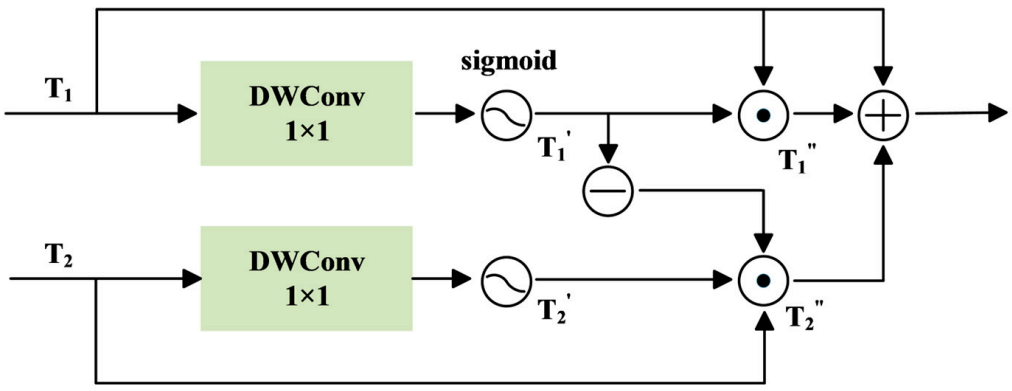


Figure 6. The structure of the *PAU* block.

Within the *PAU* module, the two input features, T_1 and T_2, undergo DWConv and a sigmoid activation function, which reduces the dimensionality of the input features to 32, resulting in T_1' and T_2'. Subsequently, the information relevant to the current task within the feature map is reinforced by performing pointwise multiplication of T_1 and T_1' to obtain T_1'', while the reverse operation is applied to T_1' to refine the imprecise and rough estimation into an accurate and complete prediction map. This refined map is then pointwise multiplied with T_2 and T_2' to enhance the information in the feature map related to the current task,

resulting in T_2''. Finally, T_1, T_1'', and T_2'' are superimposed to produce the output feature map of the *PAU* module. The *PAU* can be expressed as Equation (2) [33]:

$$PAU(T_1, T_2) = T_1{}' \odot T_1 + T_2{}' \odot T_2 \odot (\ominus(T_1{}')) + T_1 \quad (2)$$

In the above Equation, $PAU(T_1, T_2)$ represents the *PAU* operation. T_1 and T_2 are the two inputs of the *PAU* module, while T_1' and T_2' denote the intermediate states after passing through DWConv and activation functions. The symbol $\odot$ indicates the point-by-point multiplication operation, and $\ominus$ represents the reverse operation.

2.3.3. C2f_SCAA Module-Enhancing the Model's Detection Accuracy and Robustness

To enhance the detection accuracy and robustness of the YOLOv8n model for the micro-targets of corn plant roots in complex scenes, this paper focuses on the issues of insufficient global context modeling and redundant receptive field interference within the Bottleneck structure of the C2f feature fusion module. To address these challenges, we propose the CS_CAA module, which replaces Bottleneck with two series-connected CS_CAA modules, thereby forming the C2f_SCAA module. This modification significantly enhances multi-scale feature extraction and inter-channel interaction capabilities by enriching contextual information and reducing the interference from redundant receptive fields. Consequently, this improvement leads to better detection performance of the model for micro-targets in complex environments.

The CS_CAA module employs a residual structure design that first transforms the input feature map into a set of concise regional feature maps using 3×3 ordinary convolution. It then introduces a channel shuffle operation, which enhances the efficiency of inter-channel information interaction and increases the diversity of feature expression by rearranging the channels. Building on this foundation, the module integrates the Context Anchor Attention (CAA) mechanism [34] to assign weights to the feature graph post-shuffle, thereby highlighting salient features and suppressing irrelevant information. Subsequently, the CS_CAA module conducts morphological filtering on the feature maps through branch structures that utilize dilated convolution with varying dilation rates. Different convolution operations process regional feature maps of varying sizes based on specific receptive fields, thereby avoiding interference from redundant receptive fields and ensuring the diversity and integrity of feature representations. Then, the feature maps output from the branches are concatenated and subsequently feature-fused using 1×1 pointwise convolution. The introduction of pointwise convolution not only facilitates the alteration of channel dimensions but also enables the integration of information across branches, resulting in more compact and efficient fused features. Finally, by employing a residual connection with the input feature map, the module preserves the detailed information of the input features and mitigates the gradient vanishing problem, ultimately producing the final output feature map. This module design effectively addresses the challenges of insufficient global context modeling and redundant receptive field interference within the Bottleneck of C2f, thereby enhancing the model's detection accuracy and robustness for micro-targets in complex scenarios. The structure of C2f_SCAA is illustrated in Figure 7.

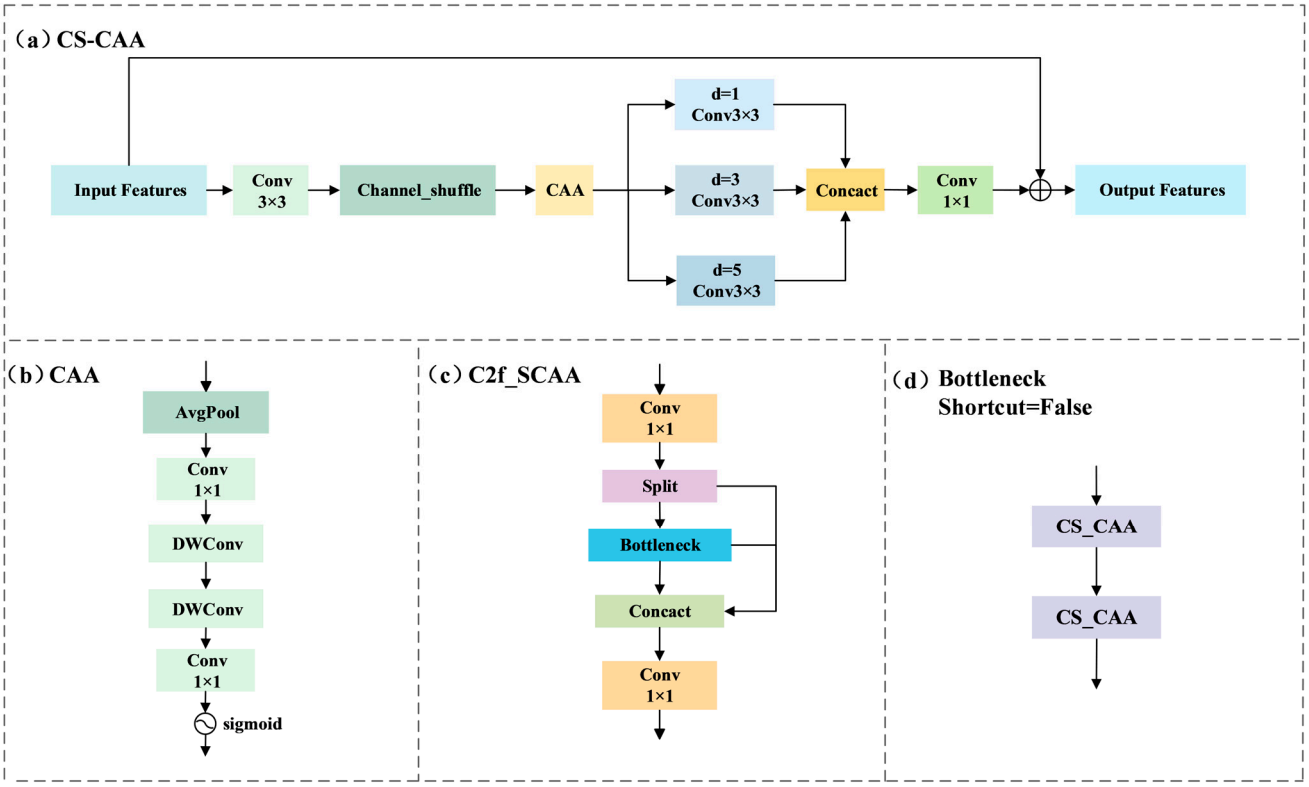


Figure 7. The structure of the C2f_SCAA module. (**a**) The structure of the CS_CAA module; (**b**) The structure of the CAA module; (**c**) The structure of the C2f_SCAA module; (**d**) The structure of the Bottleneck module with Shortcut=False, which is replaced by two tandem CS_CAA modules.

2.3.4. Optimized GAM Module Improves the Model's Immunity to Interference

To address the challenges faced by the model in detecting corn plant roots in complex environments, such as the close color similarity between roots and soil, variations in lighting, and interference from weeds, this paper introduces and enhances the GAM based on the YOLOv8n model. The optimized module retains the channel attention and spatial attention design of the original GAM. The channel attention mechanism preserves cross-dimensional information through a three-dimensional arrangement. It employs a multi-layer perception (MLP) to perform operations that reduce and increase the dimensionality of features, ultimately generating channel weight coefficients. Subsequently, the consistency of the feature structure is restored through inverse permutation and activation functions, and it is multiplied point-by-point with the original features to enhance the representation of salient characteristics. The spatial attention component fuses spatial information through two 7×7 Conv to generate spatial weight coefficients, which are normalized by the activation function and multiplied point-by-point with the channel-weighted feature maps to further optimize the spatial feature representation. Building on this, this paper adds a new 1×1 pointwise convolution in the GAM output stage to perform channel reorganization and compact processing of the optimized features. This addition not only reduces redundant features but also enhances feature integration capability, providing greater flexibility and adaptability for the model. The GAM process is represented by Equations (3) and (4):

$$F_2 = M_c(F_1) \odot F_1 \tag{3}$$

$$F_3 = M_s(F_2) \odot F_2 \tag{4}$$

In the above Equations, the input feature map of the GAM is denoted as F_1. M_c represents the channel attention module, while M_s denotes the spatial attention module. The intermediate state after passing through the channel attention module is referred to as F_2, and the intermediate state after passing through the spatial attention module is represented as F_3. The optimized structure of the GAM is illustrated in Figure 8.

