



Article

SkySeg-Net: Sky Segmentation-Based Row-Terminal Recognition in Trellised Orchards

Haiyang Gu ^{1,2,†}, Yong Wang ^{1,2,†}, Huaiyang Liu ¹ , Tong Tian ³, Changxing Geng ^{2,*} and Yun Shi ^{1,*}

¹ State Key Laboratory of Efficient Utilization of Arid and Semi-Arid Arable Land in Northern China, Institute of Agricultural Resources and Regional Planning, Chinese Academy of Agricultural Sciences, Beijing 100086, China

² School of Mechanical and Electrical Engineering, Soochow University, Suzhou 215000, China

³ Hainan Tang Huajun Academician Workstation, Key Laboratory of Applied Research on Tropical Crop Information Technology of Hainan Province, Institute of Scientific and Technical Information, Chinese Academy of Tropical Agricultural Sciences, Haikou 570000, China

* Correspondence: chxgeng@suda.edu.cn (C.G.); shiyn@caas.cn (Y.S.)

† These authors contributed equally to this work.

Abstract

Perception in trellised orchards is often challenged by dense canopy occlusion and overhead plastic coverings, which cause pronounced variations in sky visibility at row terminals. Accurately recognizing row terminals, including both row head and row tail positions, is therefore essential for understanding orchard row structures. This study presents SkySeg-Net, a sky segmentation-based framework for row-terminal recognition in trellised orchards. SkySeg-Net is built on an enhanced multi-scale U-Net architecture and employs ResNeSt residual split-attention blocks as the backbone. To improve feature discrimination under complex illumination and occlusion conditions, the Convolutional Block Attention Module (CBAM) is integrated into the downsampling path, while a Pyramid Pooling Module (PPM) is introduced during upsampling to strengthen multi-scale contextual representation. Sky regions are segmented from both front-view and rear-view camera images, and a hierarchical threshold-based pixel-sum analysis is applied to infer row-terminal locations based on sky-region distribution patterns. To support a comprehensive evaluation, a dedicated trellised vineyard dataset was constructed, featuring front-view and rear-view images and covering three representative grapevine growth stages (BBCH 69–71, 73–77, and 79–89). Experimental results show that SkySeg-Net achieves an *mIoU* of 91.21% and an *mPA* of 94.82% for sky segmentation, with a row-terminal recognition accuracy exceeding 98.17% across all growth stages. These results demonstrate that SkySeg-Net provides a robust and reliable visual perception approach for row-terminal recognition in trellised orchard environments.



Academic Editors: Laura Po and Ramón Alberto Mollineda Cárdenas

Received: 12 December 2025

Revised: 23 January 2026

Accepted: 10 February 2026

Published: 13 February 2026

Copyright: © 2026 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: sky region segmentation; end-of-row detection; agricultural robot perception; trellised orchard environments

1. Introduction

In automated and precision agriculture, the perception of orchard row structures is fundamental for supporting robotic operations such as autonomous navigation, spraying, and harvesting. In trellised orchard environments, accurately identifying row terminals, including both row head and row tail positions, remains challenging due to dense canopy occlusion, trellis structures, and complex illumination conditions.

Accurate row-terminal detection constitutes a fundamental perception task in orchard robotic systems. In trellised orchards with narrow row spacing and dense canopy structures, navigation performance critically depends on the timely identification of row boundaries. Reliable row-terminal perception is required to trigger headland maneuvers, maintain steering continuity, and prevent premature stopping or unstable transitions at the end of rows. Without explicit and robust row-terminal detection, downstream navigation and control modules may suffer from degraded reliability, particularly in environments where GNSS signals are intermittent or unavailable.

Early studies on orchard row perception and end-of-row detection largely relied on GNSS-based positioning or predefined orchard maps [1–4]. However, in real-world trellised orchards, dense foliage and supporting structures often lead to GNSS signal degradation and multipath effects, resulting in unstable positioning accuracy. Moreover, these approaches typically require prior maps or structured field layouts, which limits their adaptability across different orchards and growth stages.

To reduce reliance on GNSS, several vision-based methods have been proposed. Geometric approaches extract crop-row edges, centerlines, or vanishing points from RGB images using handcrafted features [5,6]. Although effective in relatively structured scenes, their performance degrades under severe occlusion and illumination variation. GNSS-free row-end detection using depth cameras has also been explored [7,8], demonstrating feasibility in orchard navigation scenarios. However, depth-based methods are sensitive to canopy density and often require additional hardware and calibration.

Three-dimensional perception techniques based on LiDAR and SLAM have been investigated for orchard navigation and row structure estimation [9,10]. While these methods provide rich spatial information, their high sensor cost, computational burden, and system complexity restrict their deployment on lightweight agricultural robots.

With recent advances in deep learning, convolutional neural networks have been widely applied to orchard perception tasks, including fruit detection, canopy segmentation, and row identification [11–19]. These learning-based methods exhibit strong robustness under complex visual conditions. Nevertheless, most existing deep-learning-based approaches focus on vegetation modeling or object detection, rather than explicitly addressing row-terminal recognition in trellised orchards.

An important yet underexplored observation is that sky visibility exhibits consistent structural variation along orchard rows. Within fruit-tree rows, dense canopies and trellis structures restrict the visible sky region to narrow or fragmented areas. In contrast, for row terminals that are close, the opening of the row structure leads to a significant increase in visible sky area. This characteristic remains relatively stable across different growth stages and lighting conditions, making sky regions a reliable visual cue for row-terminal recognition.

To systematically highlight the differences among existing approaches, we summarize representative methods in Table 1, including the sensing modality, core methodology, application target, commonly reported performance metrics (*mIoU*, precision, recall, and row-end detection accuracy), applicability to trellised orchard environments, and key limitations. This comparison demonstrates that most existing methods rely on GNSS, geometric analysis, or depth sensing, whereas the proposed SkySeg-Net model achieves robust row-terminal recognition using sky-region segmentation under dense canopy occlusion. Motivated by this observation, this paper proposes SkySeg-Net, a sky segmentation-based framework for row-terminal recognition in trellised orchards. The proposed method employs an enhanced multi-scale U-Net architecture to achieve accurate sky-region segmentation from camera images and infers row-terminal positions by analyzing the spatial distribution of sky pixels. Unlike GNSS-dependent or geometry-driven approaches, SkySeg-

Net does not require prior maps, depth sensing, or explicit row geometry assumptions. The main contributions of this work are summarized as follows:

1. A sky-region-based perception strategy is introduced to exploit structural variations in sky visibility for robust row-terminal recognition in trellised orchards.
2. An enhanced multi-scale U-Net architecture, termed SkySeg-Net, is developed to achieve accurate sky segmentation under dense canopy occlusion and varying illumination conditions.
3. A dedicated trellised orchard dataset covering multiple grapevine growth stages is constructed to systematically evaluate sky segmentation and row-terminal recognition performance.

Table 1. Comparison of existing methods for orchard row-end perception.