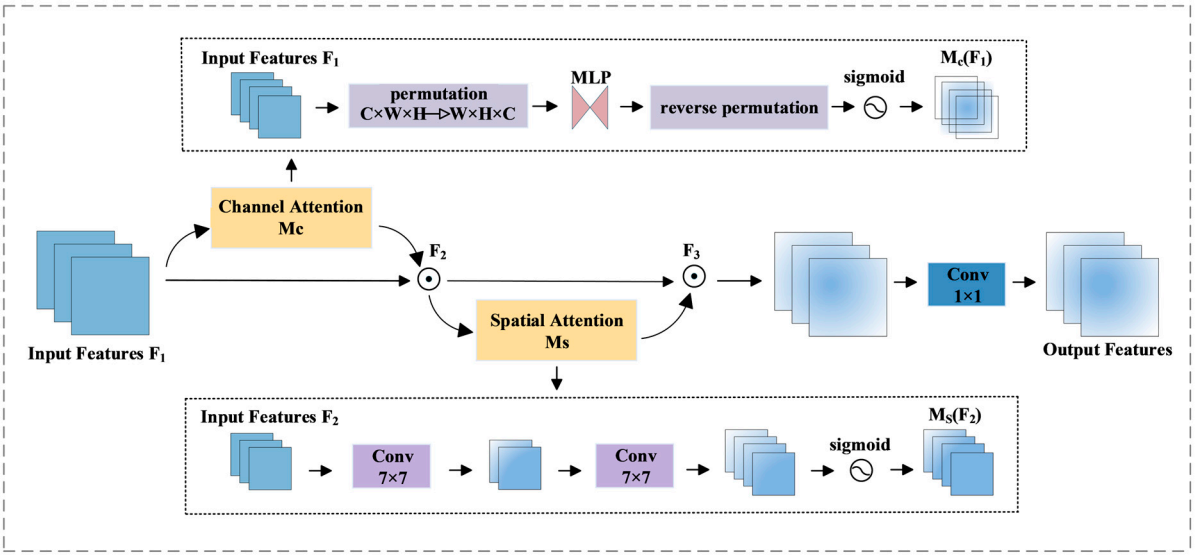


Figure 8. The structure of the GAM module.

2.3.5. PIoU2 Loss Function-Improving Model Detection Performance

CIoU is utilized as the loss function in YOLOv8n, with Equations (5)–(7) representing the expressions for the CIoU loss function:

$$L_{CIoU} = 1 - IoU(A,B) + \frac{\rho^2(A_{ctr}, B_{ctr})}{c^2} + \alpha v_t \tag{5}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^g}{h^g} - \arctan\frac{w}{h}\right)^2 \tag{6}$$

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{7}$$

In the above Equations, L_{CIoU} denotes the CIoU loss function; A, B denote the prediction frame and the actual frame; $IoU(A,B)$ denotes the ratio of the intersection to the union of A and B; (A_{ctr}, B_{ctr}) represents the center point of A and B; ρ represents the Euclidean distance between two central points; c denotes the diagonal distance of the smallest frame region that can contain the predicted and real frames; α is the weight function; v_t is the aspect ratio penalty term, which measures the similarity of aspect ratios; w^g and w represent the widths of the ground truth box and the predicted frame, respectively; h^g and h represent the heights of the ground truth box and the predicted frame, respectively.

The observation of the Equation reveals that when the width-height ratios of the real and predicted frames are the same, and the aspect ratio penalty term, v_t, is constant at 0, which cannot effectively lead the optimization; w and h are a pair of opposite numbers, and neither can increase or decrease at the same time, limiting the flexibility of the regression path. These problems can cause the prediction frame to swell during the regression process, which in turn affects the convergence speed and accuracy.

The PIoU loss function, an improvement over the CIoU loss function, effectively addresses the aforementioned issues and increases the convergence. However, detection accuracy remains relatively low in complex environments. Consequently, this paper pro-

poses an optimization of the PIoU function, resulting in the PIoU2 loss function. The PIoU2 loss function improves the model's focus on medium-quality prediction frames by incorporating a non-monotonic attention function, thereby enhancing the overall performance of target detection. Equations (8) to (13) present the Equations for the PIoU2 loss:

$$L_{PIoU2} = u(\lambda q) \cdot L_{PIoU} = 3 \cdot (\lambda q) \cdot e^{-(\lambda q)^2} \cdot L_{PIoU} \tag{8}$$

$$P = \frac{\frac{dw1}{w^{gt}} + \frac{dw2}{w^{gt}} + \frac{dh1}{h^{gt}} + \frac{dh2}{h^{gt}}}{4} \tag{9}$$

$$q = e^{-P}, q \in (0, 1] \tag{10}$$

$$u_a(x) = 3x \cdot e^{-x^2} \tag{11}$$

$$L_{PIoU} = L_{IoU} + 1 - e^{-p^2}, 0 \leq L_{PIoU} \leq 2 \tag{12}$$

$$L_{IoU} = 1 - IoU \tag{13}$$

L_{PIoU2} denotes the PIoU2 loss function. P represents the discrepancy between the predicted frame and the real frame; $dw1$, $dw2$, $dh1$, and $dh2$ are the absolute values of the distances between the predicted frame and the corresponding edges of the real frame, and w^{gt} and h^{gt} represent the width and height of the real frame, respectively. The parameter q denotes the quality assessment of the anchor frame and takes values from 0 to 1. $q = 1$ for $p = 0$, which means that the anchor and target frames are perfectly aligned. λ is a hyperparameter that controls the range and strength of the attention mechanism $u_a(x)$. $u_a(x)$ is the attention mechanism, and x represents its input; the attention function is characterized by non-monotonicity; when q is large, indicating high-quality anchor frames, the attention gradually decreases, when q is moderate, corresponding to medium-quality anchor frames, the attention reaches its peak to prioritize their optimization; when q is small, reflecting low quality anchor frames, the attention remains low to minimize interference with the optimization process. L_{PIoU} denotes the PIoU loss function, L_{IoU} denotes the IOU loss function, and $1 - e^{-p^2}$ is a smoothing function used to adaptively adjust the effect of the penalty factor.

PIoU2, by redistributing gradient weights, enables the model to focus more on optimizing medium-quality anchor boxes during the learning process, rather than solely relying on high-quality anchor boxes. This approach improves the localization and classification capabilities for micro-targets. Additionally, PIoU2 introduces only a single hyperparameter, λ, which simplifies the hyperparameter tuning process and demonstrates relatively high practical applicability.

2.3.6. LAMP Pruning Algorithm—Achieving Model Lightweighting

To reduce the occupancy of network parameters and decrease the model's complexity, facilitating its deployment on edge devices, this paper further introduces the LAMP pruning algorithm, which optimizes the RS-LineNet model structurally. Unlike traditional pruning algorithms, the LAMP pruning algorithm dynamically adjusts the sparsity across different layers, effectively mitigating the risk of layer functionality failure. The iterative process of the algorithm is illustrated in Figure 9:

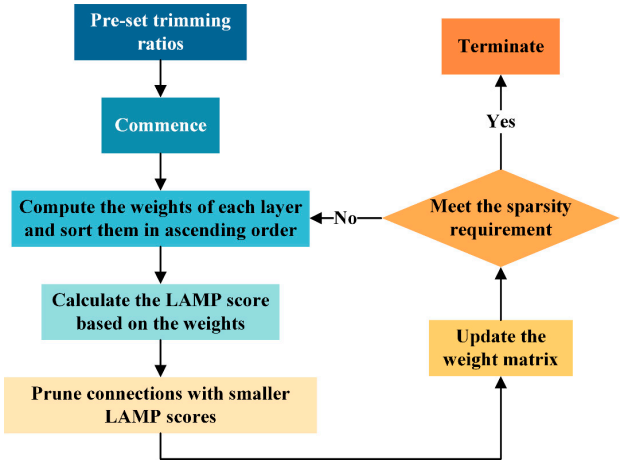


Figure 9. The iterative process of the LAMP pruning algorithm.

The core idea of the LAMP pruning algorithm is to achieve hierarchical adaptive pruning through weight magnitudes. Specifically, each weight tensor is unfolded into a one-dimensional vector, W, the magnitude $|W|$ of each vector W is computed, and these magnitudes are sorted in ascending order. Assuming that u and v both represent the indexes of the sorted magnitudes, $|W[u]|$ and $|W[v]|$ represent the vector magnitudes corresponding to the indexes u and v, respectively. After sorting, it satisfies $|W[u]| \leq |W[v]|$, and $u < v$ holds simultaneously. According to the sorting results of the weight magnitudes, the LAMP scores corresponding to each weight are computed as score$(u; W)$, as shown in Equation (14):

$$score(u; W) := \frac{(W[u])^2}{\sum_{v \geq u} (W[v])^2} \tag{14}$$

The LAMP score prioritizes pruning by measuring the relative importance of weights in the current layer. Its denominator represents the sum of the magnitudes of the weights in the current layer that are more important than index u. As index u increases, the magnitude, $|W[u]|$, also increases, and the number of weights greater than $|W[u]|$ gradually decreases, leading to a decrease in the denominator and a corresponding increase in the value of the numerator, and a gradual increase in the LAMP score. The lower the LAMP score of a weight, the lower its importance, and it is preferentially removed in pruning. According to the preset pruning ratio, the algorithm prioritizes the pruning of connections with smaller scores while dynamically adjusting the layer sparsity to meet the global sparsity requirements. Moreover, the calculation mechanism of the LAMP score ensures that at least one optimal channel with a score of 1 is retained in each layer, fundamentally avoiding the occurrence of layer function failure issues.

2.4. Subordinate Relationship Filtering Algorithm

When using the proposed RS-LineNet detection model to detect the root regions of corn plants, the model still tends to be affected by various environmental interferences, generating a relatively large number of false positive detection frames and resulting in a high misdetection rate. The common interference factors primarily stem from the following aspects: the soil color often closely resembles that of the corn plant roots, making it challenging for the model to effectively differentiate between the roots and the soil and thereby increasing the likelihood of misdetections. Additionally, lighting variations under different weather conditions interfere with the model's predictions. Under strong sunlight, overexposure and shadows may conceal the detailed information of corn-root regions, making it difficult for the model to perform accurate detection. Moreover, impurities, stones,

or other plant residues present in complex field environments often exhibit shapes and textures like those of corn roots, which can also easily lead the model to misclassify them as plant roots. The interference from these combined factors poses significant challenges to the precision of the model's detection.

To cope with the interference brought about via the aforementioned environmental factors, this paper innovatively proposes a processing strategy based on the subordination relationship of combined detection frames to filter root detection frames, building upon model optimization. This strategy can reduce the misdetection rate based on the RS-LineNet model predictions, further enhancing the reliability and stability of the output results.

This paper defines two types of labels: 'seeding' and 'root'. 'seeding' labels are designated as the primary detection targets, while 'root' labels serve as subordinate objects. To determine whether a certain 'root' detection frame is subordinate to a 'seeding' detection frame, a master–slave hierarchical screening algorithm based on the overlapping region of the bounding box is proposed. The structure of the algorithm, illustrated in Figure 10, comprises four components: model prediction results, detection frame classification and information conversion, the judgment of whether there is an overlapping region between two categories of detection frames, and the screening of associated detection frames with result storage. The specific process is described as follows:

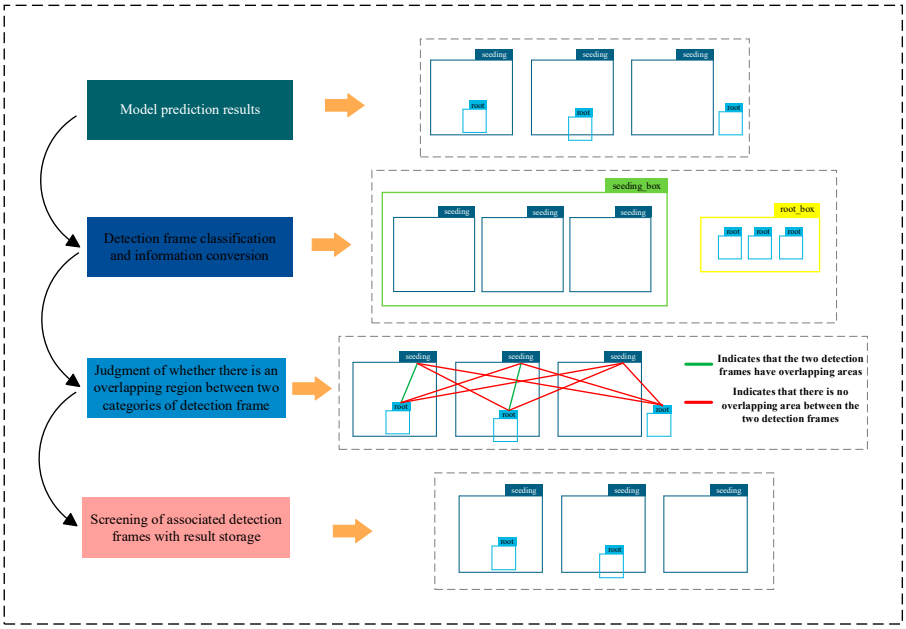


Figure 10. The diagram of the subordination relationship-filtering algorithm structure based on a combinatorial detection frame.

Step 1: model prediction results.

The prediction of the test set images is based on the model trained using the training set images. The prediction results can be categorized into three distinct cases: (1) the root detection frame is entirely contained within at least one seeding detection frame; (2) the root detection frame partially overlaps with at least one seeding detection frame; (3) there is no overlap between the root detection frame and the arbitrary seeding detection frame.

Step 2: detection frame classification and information conversion.

(1) Based on the label information in the prediction results, detection frames are divided into seeding_boxes (Class A) and root_boxes (Class B).

The YOLO model outputs detection frames in a normalized format: $\{class, x_{\text{center}}, y_{\text{center}}, w_f, h_f\}$, where the class label takes values of 0 or 1, representing

the categories 'seeding' and 'root', respectively. According to the class label, detection frames are classified into two groups: seeding_boxes (Class A) and root_boxes (Class B).

Here, *class* represents the label category, where 0 corresponds to 'seeding', and 1 corresponds to 'root'; $(x_{\text{center}}, y_{\text{center}})$ are the normalized coordinates of the center point of the detection frame; w_f, h_f are the normalized width and height of the detection frame.

(2) Coordinate conversion for spatial overlap assessment.

To facilitate subsequent processing, particularly spatial overlap assessments, which rely on geometric calculations based on pixel coordinates, the normalized YOLO-format coordinates $\left\{class, x_{\text{center}}, y_{\text{center}}, w_f, h_f\right\}$ must first be converted into the pixel-based bounding box format $\{x_{min}, y_{min}, x_{max}, y_{max}\}$. The conversion Equations are provided in Equations (15) to (18):

$$x_{\min} = (x_{center} - \frac{w_f}{2}) \cdot W \tag{15}$$

$$y_{\min} = (y_{center} - \frac{h_f}{2}) \cdot H \tag{16}$$

$$x_{\max} = (x_{center} + \frac{w_f}{2}) \cdot W \tag{17}$$

$$y_{\max} = (y_{center} + \frac{h_f}{2}) \cdot H \tag{18}$$

where W and H are the width and height of the image, respectively.

Step 3: judgment of whether there is an overlapping region between two categories of detection frames.

Based on the axis-wise overlap principle, the presence of an overlapping region between two bounding boxes is determined by evaluating their projections on both coordinate axes independently. If the detection frames overlap in both the horizontal and vertical dimensions, they are considered to have an overlapping region and classified as associated detection frames. Otherwise, they are regarded as non-overlapping. Specifically, given two detection frames, $A = \{x^A_{min}, y^A_{min}, x^A_{max}, y^A_{max}\}$ and $B = \{x^B_{min}, y^B_{min}, x^B_{max}, y^B_{max}\}$, the overlap condition is expressed as follows:

$$x^A_{min} \leq x^B_{max} \text{and } x^A_{max} \geq x^B_{min} \text{and } y^A_{min} \leq y^B_{max} \text{and } y^A_{max} \geq y^B_{min}$$

If the above conditions are satisfied, the two detection frames are considered to have an overlapping region; otherwise, they are regarded as non-overlapping.

Step 4: screening of associated detection frames with result storage

For each root detection frame in root_box, iterate through all the seeding detection frames in turn to judge whether overlapping regions exist. For this root detection frame, if there is an intersection with at least one seeding_box detection frame, the root_box detection frame is kept as a subordinate object; otherwise, it is discarded.

Finally, all the seeding_box detection frames and the screened root_box detection frames are saved together as the final prediction result of the detection frames.

2.5. Algorithm for Crop Row Line Fitting and Navigation Line Extraction

2.5.1. Feature Point Extraction and Clustering

Feature points are extracted from the corn-root detection frames and filtered based on the subordination relationship of the combined detection frames. The coordinates of the feature points, denoted as (x_0, y_0) and (x_0, y_0), are defined according to Equations (19) and (20):

$$x_0 = x_{center} \tag{19}$$

$$y_0 = y_{center} + \frac{H_{height}}{2} \tag{20}$$

where (x_{center}, y_{center}) represents the coordinates of the center point of the root detection frame of the corn plant, and H_{height} represents the height of the detection frame.

Clustering is a data analysis method that categorizes a dataset into distinct classes or clusters based on specific criteria. This paper clusters the extracted feature points with the aim of classifying the feature points located on the same ridge into the same cluster, thereby facilitating the subsequent fitting of the crop row lines. Given the limited number of feature points in the data samples and the relatively uniform data distribution, we have selected the K-means clustering algorithm. K-means is a centroid-based clustering method that iteratively updates cluster centers to minimize the distance between points and the center, enabling effective data grouping.

2.5.2. Crop Row Line Fitting and Navigation Line Extraction

Line fitting is an analytical method that represents the overall trend of data points through an optimal straight line. In this paper, line fitting is applied to the clustered feature points to derive the crop row line. Given the limited number of feature points and the high demands for real-time processing, this study employs the least squares method to fit the feature points in a straight line. The fundamental idea of the least squares method is to identify the straight line that is closest to all observation data by minimizing the sum of the squares of the errors between the fitted model and the observation data. Specifically, for n sets of observation point data (x_i,y_i), the objective of the least squares method is to minimize the objective function, as shown in Equation (21):

$$S = \sum_{i=1}^{n} (y_i - (kx_i + b))^2 \tag{21}$$

where $(y_i - (kx_i + b))^2$ denotes the squared perpendicular distance between the i_{th} observation point and the fitted straight line, and the best-fit straight line is obtained by constantly adjusting the parameters k and b to minimize S. Here, k and b are the slope and intercept of the best-fit line, respectively.

The clustered feature points on the left and right sides were separately fitted using the least squares method, yielding the expressions for the left and right crop rows, as shown in Equations (22) and (23):

$$y_1 = k_1 x + b_1 \tag{22}$$

$$y_2 = k_2 x + b_2 \tag{23}$$

where y_1 denotes the left crop row line, with k_1 and b_1 as its slope and intercept; y_2 denotes the right crop row line, with k_2 and b_2 as its slope and intercept.

After fitting the crop row lines on both the left and right sides, the intersection points of the fitted lines with the top of the image are denoted as L1 and R1, respectively, while the intersection points with the bottom of the image are labeled L2 and R2. The midpoint between L1 and R1 is designated as C1, and the midpoint between L2 and R2 is labeled as C2, as illustrated in Figure 11. Connecting C1 and C2 creates the central navigation line of the corn crop row.

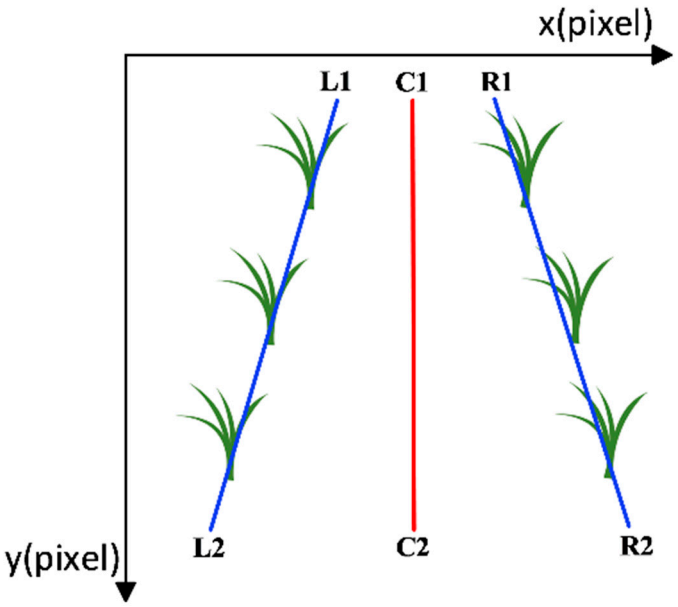


Figure 11. Extracting the center navigation line of a crop row. The coordinate origin is located at the upper-left corner of the image; x increases to the right and y increases downward. Blue lines represent crop row lines, and the red line represents the fitted navigation line.

2.6. Model Evaluation Criteria

To intuitively demonstrate the advantages of the model proposed, this paper employs *precision* (*P*), *recall* (*R*), mean average precision (*mAP*), model weight size (weight size), and the number of parameters (parameters) as evaluation metrics [35].

P is a metric used to evaluate the proportion of correctly predicted positive samples using a model, as shown in Equation (24).

$$Precision = \frac{TP}{TP + FP} \tag{24}$$

TP (true positive) refers to the number of true positive samples, which is the count of correctly predicted targets. FP (false positive) refers to the number of false positive samples, which indicates the count of incorrectly predicted targets.

R is the proportion of all true positive samples that can be identified by the evaluation model, as illustrated in Equation (25).

$$Recall = \frac{TP}{TP + FN} \tag{25}$$

FN (false negative) indicates the number of missed detection samples.

Average precision (*AP*) assesses the model's performance across each category by comprehensively considering both *P* and *R* metrics. The *AP* value is defined as the area between the *precision–recall* curve and its axes, as illustrated in Equation (26).

$$AP = \int_0^1 P(R)dR \tag{26}$$

mAP can evaluate the precision and recall of the model under different thresholds, which is applicable to the performance measurement of multi-category scenarios and is calculated as shown in Equation (27). *mAP* requires computing the *AP* for each category in

the detection task and then averaging the results, with AP_i denoting the AP value for the i_{th} category and N representing the total number of categories.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{27}$$

mAP_{50} indicates the mean precision value at a 50% IoU threshold.

2.7. Parameter Setting and Model Training

This experiment used a 64-bit Windows 11 operating system with a processor model using an Intel (R) Core (TM) i7-14700KF 3.40 GHz and 64.0 GB of RAM. The graphic card model was an NVIDIA GeForce RTX 4090 D. The experiment was accelerated using CUDA, with CUDA version 12.6. The software platform utilized for computer image processing in the experiments was PyCharm version 2024.1.6, with Python version 3.9. The specific training parameters are shown in Table 2.