No.	Reference	Sensor	Core Method	Target	Performance Metrics	Key Limitation
[1]	Li et al., 2024	GNSS + LiDAR	Sensor fusion	Navigation	RMSE: 0.12 m	GNSS dropout sensitive
[2]	Liu et al., 2024	GNSS + multi-sensor	Multivariable spraying robot	Navigation and spraying	/	Depends on GNSS
[3]	Su et al., 2024	Multi-sensor	Sensor fusion	Navigation	/	Requires multiple sensors
[4]	Winterhalter et al., 2021	GNSS	Crop row following	Navigation	Accuracy: 85%	Linear crop assumption
[5]	De Silva et al., 2024	RGB	Vision-based crop row	Row guidance	Precision: 87%	Heuristic features
[6]	Shah et al., 2024	RGB	Vision-based row guidance	Row guidance	Accuracy: 86%	Heuristic
[7]	Peng et al., 2022	Depth camera	Geometric analysis	Row-end	Accuracy: 88%	Sensitive to occlusion
[8]	Peng et al., 2023	Depth camera	GNSS-free row-end	Row-end	Accuracy: 91%	Dense canopy reduces reliability
[9]	Jiang et al., 2024	LiDAR	SLAM (NDT-ICP)	Row mapping	Localization error ~0.15 m	High cost, complex system
[10]	Xia et al., 2023	LiDAR + SLAM	Point cloud positioning	Row mapping	RMSE: 0.18 m	High computational cost
[11]	Anagnostis et al., 2021	RGB	U-Net	Tree segmentation	Accuracy: 87%	No row-terminal inference
[12]	Gao et al., 2023	RGB	YOLOv7 detection	Steering markers	Precision: 92%	Focuses on markers, not row-terminal
[13]	Gu et al., 2023	RGB	YOLO + row modeling	Tree trunk	Precision: 91%	Tree-level, not row-end
[14]	Kang et al., 2020	RGB	CNN segmentation	Fruit/canopy	<i>mIoU</i> : 94%	Canopy only
[15]	Wu et al., 2025	RGB	RT-DETR	Row navigation	<i>mAP</i> : 89%	Focused on navigation
[16]	Xu et al., 2025	UAV RGB	Row detection	Path planning	Accuracy: 90%	Aerial images only
[17]	Xue et al., 2023	RGB-D	DeepLabv3+	Canopy segmentation	<i>mIoU</i> : 95%	Not row-terminal
[18]	Zhang et al., 2024	RGB	YOLOv5 detection	Tree target	Precision: 90%	Not row-terminal
[19]	Zhang et al., 2022	UAV RGB	Mask R-CNN + PointRend	Canopy	<i>mIoU</i> : 92%	Vegetation only
SkySeg-Net	This work	RGB	SkySeg-Net	Row-terminal	<i>mIoU</i> : 91.21%, Precision: 95.8%, Recall: 94.72%, Accuracy: 98.17%	Robust under dense canopy

2. Materials and Methods

2.1. Dataset Preparation

2.1.1. Environmental Characteristics of the Trellis Orchard

The experiments were conducted at a high-quality grape production base in Maoshatang Village, Bacheng Town, Kunshan City, Suzhou, Jiangsu Province, China. The orchard employed a typical horizontal trellis cultivation system. Figure 1 presents an aerial view of the orchard captured by a drone, where the green box highlights the entire base comprising six greenhouses. Within the region, areas labeled A1–A9 (red box) correspond to boundary zones that are open and spacious, making them suitable for end-of-row discrimination. Taking area A1 as an example (Figure 2), the trellis structure consists of cement columns supporting grapevine trunks, with branches extending laterally to form a symmetrical “Y-shaped” configuration. The wire mesh was positioned approximately 1.7 m above the ground. The average canopy height at ripening was 1.4 m. The distance between fruit trees was around 2.4 m, and the spacing between rows averaged 2.2 m.



Figure 1. Full view of the trellis orchard.



Figure 2. Situation inside the trellis orchard.

2.1.2. Building a Dataset

Although several publicly available datasets have been proposed for orchard and agricultural perception tasks, most existing benchmarks are primarily designed for fruit detection, canopy segmentation, or general localization, rather than navigation-oriented row-terminal perception.

Representative datasets include the MinneApple dataset, which focuses on fruit detection and instance segmentation in open-field apple orchards [20]. UAV-based orchard semantic segmentation datasets, such as those introduced by Anagnostis et al. [11], are mainly designed for aerial perception and orchard mapping and therefore do not reflect the visual constraints encountered by ground robots operating within narrow orchard rows.

In addition, general navigation datasets such as the 4 Seasons dataset target long-term visual localization under varying weather conditions but are not tailored to agricultural environments or structured orchard row scenarios [21].

Consequently, existing datasets generally lack task-driven annotations and evaluation protocols that explicitly support row-terminal recognition in trellised orchards. Motivated by this gap, the dataset constructed in this study is specifically designed to support sky-region segmentation and navigation-relevant row-terminal perception under dense canopy occlusion, trellis structures, and narrow row spacing, thereby providing a dedicated benchmark for perception-driven orchard navigation.

Based on the environmental characteristics described in Section 2.1.1, a hardware platform for the trellis orchard robot was established, as shown in Figure 3. Intel RealSense depth cameras (Intel Corporation, Santa Clara, CA, USA) were mounted at the front and rear of the robot (hereinafter referred to as the front and rear cameras). Each camera was installed at a height of 0.8 m above the ground and aligned with the centerline of the grapevine row during data acquisition to ensure consistent visual perspectives. To ensure data diversity and enhance the generalization ability of the segmentation model for row-terminal recognition, inter-row video sequences were collected across three key grapevine growth stages: the fruit-setting stage, the expansion stage, and the maturity stage. To reduce redundancy and temporal similarity, a frame-sampling strategy was applied in which one frame was extracted every 30 frames to construct the image dataset. All images were stored at a resolution of 1280×720 pixels. The final dataset consisted of 4500 images, including 1500 images for each growth stage. The dataset covered different time periods—noon (11:00–12:00), afternoon (14:30–15:30), and evening (16:00–17:00)—as well as both sunny and cloudy weather conditions. This acquisition strategy ensured that the model would be exposed to diverse illumination and environmental variations, thereby improving robustness. Representative samples are shown in Figure 4.

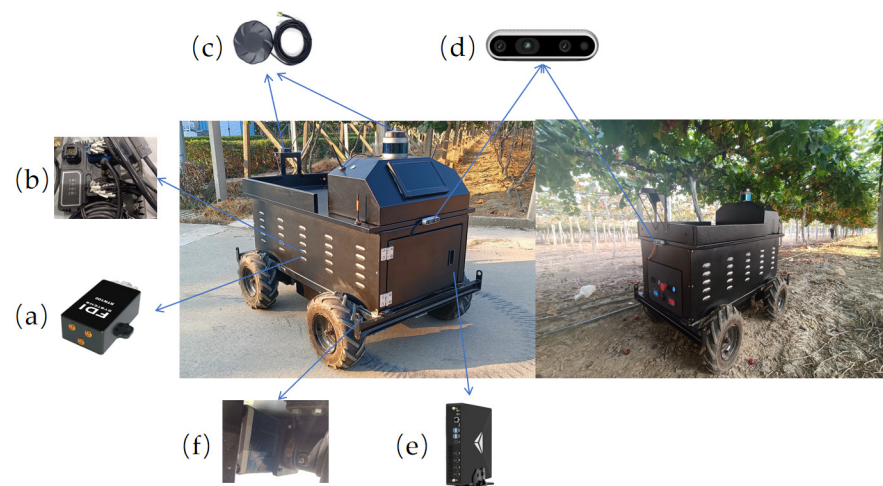


Figure 3. Trellis orchard robot, (a) GNSS dual-antenna module; (b) vehicle controller; (c) vehicle antenna; (d) Realsense cameras; (e) industrial computer; (f) motor drive mechanism.

Compared with existing agricultural vision datasets that mainly focus on fruit detection, canopy segmentation, or general navigation perception, the proposed dataset provides several distinctive features:

1. Pixel-level sky annotations explicitly designed to support navigation-oriented row-terminal discrimination;
2. Data collected exclusively in trellised orchard environments with dense canopy occlusion and narrow row spacing;

3. Coverage of multiple grapevine growth stages and diverse illumination conditions;
4. Explicit alignment with row-terminal perception tasks that serve as triggers for downstream navigation and control, rather than standalone segmentation evaluation.

These characteristics enable the dataset to serve as a dedicated benchmark for studying sky-based row-terminal recognition in trellised orchards.

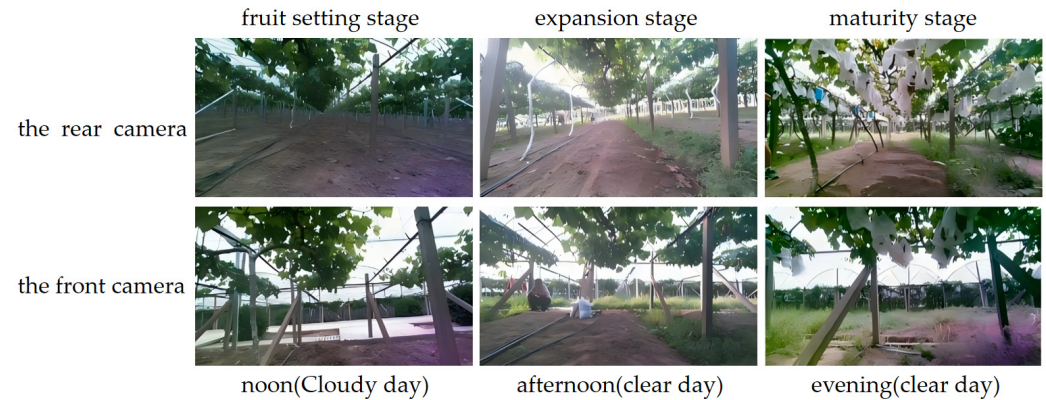


Figure 4. The trellis orchard dataset.

2.1.3. Building a Segmented Dataset

In image processing, the region of interest (ROI) refers to the specific area that the algorithm focuses on. By defining the ROI, the number of pixels to be processed can be reduced, thereby decreasing the computational load during training and improving processing efficiency. Setting an appropriate ROI allows the segmentation algorithm to concentrate on a targeted region, which facilitates accurate extraction and analysis. The tail-discrimination method determines the robot's travel status by segmenting the sky region in the vineyard images. Since the camera is positioned at the center of the grape row, the ROI can also be defined around the central area. Therefore, the tail-discrimination method only needs to focus on the sky region of the current fruit-tree row. Based on this requirement, a cropped region of [0:320, 420:840] is extracted from the original 1280×720 images, as shown by the red dashed boxes in Figure 5a–c. The resulting ROI resolution is 320×320 , corresponding to the lower-right corner in the figures. Unified image preprocessing is applied to datasets from different growth stages to construct a complete semantic segmentation dataset, ensuring that the sky regions involved are segmented consistently based on the original images.

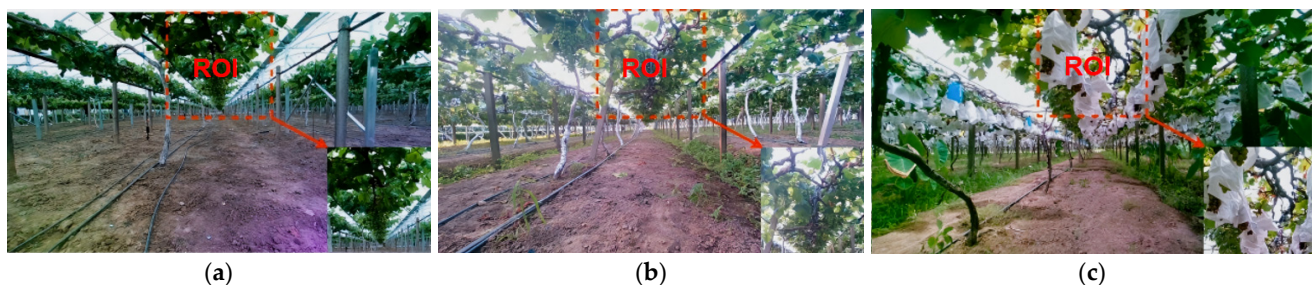


Figure 5. Semantic segmentation dataset at different growth stages from the perspective of the front camera, (a) fruit setting stage; (b) expansion stage; (c) maturity stage.

2.1.4. Preprocessing of the Sky Segmentation Dataset

In this study, the Labelme tool was used to annotate the collected data. The ROI image was divided into two categories: foreground and background. The foreground corresponds to the sky region to be segmented, while the background includes all remaining areas.

Objects in each image were manually selected and annotated according to their category, and the generated JavaScript Object Notation (JSON) files were stored in a custom directory. Data augmentation is an effective approach for constructing robust deep learning models [22,23]. To enrich the dataset and improve the generalization ability of the segmentation model—while mitigating overfitting and reducing the cost of data collection and annotation—a total of 355 images were manually labeled (114 images in the fruit-setting stage, 135 images in the expansion stage, and 106 images in the maturity stage). As illustrated in Figure 6, the original images and their corresponding label masks were expanded to 2840 samples through augmentation operations such as 90° rotation, random splicing, salt-and-pepper and Gaussian noise reduction, darkening, and brightness enhancement. The augmented dataset covers the three grape growth stages, resulting in 909 images for the fruit-setting stage, 1079 for the expansion stage, and 852 for the maturity stage. The dataset was subsequently divided into training, validation, and test subsets in an 8:1:1 ratio.