Table 2. Model training parameters.

Parameter	Value
Epochs	200
Batch size	32
Image size	640 × 640
Optimizer algorithm	SGD
Learning rate	0.01
Momentum	0.937
Weight decay	0.0005

3. Results and Discussion

3.1. Analysis of Ablation Experiments

To comprehensively evaluate the impact of the introduced improvements on model detection performance, this study is based on YOLOv8n and designs ablation experiments that gradually add modules to quantitatively analyze the model's performance before and after the improvements. As shown in Table 3, the model performance was evaluated using the P, R, mAP_{50}, and weight size parameters and the reasoning time as evaluation indicators.

Table 3. Results of model performance comparison in terms of detection performance. The √ symbol indicates that the module is included in the model, while the × symbol indicates its exclusion.

Model						Weight Size (M)	Parameters (10^6)	Reasoning Time (ms)	Root		
Micro-Target Detection Head	DBA	C2f-SCAA	GAM	PIoU2	LAMP				P (%)	R (%)	mAP_{50} (%)
×	×	×	×	×	×	6.0	3.0	147.5	88.4	68.1	79.3
√	×	×	×	×	×	6.0	2.9	173.8	90.5	83.5	89.3
√	√	×	×	×	×	5.7	2.7	221.2	93.2	81.6	90.2
√	√	√	×	×	×	6.0	2.9	242.2	92.1	83.2	90.0
√	√	√	√	×	×	6.2	3.0	231.7	93.4	84.1	90.9
√	√	√	√	√	×	5.9	3.0	226.5	93.4	84.9	92.0
√	√	√	√	√	√	1.9	0.7	63.2	92.6	84.3	91.1

Based on the data presented in the table above, it is evident that the addition of the micro-target detection head module to the original model resulted in a 0.9% increase in the

p value for root detection, a 15.4% increase in the R value, and a 10% increase in the mAP_{50} value, at a computational cost of 26.3 ms. The micro-target detection head module significantly enhanced the model's robustness and feature expression capabilities for micro-target root detection by introducing high-resolution feature layers and facilitating multi-scale fusion, thereby improving detection accuracy and recall rate, which laid a foundation for subsequent optimization. Building upon this, the integration of the DBA resulted in a 2.7% improvement in the root's *P* and a 0.9% increase in mAP_{50}, though it introduced the largest single latency increase of 47.4 ms. This suggests that the DBA has improved positioning accuracy. However, due to stringent target screening and reduced sensitivity to low-confidence targets, some targets went undetected, resulting in a decrease in the recall rate. The addition of C2f-SCAA further increased the R value for root detection by 1.6%, although the *p* value decreased by 1.1%, with the mAP_{50} value remaining relatively unchanged. C2f-SCAA improved multi-scale feature extraction and channel interaction, thereby enhancing robustness. Nonetheless, the weight allocation that emphasized significant features diminished the positioning accuracy of certain targets. Upon further addition of the optimized GAM attention mechanism, each metric of the root has seen an uplift. The GAM module comprehensively improves the detection performance of roots by enhancing the feature-capturing ability and robustness for micro-targets. On this basis, after using the PIoU2 loss function to replace CIoU, the mAP_{50} value for root increased by 1.1%, and other metrics remained largely unchanged. PIoU2 enhances the optimization of medium-quality prediction frames through a non-monotonic attention mechanism, which improves the robustness and detection accuracy of the model in complex environments. After incorporating the LAMP pruning algorithm, the model's detection performance on roots experienced a slight decrease. However, this module played a decisive role in enhancing efficiency, sharply reducing the model's inference time from 226.5 ms to 63.2 ms. Furthermore, the model's weight size and parameters were reduced by 4.1 M and 2.3 × 106, respectively. In comparison to the original YOLOv8n model, the detection performance on roots was significantly enhanced.

Overall, compared to the original YOLOv8n model, the proposed RS-LineNet model demonstrates a significant improvement in root detection performance. Specifically, the *p* value has increased by 4.2%, the R value has risen by 16.2%, and the mAP_{50} value has grown by 11.8%. These enhancements have notably increased the model's detection accuracy for roots and established a solid foundation for extracting feature points based on detection frames and conducting straight-line fitting in subsequent analyses. Furthermore, the weight size and parameters of the proposed model are reduced to 32% and 23% of those of the original model, respectively. And our ablation study further confirmed its advantage in reasoning speed. Through systematic analysis of each component, especially the introduction of the LAMP module, the inference time of the final model was optimized from 147.5 ms at the baseline to 63.2 ms, demonstrating outstanding computational efficiency. Therefore, the RS-LineNet model has successfully improved the accuracy of root detection while significantly achieving model lightweight and inference acceleration, proving that it has high deployment value in practical application scenarios.

3.2. Hyperparameter Sensitivity Analysis

To further validate the robustness and competitive advantage of our proposed RS-LineNet, and to ensure that its superior performance is not dependent on a specific hyperparameter configuration, we conducted a sensitivity analysis on the learning rate. In this experiment, we compared the performance of RS-LineNet against the baseline YOLOv8n model under three different learning rates: 0.01, 0.001, and 0.0001. The detailed results are presented in Table 4.

Table 4. Performance comparison of RS-LineNet and the baseline YOLOv8n model under different learning rates (lr).

lr	Model	*P* (%)	*R* (%)	mAP_{50} (%)
0.01	YOLOv8n	88.4	68.1	79.3
	RS-LineNet	92.6	84.3	91.1
0.001	YOLOv8n	87.2	66.5	77.9
	RS-LineNet	91.8	82.5	90.1
0.0001	YOLOv8n	85.1	63.2	75.4
	RS-LineNet	90.3	80.1	88.5

To further validate the robustness and competitive advantage of our proposed model, we conducted a hyperparameter sensitivity analysis by comparing the performance of RS-LineNet and the baseline YOLOv8n model under different learning rates. As shown in Table 4, RS-LineNet consistently and significantly outperforms the baseline YOLOv8n model across all tested learning rates.

While the performance of both models varies with the learning rates, with the optimal results achieved at lr = 0.01, RS-LineNet maintains a substantial performance margin in all scenarios. For instance, even at its lowest performing setting (lr = 0.0001), RS-LineNet's mAP_{50} of 88.5% is still considerably higher than the baseline model's best performance of 79.3%. This result demonstrates that the superiority of RS-LineNet is not dependent on a specific hyperparameter configuration and confirms the effectiveness of our proposed architectural improvements.

3.3. Comparative Analysis of Multi-Model Performance

To validate the advantages of the proposed RS-LineNet model proposed in this paper for detecting the micro-targets, the root of the corn plant, we conducted comparative experiments between the proposed model and other deep learning networks known for their strong performance in detection tasks. The evaluation indicators include the *P*, *R*, mAP_{50}, and weight size parameters.

As shown in Table 5, the weight sizes of YOLOv7, SSD, and Faster-RCNN are 11.7 M, 28.5 M, and 523.6 M, respectively. These models are excessively large, making it challenging to meet the deployment requirements for resource-constrained devices. In contrast, YOLOv5, as a lightweight model, has a weight size of 3.7 M, offering significant deployment advantages. However, despite YOLOv5 demonstrating a certain level of competitiveness in terms of weight size, its performance in detecting the two types of targets is relatively poor. Specifically, the average *p* value of YOLOv5 for two types of target detection is 88.7%, the average R value is 79%, and the average mAP_{50} value is 81.6%. These metrics are lower than the improved RS-LineNet model by 5.6%, 11.1%, and 13%, respectively, which suggests that YOLOv5 encounters significant deficiencies in extracting valid information from the seeding and root detection frames. Furthermore, the weight sizes of YOLOv8, YOLOv9, and YOLOv10 are 6.0 M, 5.8 M, and 5.5 M, respectively. Although these models are slightly larger than RS-LineNet, they remain within a reasonable deployable range overall. The performance of these three models in the seeding target detection task is comparable to that of RS-LineNet, with fluctuations in each performance index within 2.3%, and generally exceeding 95%. However, the performance of these three models in the root target detection task is significantly inferior to that of RS-LineNet, especially in terms of the R value, which, to some extent, reflects the model's missed detection effectiveness, with a higher R value representing fewer missed detections. The R values for root detection are 16.2%, 17.4%, and 18.4% lower for YOLOv8, YOLOv9, and YOLOv10, respectively, compared to RS-LineNet, suggesting a higher likelihood of missing detections in these models. Similarly, the three lightweight models, Starnet, HGNetV2, and EfficientViT, exhibit comparable

results. While their performance in seeding detection remains high, generally above 95%, their root detection performance is notably lower. In particular, their R values are 19.0%, 18.4%, and 19.8% lower than RS-LineNet, respectively. This indicates that these lightweight models face a higher risk of missed root detections, which can affect subsequent feature point extraction and crop row line fitting accuracy. Given that the subsequent feature point extraction and crop row line fitting accuracy in this paper heavily depend on the model's root detection accuracy, RS-LineNet demonstrates significant advantages in this context.

Table 5. Comparative analysis of multi-model performance.

Model	Weight Size (M)	Parameters (10^6)	All			Seeding			Root		
			P (%)	*R* (%)	mAP_{50} (%)	*P* (%)	*R* (%)	mAP_{50} (%)	*P* (%)	*R* (%)	mAP_{50} (%)
YOLOv5	3.7	1.8	88.7	79.0	81.6	94.2	88.1	91.3	83.1	69.9	71.9
YOLOv7	11.7	6.0	93.0	83.9	87.4	96.1	89.9	92.9	90.0	77.8	82.0
YOLOv8	6.0	3.0	92.2	82.3	88.9	96.1	96.5	98.5	88.4	68.1	79.3
YOLOv9	5.8	2.6	90.0	81.4	87.9	93.8	95.9	98.0	86.1	66.9	77.8
YOLOv10	5.5	3.0	91.3	81.0	87.3	96.4	96.1	98.2	86.1	65.9	76.3
SSD	28.5	23.7	46.2	24.3	30.5	77.7	47.9	59.7	14.7	0.7	1.3
Faster-RCNN	523.6	136.7	35.4	42.4	41.2	64.8	83.3	82.2	6.0	1.5	0.2
Starnet	4.46	2.25	90.5	80.6	87.3	94.3	95.9	98.1	86.8	65.3	76.6
HGNetV2	4.77	2.35	90.9	81.2	87.8	95	96.5	98.2	86.7	65.9	77.3
EfficientViT	8.37	4	91.2	80.2	87.1	95.1	95.9	98	87.2	64.5	76.2
RS-LineNet	1.9	0.7	94.3	90.1	94.6	96.1	95.8	98.2	92.6	84.3	91.1

In summary, while other commonly used detection models possess certain advantages in some respects, RS-LineNet demonstrates superior performance in particular tasks, especially in the detection accuracy of root targets. Considering the overall performance, weight size, and parameters of the model, RS-LineNet, with its smaller weight size and fewer parameters, can ensure high detection accuracy while effectively meeting the practical application needs of agricultural target detection. Therefore, RS-LineNet represents the best overall performance and is the ideal choice.