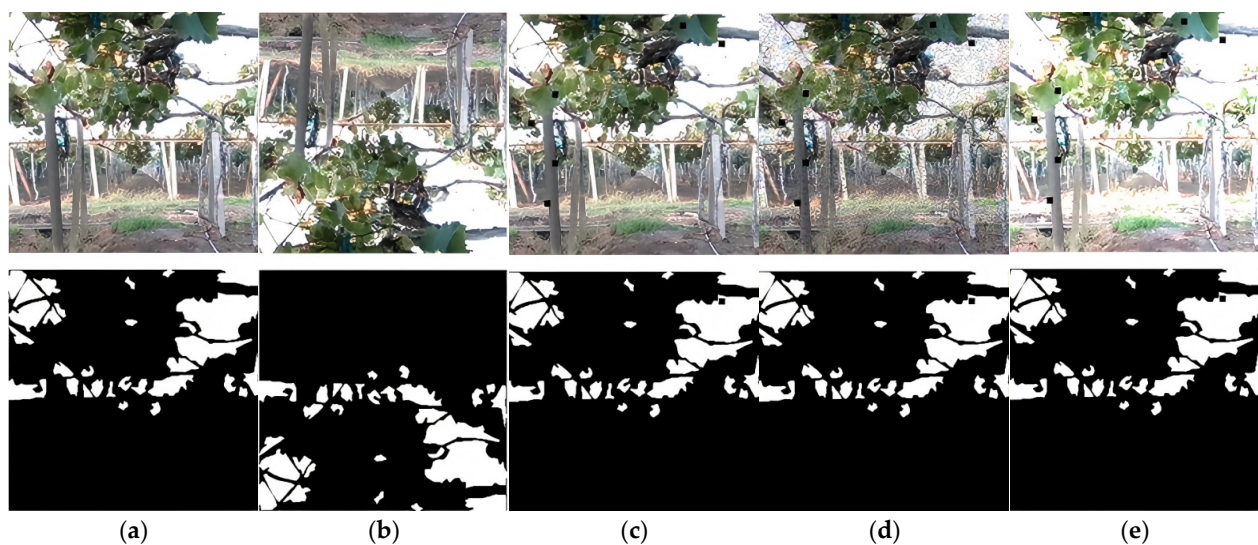


Figure 6. Effect of dataset augmentation, (a) raw image; (b) flip; (c) salt-and-pepper noise; (d) gaussian and salt-and-pepper; (e) Salt and pepper and brightness.

2.2. SkySeg-Net: An Improved Multi-Scale U-Net for Sky-Region Segmentation

2.2.1. U-Net Semantic Segmentation Model

U-Net [24] is a widely used convolutional neural network for pixel-level semantic segmentation, making it particularly suitable for precise object boundary extraction. It uses a symmetric encoder–decoder structure with skip connections to preserve spatial details while combining low-level and high-level features. In the encoder, convolutional blocks extract features while reducing spatial resolution. The decoder upsamples feature maps to recover original resolution and reconstruct spatial information. U-Net provides a strong baseline for semantic segmentation, but its fixed receptive field and limited feature diversity can reduce performance in complex orchard scenes with occlusions and narrow sky regions.

2.2.2. Deep Residual Network Model (ResNeSt)

ResNeSt [25] enhances conventional residual networks by introducing nested skip connections and a multi-branch attention module (MBAM) within each residual block. These improvements increase feature diversity and allow the network to capture fine-grained details across multiple scales. By integrating ResNeSt into the U-Net encoder, SkySeg-Net can better represent sky-edge and branch features under dense canopy occlusion and varying illumination, which is critical for robust row-terminal recognition in trellised orchards.

2.2.3. Convolution Block Attention Module (CBAM)

CBAM [26] applies channel attention and spatial attention sequentially to the feature maps. Channel attention emphasizes the importance of different channels, while spatial attention highlights relevant spatial regions. In SkySeg-Net, the CBAM is inserted after each ResNeSt downsampling stage, enabling the network to focus on critical sky-edge and canopy contours while suppressing background interference. This attention mechanism is particularly important in narrow trellis rows, where sky visibility is limited.

2.2.4. Pyramid Pooling Module (PPM)

The PPM [27] aggregates contextual information at multiple scales. By applying pooling at different window sizes, the PPM captures both the global context and local details, improving segmentation accuracy for both narrow sky strips and larger sky regions near row terminals. Integrating the PPM into the decoder complements ResNeSt feature extraction, enhancing multi-scale perception in complex orchard scenes.

2.2.5. SkySeg-Net: Multi-Scale Mechanism U-Net for Sky Segmentation

To improve the segmentation accuracy of grape branch and sky-edge features, this study proposes SkySeg-Net, an enhanced U-Net architecture illustrated in Figure 7. The fixed receptive field of a conventional backbone is often insufficient for multi-scale targets, leading to poor representation of large objects and interference for small ones. To address this, standard convolution modules are replaced with ResNeSt blocks, which employ grouped convolutions to increase feature diversity, expand the receptive field, and better capture semantic information and target boundaries.

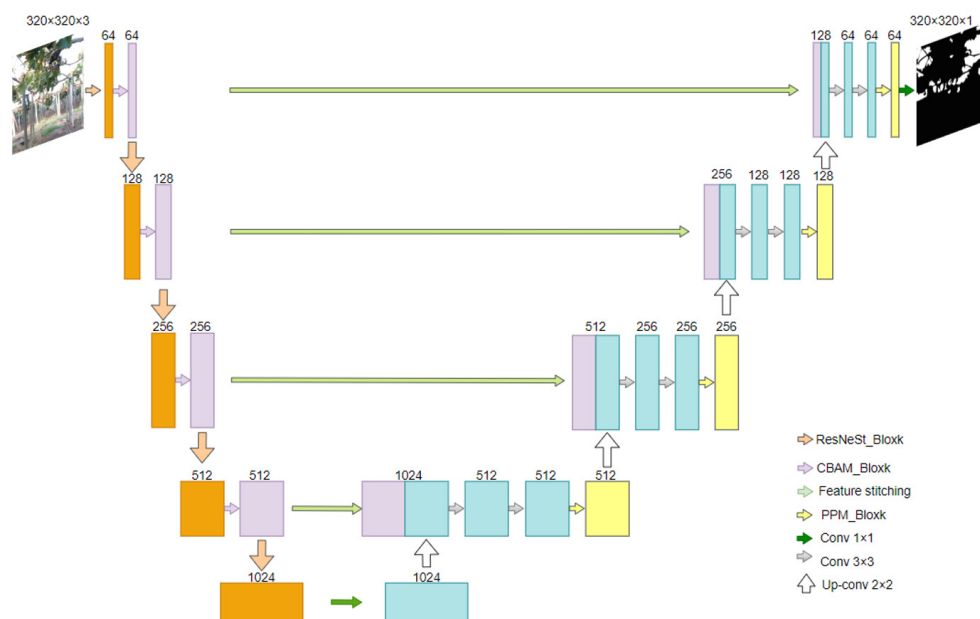


Figure 7. Diagram of the improved U-Net network structure.

To preserve low-level spatial details, a CBAM attention module is appended after each downsampling stage in the ResNeSt-U-Net encoder. The CBAM recalibrates features across channel and spatial dimensions, emphasizing sky–canopy boundaries while suppressing background noise. In the decoder, a Pyramid Pooling Module (PPM) fuses multi-scale contextual information, enabling the network to recognize both small and large sky regions and improving robustness under illumination changes and canopy occlusion.

SkySeg-Net operates on a 320×320 ROI input following a standard U-Net encoder–decoder paradigm. Hierarchical features are extracted in the encoder, enhanced via the

CBAM, and progressively upsampled in the decoder. Skip connections transfer high-resolution information from the encoder to the decoder. The PPM aggregates contextual cues at multiple scales, and the fused features are combined to generate a single-channel logit map. A sigmoid function converts the logits to a probability map, which is then binarized to produce the sky mask used by the downstream row-terminal discrimination method.

The proposed SkySeg-Net method combines the components mentioned above to enhance sky-region segmentation for trellised orchards. Compared to conventional U-Nets, the innovations of SkySeg-Net include

1. Multi-scale feature representation through ResNeSt blocks and the PPM, thereby improving the recognition of both small and large sky regions.
2. Attention-guided feature enhancement via the CBAM, allowing the network to focus on critical sky-edge and canopy features under occlusion.
3. Task-specific sky-region utilization for row-terminal recognition, translating structural sky variations into actionable navigation cues, which is a novel contribution not addressed by prior segmentation methods.

These design choices collectively provide task-driven improvements, enhancing segmentation robustness in complex agricultural environments and directly supporting autonomous orchard navigation through accurate and reliable row-terminal perception.

2.2.6. Model Training and Evaluation Indicators

This study trains SkySeg-Net, the proposed multi-scale mechanism U-Net for sky-region segmentation, using the deep learning environment configured in Table 2. The network is trained for 100 epochs with a batch size of 8 using stochastic gradient descent (SGD). The initial learning rate is set to 1×10^{-4} and the momentum is 0.95. The training configuration (100 epochs and a batch size of 8) is selected to balance convergence sufficiency and computational constraints. After augmentation, the dataset contains 2840 ROI samples and is split into training/validation/test sets with an 8:1:1 ratio. With a batch size of 8, the training set provides 284 iterations per epoch ($2272/8$), resulting in 28,400 SGD updates over 100 epochs, which is sufficient to achieve stable convergence under the adopted learning-rate schedule. Meanwhile, the incorporation of ResNeSt together with the CBAM and PPM increases memory usage at an input resolution of 320×320 , and a batch size 8 is a practical choice to maintain training stability within GPU memory limits. Binary cross-entropy with logits (BCEWithLogitsLoss) is used as the loss function. At the start of training, the pre-trained weights from “UNet.pth” are loaded to initialize the network. During subsequent runs, we initialize training using the best checkpoint from the previous session and select the final model according to the best *mIoU* validation score, which helps avoid overfitting and ensures reproducible performance.

Table 2. Configurations of the training and testing environments.

Configuration	Parameter
Operating System	Ubuntu 20.04
Central Processing Unit (CPU)	Inter(R) Xeon(R)Gold 6226R CPU
Graphics Processing Unit (GPU)	NVIDIA Tesla V100 SXM2
Random Access Memory (RAM)	256 G
Deep Learning Environment	CUDA11.3
Pytorch	1.13.1

The performance of SkySeg-Net is assessed using standard semantic segmentation metrics, including the mean Intersection over Union (*mIoU*), mean Pixel Accuracy (*mPA*), precision, and recall, as defined in Equations (1)–(6). These metrics are widely adopted

in the semantic segmentation literature [28,29]. The *mIoU* calculates the average overlap between predicted and ground-truth regions across all classes. The *mPA* measures the average pixel-level accuracy by computing the ratio of correctly predicted pixels to the total number of pixels. Precision indicates the proportion of true positives among all predicted positives, while recall represents the proportion of true positives among all actual positives. These calculations rely on the counts of true positives (*TPs*), false positives (*FPs*), false negatives (*FNs*), and true negatives (*TNs*).

$$IoU = \frac{TP}{TP + FP + FN}, \quad (1)$$

$$mIoU = \frac{1}{k+1} \sum_{i=0}^k \frac{TP}{TP + FP + FN}, \quad (2)$$

$$Pixel\ Accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \quad (3)$$

$$mPA = \frac{1}{n} \sum_{i=0}^n PA_i \times 100\%, \quad (4)$$

$$Precision = \frac{TP}{TP + FP}, \quad (5)$$

$$Recall = \frac{TP}{TN + FP}, \quad (6)$$

2.3. Vineyard Row Tail Discrimination Method Based on SkySeg-Net

2.3.1. Row-End Discrimination Method

Row-end discrimination is a key step for enabling adaptive navigation switching in trellis orchard robots. The procedure is as follows:

1. The front and rear cameras are mounted along the central axis of the robot and aligned with the center of the grapevine row. RGB images are acquired in real time, and ROI patches with a resolution of 320×320 are extracted.
2. The patches are fed into the trained SkySeg-Net model for semantic segmentation, producing a corresponding mask map, as illustrated in Figure 8. The number of pixels within the blank region of the mask is summed to obtain the area S , which is then compared with a predefined threshold M . According to Equation (7), the robot's state is determined as follows: $S \geq M$ represents the outline, while $S < M$ represents the inline.

$$output_position = \begin{cases} outline & , \text{ if } S \geq M \\ inline & , \text{ if } S < M' \end{cases} \quad (7)$$



Figure 8. Mask map after slicing.

Figure 9 illustrates how the ROI regions of the front and rear cameras capture different scene characteristics at various positions. At Position 1, both ROIs mainly capture the sky. At Position 2, the front camera's ROI primarily observes the trellis orchard, while the rear camera's ROI focuses on the sky. At Position 3, both ROIs mainly capture the trellis orchard. At Position 4, the front camera captures the sky, while the rear camera captures the trellis orchard. At Position 5, both ROIs once again capture the sky. Based on these observations, the robot's position is determined using the following rules:

1. At Positions 1 and 5, both the front and rear cameras output outline signals, indicating that the robot is out of line.
2. At Position 3, both cameras output inline signals, indicating that the robot is properly aligned.
3. At Position 2, the front camera outputs the inline, while the rear camera outputs the outline, which is determined as the row boundary (entry stage).
4. At Position 4, the front camera outputs the outline, while the rear camera outputs the inline, which is determined as the row boundary (exit stage), where navigation switching is required.

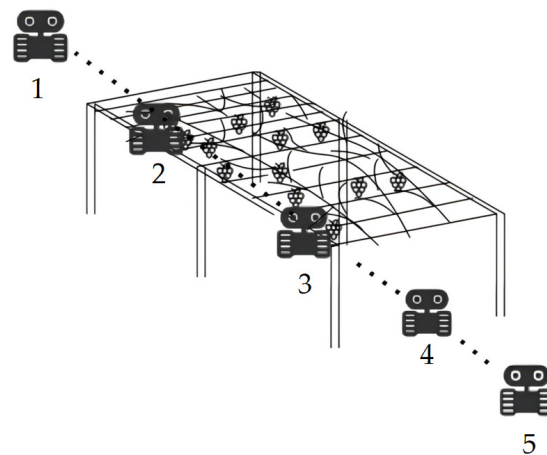


Figure 9. In-line, out-of-line, and line boundary location maps.

2.3.2. Line-Tail Discrimination Method

To evaluate the accuracy of row-tail discrimination, we apply a pixel-area thresholding strategy based on the sky-mask predictions of SkySeg-Net. Equations (8) and (9) are novel threshold calibration formulas proposed in this study to obtain a universal pixel-area threshold that is robust to variations in canopy density and illumination. Specifically, for a single test trial, we compute the mean sky-pixel area K_i across m predicted label maps, where S_i denotes the sky-region pixel area of the i -th label map. This averaging operation (Equation (8)) reduces the influence of frame-level noise and occasional segmentation fluctuations. The process is then repeated for e independent data groups (e.g., different runs or different canopy-density conditions), and the resulting K_i values are averaged to obtain a single universal threshold, M (Equation (9)). During online verification, the sky-region pixel area S is measured in real time from the predicted mask, and the row-tail position is determined by comparing S with M according to Equation (7).

$$K_i = \frac{1}{m} \cdot \sum_{i=1}^m S_i, \quad (8)$$

$$M = \frac{1}{e} \cdot \sum_{i=1}^e K_i \quad (9)$$

3. Results

3.1. Ablation Experiment

To evaluate the individual contributions of the proposed modules in SkySeg-Net, four variants of U-Net-based networks were constructed: (1) SkySeg-Net (Baseline), the original U-Net; (2) SkySeg-Net (ResNeSt), where the standard convolution in the downsampling stages was replaced with ResNeSt blocks; (3) SkySeg-Net (ResNeSt + CBAM), which further incorporated the CBAM attention module after each downsampling stage; and (4) SkySeg-Net (Full, ResNeSt + CBAM + PPM), which additionally integrated the Pyramid Pooling Module (PPM) after the upsampling convolution layers. All models were trained with identical parameters as described in Section 2.2.6.