3.4. Analysis of Visualization Results for Corn Plant Root Detection

To evaluate the performance improvement of the proposed RS-LineNet model in root micro-target detection, Figure 12 presents a comparison of prediction results across different models, including YOLOv7, YOLOv8n, Starnet, and RS-LineNet, on the same images. The red arrows in Figure 12b–d,g–i highlight the missed detections that occur when using YOLOv7, YOLOv8n, and Starnet, respectively, under complex backgrounds, whereas Figure 12e,j demonstrate that RS-LineNet successfully identifies these missed roots, effectively reducing detection errors and significantly improving detection performance.

This performance improvement results from a series of systematic optimizations to the original YOLOv8n architecture. Specifically, the introduction of a micro-target detection head significantly enhances the model's fundamental capability to detect micro-target, leading to a notable improvement in detection accuracy and establishing a solid foundation for subsequent modules. The DBA module improves localization accuracy, resulting in a further increase in the *p* value. The C2f-SCAA module enhances multi-scale feature extraction and inter-channel interaction, effectively improving the R value. The optimized GAM attention mechanism reinforces the model's focus on critical regions, thereby improving overall detection robustness and contributing to greater detection accuracy. In addition, replacing the CIoU loss function

with the PIoU2 loss function effectively refines the regression of medium-quality bounding boxes, further promoting the enhancement of detection accuracy.

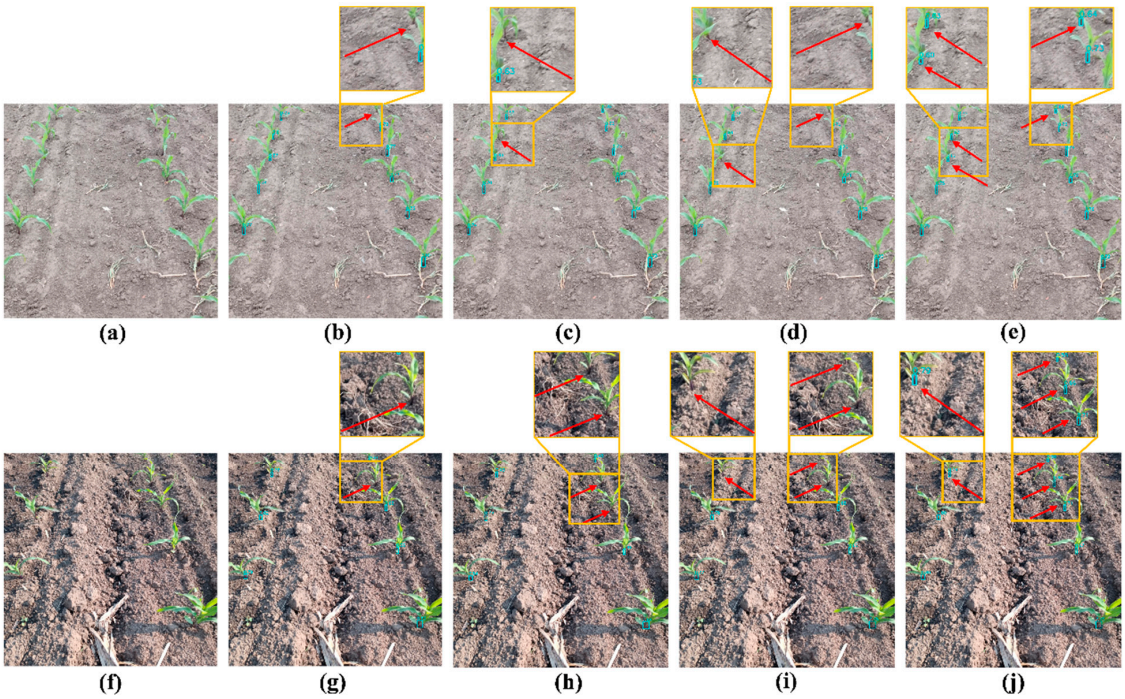


Figure 12. Comparative analysis of the actual detection performance for roots before and after model optimization. (**a**,**f**) Original image; (**b**,**g**) detection results of roots using the YOLOv7 model, with the red arrow indicating the missed detections that occur when using YOLOv7 for prediction; (**c**,**h**) detection results of roots using the YOLOv8n model, with the red arrow indicating the missed detections that occur when using YOLOv8n for prediction; (**d**,**i**) detection results of roots using the Starnet model, with the red arrow indicating the missed detections that occur when using Starnet for prediction; (**e**,**j**) detection results of roots using the RS-LineNet model, with the red arrows indicating the roots that were missed in (**b**–**d**) and (**g**–**i**) but that can be successfully identified with RS-LineNet.

In summary, RS-LineNet has comprehensively enhanced the detection accuracy of the model for the micro-target of root through multi-module collaborative optimization, making its detection effect significantly better than that of the other comparative models.

To verify the potential advantage of the RS-LineNet model proposed in this paper over the YOLOv8n model in terms of convergence speed during the training process, a comparative analysis of the bounding box loss function of the two models was conducted, and the results are shown in Figure 13. box_loss is an important measure of the discrepancy between predicted and real frames, which is used to evaluate the model's localization accuracy. From Figure 13, both models exhibit the property of stabilizing the loss value in the late stage of training. Compared with the YOLOv8n model, the RS-LineNet model proposed in this paper shows a faster convergence trend after about 20 training rounds, the loss value decreases more rapidly, and the overall loss level is lower in the subsequent training process, and the loss value at the final stabilization is slightly better than that of the YOLOv8n, which suggests that the RS-LineNet achieves a certain advantage in terms of the convergence speed and localization accuracy.

To verify that the proposed RS-LineNet model exhibits superior detection performance for micro-targets compared to the original YOLOv8n model, this paper visualizes the output results of the minimum detection layer from both models on the heat maps. In the heat maps, the greater the model's attention to the target, the closer its color is to warmer hues. As illustrated in Figure 14, row (a) displays the original image, row (b) presents

the output results from the smallest size detection head of the YOLOv8n model, and row (c) showcases the output results from the smallest size detection head of the proposed RS-LineNet model. An observation of Figure 14 reveals that the proposed RS-LineNet model exhibits significantly higher sensitivity to the root location compared to the original YOLOv8n network, thereby intuitively demonstrating the advantage of the proposed RS-LineNet model in detecting these micro-targets, the roots of the corn plant.

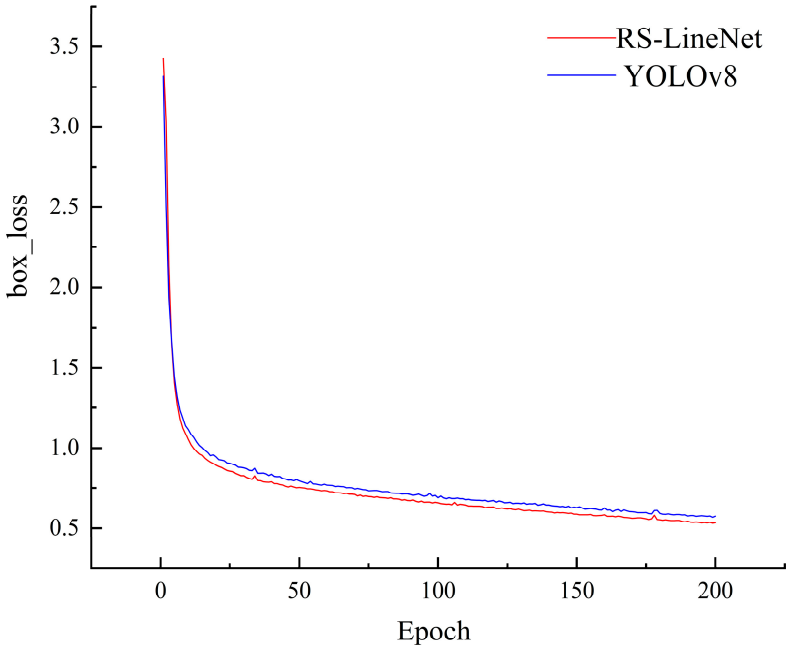


Figure 13. Comparative analysis of bounding box loss (box_loss) before and after model optimization during the training process.

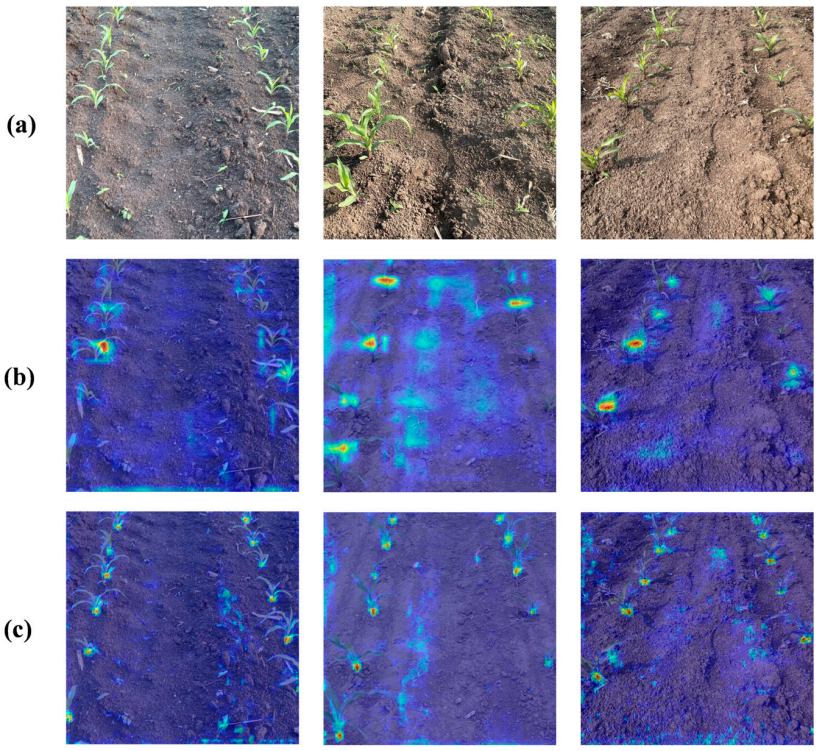


Figure 14. Comparative analysis of the effect of heat maps before and after model optimization. Row (**a**) shows the original image; row (**b**) shows the heat map visualization results for YOLOv8n; row (**c**) shows the heat map visualization results for RS-LineNet.

3.5. Effectiveness Evaluation of the Subordination Relationship Filtering Algorithm in Root Misdetection Suppression

To verify the effectiveness of the filtering algorithm proposed in this paper, based on the subordination relationship of the combined detection frame for filtering out isolated root misdetection frames, this paper conducted a comparative experiment on whether to use this algorithm after using the model to perform prediction, and the results are shown in Figure 15.

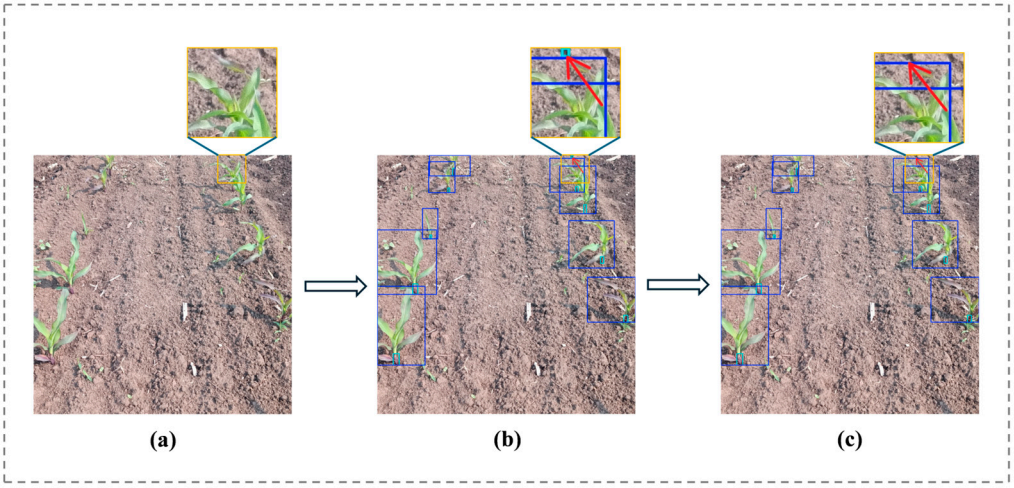


Figure 15. Comparative impact of subordination relationship filtering algorithm on model performance. (**a**) Original image; (**b**) RS-LineNet prediction results; (**c**) application of the subordination relationship filtering algorithm based on RS-LineNet prediction results. The deep-blue boxes represent seeding detection frames, while the light-blue boxes represent root detection frames. The arrow highlights an isolated root misdetection frame in the upper right corner, which is effectively removed after filtering.

In Figure 15, three images are presented separately. Figure 15a displays the original image, Figure 15b illustrates the detection map generated via the model's direct predictions, and Figure 15c presents the detection map after applying the subordination relationship filtering algorithm to the model's predictions. By comparing Figure 15b,c, it can be observed that there is an isolated root misdetection frame in the upper right corner of Figure 15b. However, in Figure 15c, after applying filtering based on subordinate relationships, this misdetection frame is successfully removed. This comparison intuitively highlights the advantages of the filtering algorithm proposed in this paper in suppressing isolated root misdetection frames. To further validate the effectiveness of this algorithm, the test set images were predicted using the RS-LineNet network, followed by filtering the root detection frames through the subordination relationship filtering algorithm, and recalculating the p value. The results demonstrated that, after the filtering process, the p value of the root detection frame increased from 92.6% to 93.4%, indicating an improvement of 0.8%. This result fully demonstrates that the subordination relationship filtering algorithm proposed in this paper can effectively improve the detection accuracy of root detection frames, thereby further enhancing the accuracy and reliability based on the model's detection results.

In summary, the subordination relationship filtering algorithm based on combined detection frames proposed in this paper provides a completely new idea for addressing the issue of misdetections that target detection models are prone to in complex scenarios by effectively filtering out isolated misdetection frames. In the field of agriculture, the algorithm can provide accurate support for the task of crop row line extraction in complex farmland environments, especially in the face of challenges such as dense plants and

complex environments, which can significantly improve the precision and stability of the recognition of key parts. The application of this technology is expected to promote the development of agricultural research in a more intelligent and refined direction.

3.6. Comparison of Angular Deviation of Crop Row Line Fitting Based on Canopy and Root Feature Points

In order to verify that the use of the root detection frame to extract feature points for crop line fitting involves a smaller angle deviation compared with the use of the plant canopy detection frame to extract feature points for crop line fitting, this paper conducted a comparative experiment using these two different feature point extraction methods, and the experimental results are shown in Figure 16.

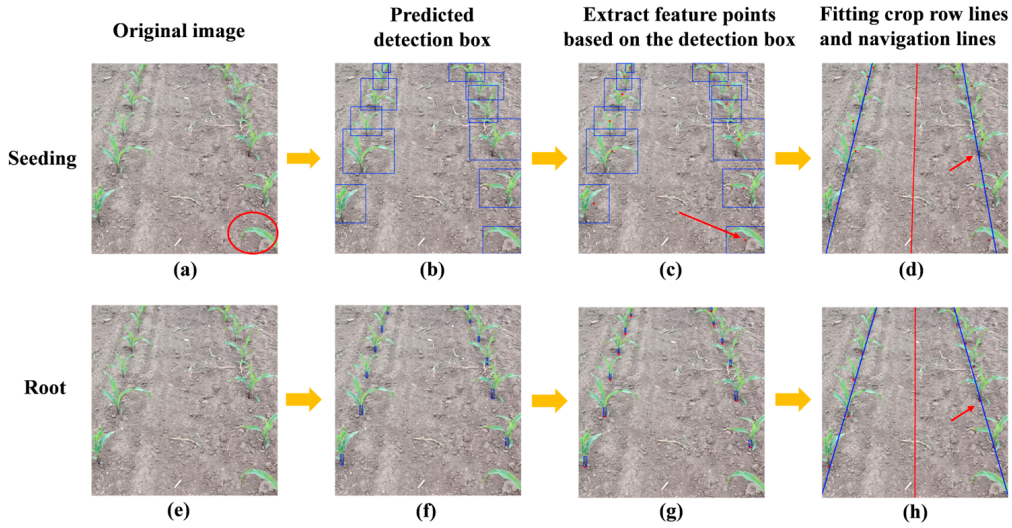


Figure 16. Comparison of straight-line fitting effect of extracted feature points of canopy detection frame and root detection frame. (**a**) Original image; (**b**) prediction results of the canopy detection frame; (**c**) extracted feature points based on the canopy detection frame; (**d**) fitting crop row lines and navigation lines based on the canopy feature points; (**e**) original image; (**f**) prediction results of the root detection frame; (**g**) extraction of feature points based on the root detection frame; (**h**) fitting crop row lines and navigation lines based on root feature points. In (**b**) and (**c**), the blue boxes represent seeding detection frames, while in (**f**) and (**g**), the blue boxes represent root detection frames. The red circles highlight regions where the roots are not captured within the capturing range, leading to deviations when using canopy-based feature points. The red arrows indicate interference feature points and the severely deviated fitted crop row lines caused by these points, whereas the root-based approach avoids such deviations.

As shown in Figure 16, the first row illustrates the crop row line fitting effect achieved by extracting feature points using the corn canopy detection frame, while the second row demonstrates the crop row line fitting effect obtained through the extraction of feature points using the corn plant root detection frame. Figure 16a,e present the same original images, whereas Figure 16b,f present the model's detection effects for seeding and root, respectively. The red feature points within the detection frames in Figure 16c,g are extracted from the center point of the seeding detection frame and the center point at the bottom of the root detection frame, respectively. In Figure 16d,h, the blue lines represent the crop row lines fitted based on the feature points, while the red lines indicate the navigation lines extracted from the crop row lines.

Observing the red-circled marked areas in Figure 16a, it can be seen that only a portion of the plants are captured, and the roots of the plants are not included within the shooting range. In this case, the method using canopy feature points for fitting will

still predict the plant and adopt the center point of the canopy detection frame as the feature point (as the position pointed to by the red arrow in Figure 16c), this approach will result in a large deviation between the extracted feature points and the actual plant feature points, introducing interference feature points and leading to a severe deviation in the fitted crop row lines (as indicated by the fitted crop row line pointed to by the red arrow in Figure 16d). In contrast, the method that uses root feature points for fitting will not produce a predicted root detection frame in this case, thereby avoiding the extraction of misleading feature points in edge regions, significantly improving the fitting accuracy of the crop row lines (as indicated by the fitted crop row lines pointed to by the red arrows in Figure 16h).

This paper proposes using the bottom center point of the root detection frame as the feature point. This approach offers the advantage of being stable in position and exhibiting strong regularity, which can effectively mitigate the issue of canopy feature point deviation caused by natural environmental disturbances such as wind. In addition, when the edge of the image shows incomplete plants, the extraction of the root feature points can avoid the feature point extraction bias problem caused by the canopy feature points due to capturing only part of the plant and significantly improve the feature point extraction and fitting accuracy. The experimental results indicate that the average angle error of the navigation line based on canopy feature point extraction is 3.9°, whereas the average angle error of the navigation line based on root feature point extraction is only 0.8°. This demonstrates that root feature points can extract the navigation line more accurately and reduce angle error, thereby providing an efficient and reliable solution for automated agricultural management.

3.7. Algorithm Robustness Experiment

To verify the adaptability of the algorithm proposed in this paper across different natural environments, we selected four typical growing environments to test its fitting accuracy. We evaluated the effectiveness of the algorithm by comparing the angular error between the manually annotated navigation line and the algorithm-fitted navigation line. Figure 17 illustrates the fitting results of navigation lines extraction using the method proposed in this paper under various growth environments. The blue lines in the figure represent the crop row lines fitted based on the root feature points, the red lines indicate the navigation lines extracted from the crop row lines, and the black lines denote the manually marked navigation lines.

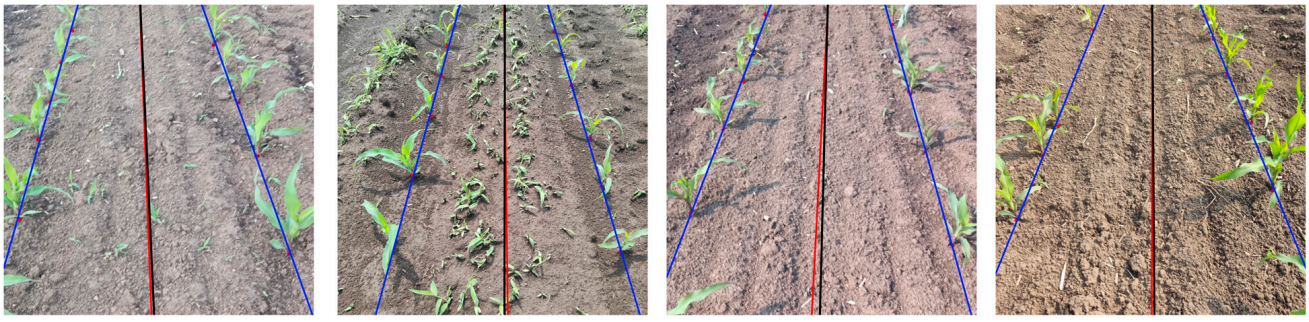


Figure 17. Analysis of the effect of crop row line fitting under different growing environments. (**a**) Normal growth; (**b**) weed symbiosis; (**c**) adhesive growth; (**d**) seedling-missing growth. The blue lines represent the crop row lines fitted based on the root feature points, the red lines indicate the navigation lines extracted from the crop row lines, and the black lines denote the manually marked navigation lines.