The segmentation performance of each variant was evaluated on test dataset A (a robot operating along the row of the trellis orchard) and test dataset B (a robot located at the end of the row), covering the fruit-setting, expansion, and maturity stages. Representative segmentation outputs are shown in Figures 10 and 11, while the quantitative results are summarized in Table 3.

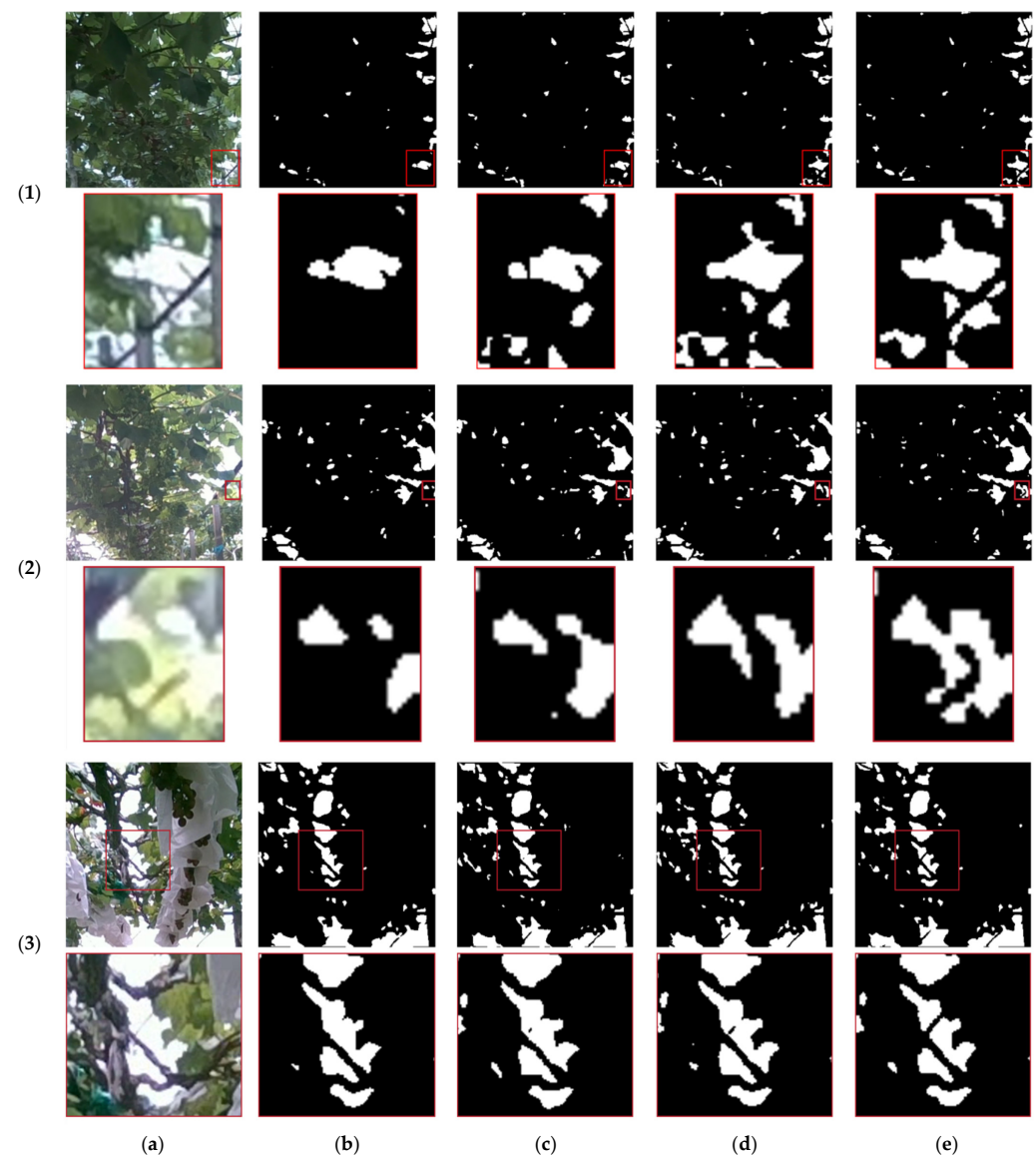


Figure 10. Comparison of ablation experimental results based on test dataset A. Note: (a) Original image, (b) SkySeg-Net (Baseline), (c) SkySeg-Net (ResNeSt), (d) SkySeg-Net (ResNeSt + CBAM), and (e) SkySeg-Net (Full).