Table 6 presents the average angular error between the fitted navigation line and the manually marked navigation line of this algorithm across various growth environments. The data in Table 6 indicates that the algorithm in this paper demonstrates high fitting accuracy in different growth conditions. In the normal growth environment, the average angular error of the fitted algorithm is merely 0.32°, reflecting excellent performance. In environments characterized by weed symbiosis and missing seedlings, the average angular error of the fitted line increases due to factors such as weed occlusion and the absence of plants. Nevertheless, the error under these conditions remains low. In the adhesion growth environment, the average angular error of the fitting reaches its maximum value, primarily because adhesion leads to the generation of multiple root detection frames with similar intervals, thus introducing densely and irregularly distributed adjacent feature points. These feature points significantly interfere with the linear fitting process, resulting in an increased angular error. However, even in the most complex adhesion growth environment, the average fitting angular error is maintained within 3°. This performance is well within the acceptable limits for agricultural navigation, where prior works such as those of Gong et al. [25] have reported errors of up to 5°. This demonstrates the robustness of our method, providing a reliable technical foundation for practical automated navigation.

Table 6. Comparison of average angular error of crop row line fitting in different growth environments.

Growth Environments	Average Angular Error (°)
Normal growth	0.32
Weed symbiosis	0.71
Adhesive growth	0.54
Seedling-missing growth	1.63

In summary, this paper significantly improves the accuracy of feature point extraction and the precision of the fitted line by improving the model to improve the detection accuracy of the root detection frame and reducing the misdetection rate of the root detection frame through the filtering algorithm. Under four typical farmland environments, the algorithms are all capable of effectively dealing with the interference brought by the complex scenes and ensuring that the fitting error is controlled within an acceptable range, with strong robustness, which provides a reliable guarantee for the precise navigation and automated operation of smart agricultural machines.

3.8. Ablation Study on Key Innovations

To precisely quantify the individual contributions of the two main innovations in our work—the shift to root-based detection and the application of the subordination filtering algorithm—we conducted a dedicated ablation study. As presented in Table 7, we compared our full method against a baseline canopy-based extraction method and an intermediate version of our method using root-based detection without the filtering step.

Table 7. Ablation study on the contributions of root-based detection and subordination filtering.

Method	Target	P (%)	Average Angular Error (°)
Canopy-based extraction	Seeding	96.1	3.9
Root-based only, no filtering	Root	92.6	1.2
Full method, with filtering	Root	93.4	0.8

By comparing the traditional navigation line extraction method based on canopy feature points with the proposed navigation line extraction method based on root feature

points, the intuitive advantages of the latter can be directly observed. When using the proposed RS-LineNet network to perform prediction on both seeding and root targets, the detection precision for roots is 3.5 percent lower than that for seeding. However, the average angular error of the navigation line based on root feature point extraction is 2.7 degrees lower than that based on canopy feature point extraction. This result indicates that, in navigation line extraction tasks, the stability of feature point positions, such as those provided by roots, is more critical than merely pursuing detection precision, as in the case of canopies. Furthermore, by incorporating the subordination relationship filtering algorithm for post-processing, the detection precision of root-based detection is further increased to 93.4 percent, while the average angular error of the navigation line based on root feature point extraction is ultimately optimized to 0.8 degrees.

These results demonstrate that the proposed navigation line extraction method achieves a substantial improvement in both accuracy and reliability compared with the traditional method based on canopy feature points. By shifting the feature extraction target from canopy to root, the method significantly enhances the stability of feature points, which is crucial for precise navigation line estimation. Moreover, the introduction of the subordination relationship-filtering algorithm further improves the precision of root feature points with stable positions, effectively reducing angular errors. Overall, the proposed approach not only strengthens the robustness of feature point extraction but also improves the reliability and stability of navigation line extraction.

3.9. Embedded Deployment Experiment

In response to the demand for real-time extraction of agricultural navigation lines, this study employed a pruning technique to reduce the weight size and parameters. The navigation line extraction algorithm proposed in this paper was deployed and validated on a Jetson TX2 edge device. The experimental platform is equipped with an ARMv8 Cortex-A57 central processor and an NVIDIA Tegra X2 architecture GPU, and it operates on the Ubuntu 18.04.6 LTS operating system. The software framework was built using Python 3.9 and PyTorch 2.1.2. To thoroughly evaluate the real-time performance, a detailed latency breakdown of the entire processing pipeline was conducted on Jetson TX2, separating the contributions of model detection, subordination relationship filtering, and line fitting. As illustrated in Figure 18, the analysis confirms that the RS-LineNet model detection is the most computationally intensive stage, responsible for most of the total processing latency. In contrast, the subsequent subordination relationship filtering and the final line fitting stages are computationally efficient, adding only minimal overhead to the total processing time. This efficient structure ensures that the entire system demonstrates relatively accurate extraction accuracy, with a consistently maintained frame rate above 12 FPS.

In summary, the navigation line extraction algorithm proposed in this paper operates stably on resource-constrained edge devices, with the corresponding latency remaining within an acceptable range. This provides a solid foundation for the future deployment of the algorithm in agricultural equipment.

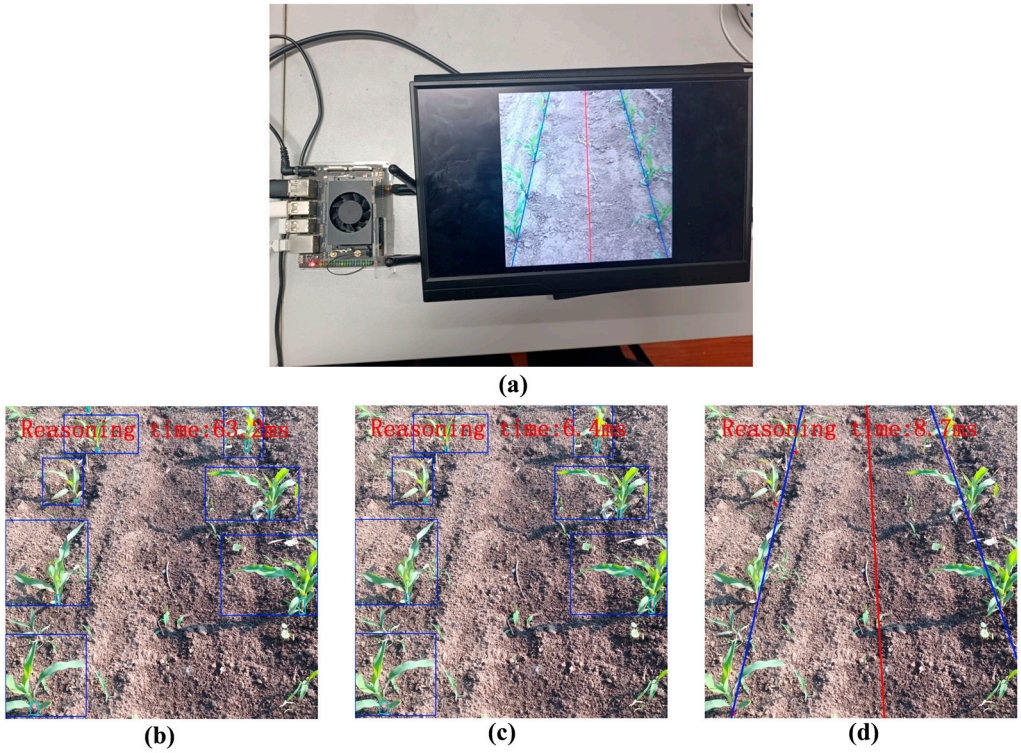


Figure 18. Navigational line extraction algorithm edge deployment measured results. (**a**) Navigation line extraction results based on Jetson TX2; (**b**) RS-LineNet prediction stage results, where the dark blue boxes represent the seeding detection frames and the light blue boxes represent the root detection frames; (**c**) the stage results of the subordinate relationship filtering algorithm, where the dark blue boxes represent the seeding detection frames and the light blue boxes represent the root detection frames; (**d**) the result of the navigation line extraction stage, where the red and dark red point sets represent the feature points on different ridges, the blue lines represent the crop row lines, and the red line represents the navigation line.

4. Conclusions

To address the problem in visual navigation where plant canopy detection frames are easily affected by environmental interference, leading to large deviations in navigation line extraction, this study proposed an algorithm to optimize the root detection frame to extract feature points and achieve navigation fitting.

The main research work is as follows:

1. This study established a corn crop row dataset that encompasses multiple growing environments, including conditions such as normal growth, weed symbiosis, adherent growth, and seedling-missing growth, thereby enhancing the applicability and robustness of the algorithm.
2. This research is based on the YOLOv8n model. By incorporating a micro-target detection head module, introducing the DBA based on the SBA, proposing the C2f_SCAA module, optimizing the GAM attention mechanism, utilizing the PIoU2 loss function in place of the CIoU loss function, and implementing the LAMP pruning algorithm, we constructed the RS-LineNet model. The proposed RS-LineNet network achieves model lightweighting while enhancing the detection accuracy of corn plant roots. Compared to the YOLOv8n model, the precision of root detection through RS-LineNet improved by 4.2%, the recall increased by 16.2%, and the mean average precision rose by 11.8%. Furthermore, the model's weight size is only 32% of that of the YOLOv8n model, and the number of parameters was reduced to 23% of the YOLOv8n model. Compared with lightweight YOLO variants such as YOLOv5, YOLOv9, and YOLOv10,

as well as lightweight detection models including Starnet, HGNetV2, and EfficientViT, RS-LineNet exhibits distinct superiority in the task of corn-root detection. While these models demonstrate good detection performance for seeding targets, they exhibit significant limitations in detecting the micro-target of corn roots, particularly with respect to the R value, which, to some extent, reflects the model's missed detection effectiveness, with a higher R value representing fewer missed detections. Given that the subsequent feature point extraction and crop row line fitting heavily depend on accurate root detection, RS-LineNet shows clear advantages in this context. Moreover, RS-LineNet not only achieves markedly higher root detection performance compared with these models but also maintains the lowest parameter count and model weight, making it the most suitable and optimal choice for this task.

3. This research innovatively proposes a processing algorithm for filtering root detection frames based on the subordinate relationships among combined detection frames. The algorithm leverages the spatial correlation between plant detection frames and root detection frames, effectively identifying and eliminating isolated root misdetection frames that do not conform to actual positions. Following the filtering process based on subordinate relationships, the detection precision of root detection frames is enhanced from 92.6% to 93.4%, thereby further improving the detection accuracy of corn plant roots in complex environments.
4. This research proposes a method for extracting navigation lines based on root feature points. Compared to traditional extraction methods that rely on canopy feature points, this approach effectively mitigates the interference caused by the natural environment on feature point extraction, thereby ensuring the stability and accuracy of the feature points. The verification results indicate that the average angular error of the navigation lines extracted using this method across various growth environments is only 0.8°, a 3.1° reduction over canopy-based methods. Additionally, the frame rate consistently exceeds 12 FPS when implemented on the Jetson TX2 edge device, thereby robustly demonstrating the effectiveness of this algorithm in extracting navigation lines.

This study innovatively utilizes root feature points for crop navigation line fitting, effectively addressing the instability of canopy feature points under environmental interference, particularly the challenges encountered in field deployment. Future research will enhance the model's adaptability to varying lighting and soil conditions through data augmentation and mitigate occlusion effects by incorporating attention mechanisms or dynamic feature weighting. Furthermore, by integrating features from different growth stages of corn and other crops (such as cotton, soybean, sorghum, and fruit trees), and considering the practical conditions in the field, the generalization ability of the method will be further improved, thereby enhancing the overall crop monitoring and management efficiency.

Author Contributions: Conceptualization, Y.X. and Z.L.; methodology, Y.X., Y.Z. (Yang Zhou) and Z.L.; software, J.L. and Y.Z. (Yuting Zhai); validation, Y.Z. (Yuting Zhai), C.L. and X.Z.; formal analysis, Z.L. and C.L.; investigation, J.L. and X.Z.; resources, Y.X. and Y.Z. (Yang Zhou); data curation, Y.Z. (Yang Zhou); writing—original draft preparation, Z.L.; writing—review and editing, Y.X., Y.Z. (Yang Zhou) and Z.L.; visualization, Y.Z. (Yuting Zhai) and C.L.; supervision, Y.X. and Y.Z. (Yang Zhou); project administration, Y.X.; funding acquisition, Y.X. and Y.Z. (Yang Zhou). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Jilin Provincial Department of Science and Technology—Key Research and Development, grant number 20230202035NC.

Data Availability Statement: The source code for the proposed RS-LineNet model and a representative subset of the dataset generated and analyzed during the current study are publicly available

through GitHub 3.17.5 with the DOI: https://github.com/Zhenrrrrr/RS-LineNet (accessed on 24 July 2025). The full dataset is not publicly available due to ongoing research.

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

References

1. Wang, W.; Huang, Z.; Fu, Z.; Jia, L.; Li, Q.; Song, J. Impact of digital technology adoption on technological innovation in grain production. *J. Innov. Knowl.* **2024**, *9*, 100520. [CrossRef]
2. Saha, B.; Biswas, S.; Datta, S.; Mojumdar, A.; Pal, S.; Mohanty, P.S.; Giri, M.K. Sustainable Nano solutions for global food security and biotic stress management. *Plant Nano Biol.* **2024**, *9*, 100090. [CrossRef]
3. Mana, A.A.; Allouhi, A.; Hamrani, A.; Rehman, S.; el Jamaoui, I.; Jayachandran, K. Sustainable AI-based production agriculture: Exploring AI applications and implications in agricultural practices. *Smart Agric. Technol.* **2024**, *7*, 100416. [CrossRef]
4. Sara, G.; Todde, G.; Pinna, D.; Caria, M. Evaluating an autonomous electric robot for real farming applications. *Smart Agric. Technol.* **2024**, *9*, 100595. [CrossRef]
5. Wakchaure, M.; Patle, B.K.; Mahindrakar, A.K. Application of AI techniques and robotics in agriculture: A review. *Artif. Intell. Life Sci.* **2023**, *3*, 100057. [CrossRef]
6. Mandal, S.; Yadav, A.; Panme, F.A.; Devi, K.M.; Kumar, S.M.S. Adaption of smart applications in agriculture to enhance production. *Smart Agric. Technol.* **2024**, *7*, 100431. [CrossRef]
7. Alrowaily, M.A.; Alruwaili, O.; Alghamdi, M.; Alshammeri, M.; Alahmari, M.; Abbas, G. Application of extreme machine learning for smart agricultural robots to reduce manoeuvering adaptability errors. *Alex. Eng. J.* **2024**, *109*, 655–668. [CrossRef]
8. Jiang, S.; Qi, P.; Han, L.; Liu, L.; Li, Y.; Huang, Z.; Liu, Y.; He, X. Navigation system for orchard spraying robot based on 3D LiDAR SLAM with NDT_ICP point cloud registration. *Comput. Electron. Agric.* **2024**, *220*, 108870. [CrossRef]
9. Soori, M.; Arezoo, B.; Dastres, R. Artificial intelligence, machine learning and deep learning in advanced robotics, a review. *Cogn. Robot.* **2023**, *3*, 54–70. [CrossRef]
10. Xie, B.; Jin, Y.; Faheem, M.; Gao, W.; Liu, J.; Jiang, H.; Cai, L.; Li, Y. Research progress of autonomous navigation technology for multi-agricultural scenes. *Comput. Electron. Agric.* **2023**, *211*, 107963. [CrossRef]
11. Yao, Z.; Zhao, C.; Zhang, T. Agricultural machinery automatic navigation technology. *iScience* **2024**, *27*, 108714. [CrossRef] [PubMed]
12. Bai, Y.; Zhang, B.; Xu, N.; Zhou, J.; Shi, J.; Diao, Z. Vision-based navigation and guidance for agricultural autonomous vehicles and robots: A review. *Comput. Electron. Agric.* **2023**, *205*, 107584. [CrossRef]
13. Zhang, X.; Li, X.; Zhang, B.; Zhou, J.; Tian, G.; Xiong, Y.; Gu, B. Automated robust crop-row detection in maize fields based on position clustering algorithm and shortest path method. *Comput. Electron. Agric.* **2018**, *154*, 165–175. [CrossRef]
14. Yang, Z.; Yang, Y.; Li, C.; Zhou, Y.; Zhang, X.; Yu, Y.; Liu, D. Tasseled Crop Rows Detection Based on Micro-Region of Interest and Logarithmic Transformation. *Front. Plant Sci.* **2022**, *13*, 916474. [CrossRef]
15. Zhou, Y.; Yang, Y.; Zhang, B.L.; Wen, X.; Yue, X.; Chen, L.Q. Autonomous detection of crop rows based on adaptive multi-ROI in maize fields. *Int. J. Agric. Biol. Eng.* **2021**, *14*, 217–225. [CrossRef]
16. Zhang, B.; Zhao, D.; Chen, C.; Li, J.; Zhang, W.; Qi, L.; Wang, S. Extraction of Crop Row Navigation Lines for Soybean Seedlings Based on Calculation of Average Pixel Point Coordinates. *Agronomy* **2024**, *14*, 1749. [CrossRef]
17. Meng, C.-B.; Zhang, Z.; Zhang, T.; Jiang, X.-L.; Zhao, C.-Y.; Qiao, S. Comprehensive quality assessment method for neutron radiographic images based on CNN and visual salience. *Nucl. Sci. Tech.* **2025**, *36*, 118. [CrossRef]
18. Wang, S.; Su, D.; Jiang, Y.; Tan, Y.; Qiao, Y.; Yang, S.; Feng, Y.; Hu, N. Fusing vegetation index and ridge segmentation for robust vision based autonomous navigation of agricultural robots in vegetable farms. *Comput. Electron. Agric.* **2023**, *213*, 108235. [CrossRef]
19. Gong, H.; Wang, X.; Zhuang, W. Research on Real-Time Detection of Maize Seedling Navigation Line Based on Improved YOLOv5s Lightweighting Technology. *Agriculture* **2024**, *14*, 124. [CrossRef]
20. Diao, Z.; Guo, P.; Zhang, B.; Zhang, D.; Yan, J.; He, Z.; Zhao, S.; Zhao, C.; Zhang, J. Navigation line extraction algorithm for corn spraying robot based on improved YOLOv8s network. *Comput. Electron. Agric.* **2023**, *212*, 108049. [CrossRef]
21. Ju, J.; Chen, G.; Lv, Z.; Zhao, M.; Sun, L.; Wang, Z.; Wang, J. Design and experiment of an adaptive cruise weeding robot for paddy fields based on improved YOLOv5. *Comput. Electron. Agric.* **2024**, *219*, 108824. [CrossRef]
22. Cao, Z.; Gong, C.; Meng, J.; Liu, L.; Rao, Y.; Hou, W. Orchard Vision Navigation Line Extraction Based on YOLOv8-Trunk Detection. *IEEE Access* **2024**, *12*, 104126–104137. [CrossRef]
23. Liu, T.-H.; Zheng, Y.; Lai, J.-S.; Cheng, Y.-F.; Chen, S.-Y.; Mai, B.-F.; Liu, Y.; Li, J.-Y.; Xue, Z. Extracting visual navigation line between pineapple field rows based on an enhanced YOLOv5. *Comput. Electron. Agric.* **2024**, *217*, 108574. [CrossRef]
24. Diao, Z.; Ma, S.; Zhang, D.; Zhang, J.; Guo, P.; He, Z.; Zhao, S.; Zhang, B. Algorithm for Corn Crop Row Recognition during Different Growth Stages Based on ST-YOLOv8s Network. *Agronomy* **2024**, *14*, 1466. [CrossRef]

25. Gong, J.; Wang, X.; Zhang, Y.; Lan, Y.; Mostafa, K. Navigation line extraction based on root and stalk composite locating points. *Comput. Electr. Eng.* **2021**, *92*, 107115. [CrossRef]
26. Zheng, Z.; Hu, Y.; Li, X.; Huang, Y. Autonomous navigation method of jujube catch-and-shake harvesting robot based on convolutional neural networks. *Comput. Electron. Agric.* **2023**, *215*, 108469. [CrossRef]
27. Hu, H.; Liu, J.; Zhang, X.; Fang, M. An Effective and Adaptable K-means Algorithm for Big Data Cluster Analysis. *Pattern Recognit.* **2023**, *139*, 109404. [CrossRef]
28. Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. Enhancing geometric factors in model learning and inference for object detection and instance segmentation. *IEEE Trans. Cybern.* **2021**, *52*, 8574–8586. [CrossRef] [PubMed]
29. Liu, C.; Wang, K.; Li, Q.; Zhao, F.; Zhao, K.; Ma, H. Powerful-IoU: More straightforward and faster bounding box regression loss with a nonmonotonic focusing mechanism. *Neural Netw.* **2024**, *170*, 276–284. [CrossRef]
30. Tang, F.; Xu, Z.; Huang, Q.; Wang, J.; Hou, X.; Su, J.; Liu, J. DuAT: Dual-Aggregation Transformer Network for Medical Image Segmentation. In Proceedings of the Pattern Recognition and Computer Vision: 6th Chinese Conference, PRCV 2023, Xiamen, China, 13–15 October 2023; pp. 343–356.
31. Liu, Y.; Shao, Z.; Hoffmann, N. Global attention mechanism: Retain information to enhance channel-spatial interactions. *arXiv* **2021**, arXiv:2112.05561. [CrossRef]
32. Lee, J.; Park, S.; Mo, S.; Ahn, S.; Shin, J. Layer-adaptive sparsity for the magnitude-based pruning. *arXiv* **2020**, arXiv:2010.07611.
33. Wu, D.; Yang, W.; Li, J.; Du, K.; Li, L.; Yang, Z. CRL-YOLO: A Comprehensive Recalibration and Lightweight Detection Model for UAV Power Line Inspections. *IEEE Trans. Instrum. Meas.* **2025**, *74*, 1–21. [CrossRef]
34. Cai, X.; Lai, Q.; Wang, Y.; Wang, W.; Sun, Z.; Yao, Y. Poly kernel inception network for remote sensing detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 27706–27716.
35. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.