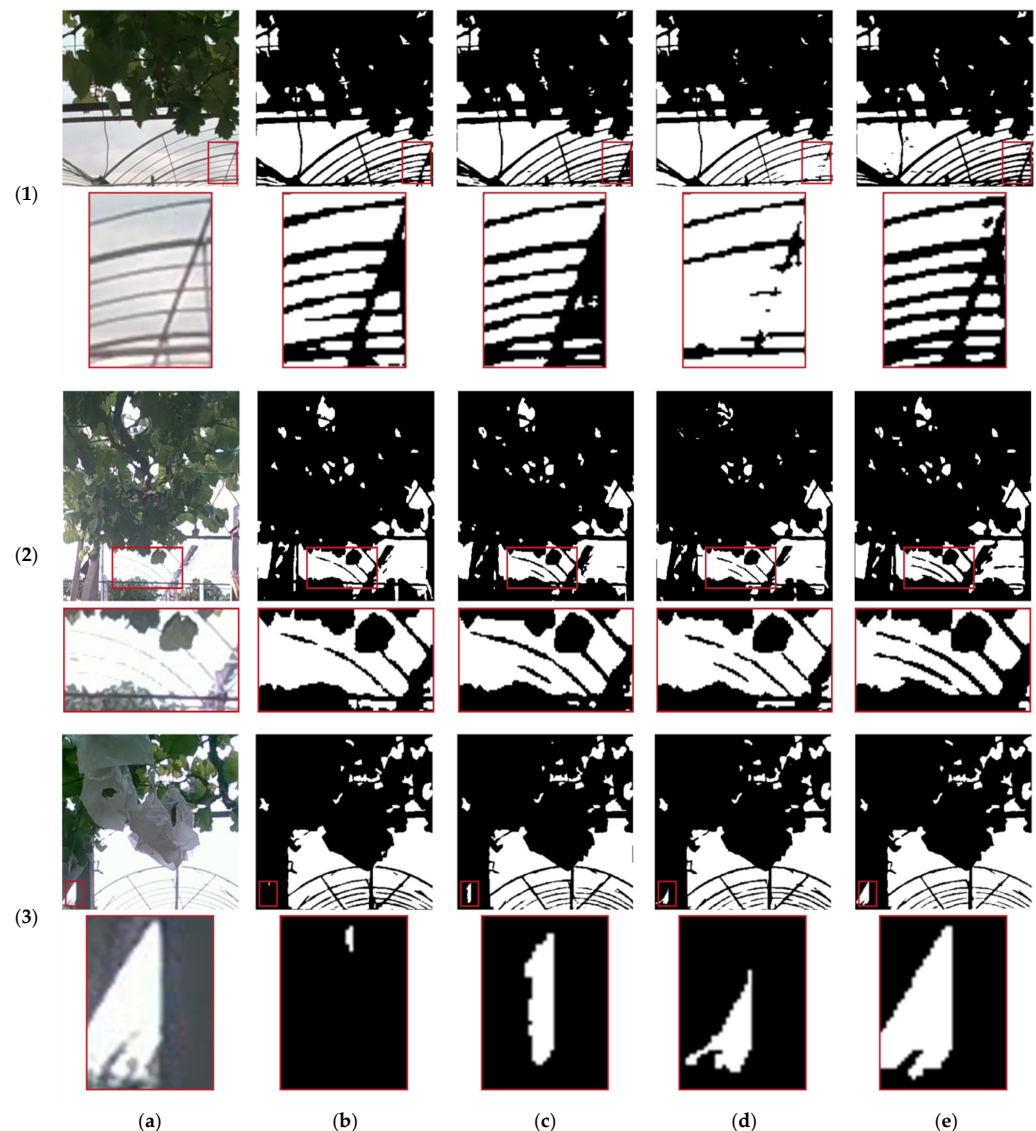


Figure 11. Comparison of ablation experimental results based on test dataset B. Note: (a) Original image, (b) SkySeg-Net (Baseline), (c) SkySeg-Net (ResNeSt), (d) SkySeg-Net (ResNeSt + CBAM), and (e) SkySeg-Net (Full).

Table 3. Comparison of the effects of different improved network structures.

Model	<i>mIoU</i> /%	<i>mPA</i> /%	Precision/%	Recall/%
SkySeg-Net (Baseline)	90.12	92.75	94.74	93.92
SkySeg-Net (ResNeSt)	90.34	92.60	94.71	92.04
SkySeg-Net (ResNeSt + CBAM)	90.61	94.64	95.73	93.64
SkySeg-Net (Full)	91.21	94.82	95.80	94.72

From Table 3, it can be observed that replacing standard convolution with ResNeSt blocks (Experiment 2) significantly enhanced multi-scale feature representation, improving the model's ability to capture both large sky regions and fragmented canopy structures. Specifically, the *mIoU*, *mPA*, precision, and recall reached 90.34%, 92.60%, 94.71%, and 92.04%, respectively (Figures 10(1c–3c) and 11(1c–3c)). Adding the CBAM after each downsampling stage (Experiment 3) allowed the network to adaptively focus on critical sky-canopy boundaries while suppressing background noise caused by illumination variations and occlusions. This led to modest yet consistent performance gains: the *mIoU*, *mPA*, precision, and recall increased by 0.27%, 2.04%, 1.02%, and 1.6%, respectively (Figures 10(1d–3d) and 11(1d–3d)).

Finally, incorporating PPM in the decoder (Experiment 4) further enhanced the perception of fine details and multi-scale contextual information, leading to improved segmentation accuracy. Compared with the original U-Net, the *mIoU*, *mPA*, precision, and recall increased by 1.09%, 2.07%, 1.06%, and 1.79%, respectively (Figures 10(1e–3e) and 11(1e–3e)).

Overall, the ablation experiments confirm that ResNeSt improves feature representation, and the CBAM focuses attention on relevant targets, while the PPM enhances multi-scale context perception, making SkySeg-Net (Full) the best-performing variant. These results demonstrate the effectiveness of the proposed architectural enhancements in capturing the intricate features of trellised orchards and provide the foundation for robust row-tail discrimination.

3.2. Comparison of Segmentation Performance Against Other Models

To evaluate the segmentation performance of SkySeg-Net (Full), comparative experiments were conducted. The comparison models included U-Net derivatives (R2U-Net [30], AttenU-Net [31], R2AttenU-Net [32], and U-Net++ [33]) as well as other semantic segmentation models (PSPNet [27], HRNet [34], and DeepLabv3+ [35]). All models were trained under identical conditions as described in Section 2.2.6 and evaluated on two test datasets: dataset A, representing the robot operating along the row of the trellis orchard, and dataset B, representing the robot at the end of the row. Each dataset covers different grape growth stages (fruit-setting, expansion, and maturity). Figures 12 and 13 illustrate representative segmentation results across models and growth stages, and the corresponding quantitative metrics are summarized in Table 4.

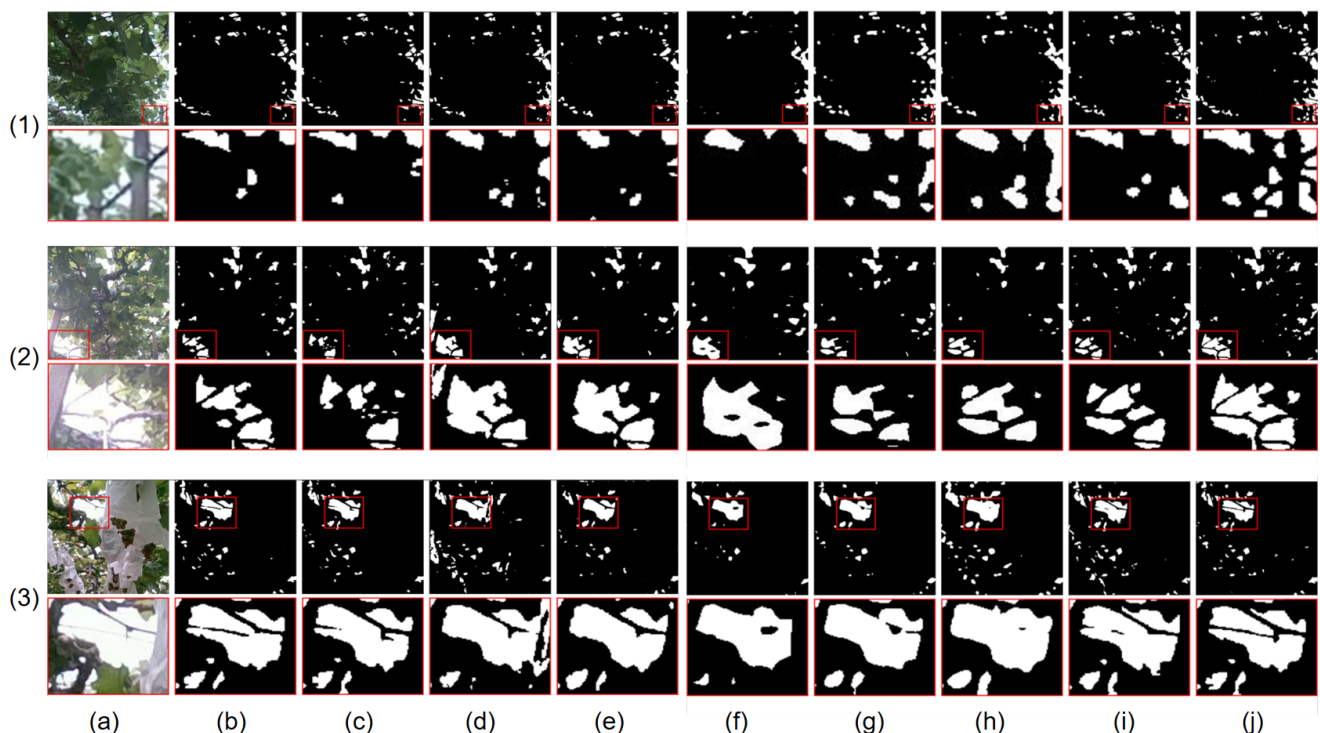


Figure 12. The experimental results of different algorithms based on test dataset A are compared during (1) the fruit-setting period, (2) the expansion period, and (3) the maturity period. Note: (a) Original, (b) U-Net, (c) PSPNet, (d) HRNet, (e) DeepLabv3+, (f) R2U-Net, (g) AttenU-Net, (h) R2AttU-Net, (i) U-Net++, and (j) SkySeg-Net.

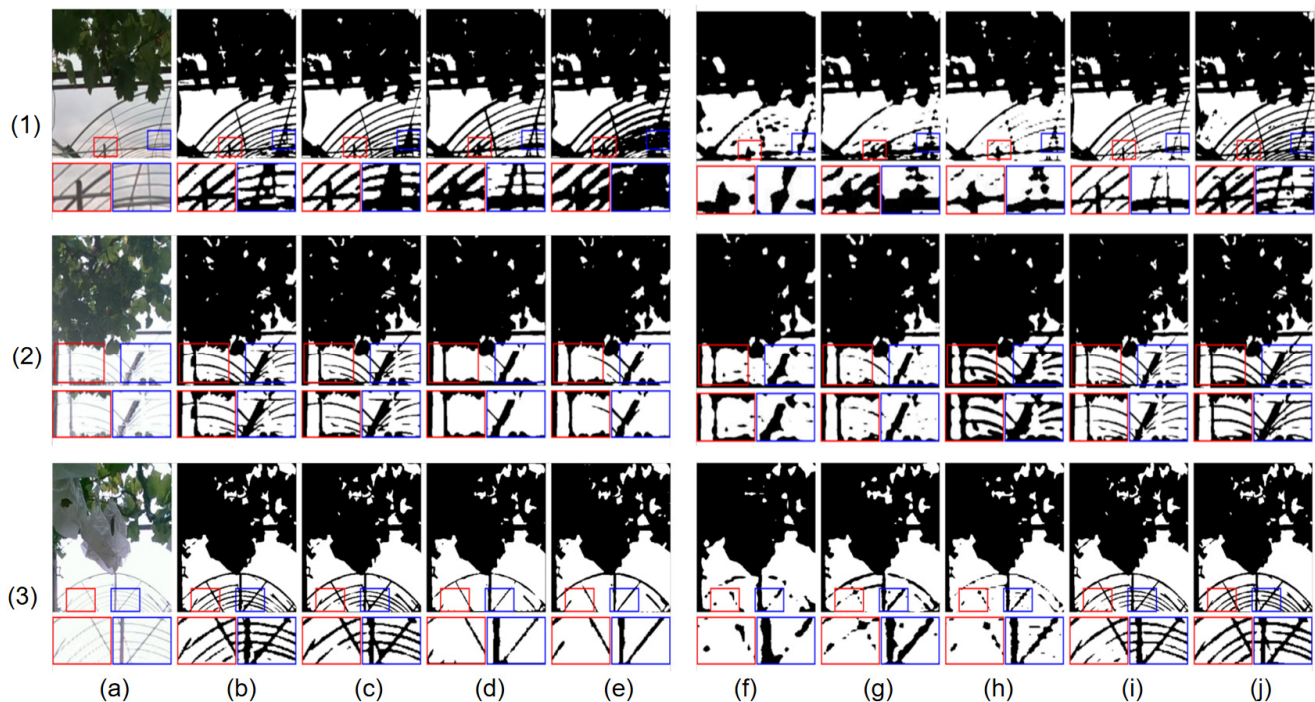


Figure 13. The experimental results of different algorithms based on test dataset B are compared during (1) the fruit-setting period, (2) the expansion period, and (3) the maturity period. Note: (a) Original, (b) U-Net, (c) PSPNet, (d) HRNet, (e) DeepLabv3+, (f) R2U-Net, (g) AttenU-Net, (h) R2AttenU-Net, (i) U-Net++, and (j) SkySeg-Net.

Table 4. Comparison of effects between different algorithms.

Model	<i>mIoU</i> /%	<i>mPA</i> /%	Precision/%	Recall/%
PSPNet	89.54	88.97	90.38	89.97
HRNet	90.50	90.35	91.28	90.45
DeepLabv3+	91.22	93.16	93.42	94.16
R2U-Net	77.52	82.02	80.05	81.98
AttenU-Net	82.58	83.40	82.74	84.36
R2AttenU-Net	83.15	83.34	84.65	85.35
U-Net++	92.95	94.45	95.59	93.45
SkySeg-Net	91.21	94.82	95.80	94.72

Non-U-Net models showed varying performance levels. PSPNet achieved relatively high *mIoU* and precision scores (~90%), but global pooling reduced its resolution, limiting fine branch-edge segmentation (Figure 13(1c)). HRNet slightly improved its *mIoU* and precision scores (0.96% and 0.9%) via multi-scale fusion, but its attention to fine details was insufficient, and computational complexity hindered real-time use. DeepLabv3+ performed well in fruit-setting stages but required extensive training data, with suboptimal edge segmentation under strong illumination (Figures 12(1e) and 13(1e)).

U-Net derivatives exhibited significant performance differences. R2U-Net performed worst, struggling to capture branch edges under strong light (Figure 12(2f)). AttenU-Net improved detail segmentation via attention but missegmented bright regions (e.g., white grape bags, Figure 12(3g)). R2AttenU-Net further improved the metrics but remained sensitive to illumination changes. U-Net++ achieved the best derivative performance, accurately segmenting branch edges (Figure 12(1i–3i)), but a limited receptive field constrained large-scale and fine-edge object segmentation.

SkySeg-Net (Full) demonstrated competitive and task-relevant performance compared with state-of-the-art segmentation models. By integrating ResNeSt, the CBAM, and the

PPM, it effectively captured multi-scale semantic and structural features, enhanced attention to key regions, and suppressed noise. On the test sets, $mIoU$, mPA , precision, and recall reached 91.21%, 94.82%, 95.80%, and 94.72%, respectively. Compared with DeepLabv3+, its mPA , precision, and recall improved by 1.66%, 2.38%, and 0.56%, respectively, and compared with U-Net++, these metrics increased by 0.37%, 0.21%, and 1.27%, respectively. These results highlight SkySeg-Net's superior branch-edge capture and blank canopy segmentation capabilities, validating its effectiveness for trellised orchard perception.

3.3. The Trellis Orchard Row Tail Discrimination Experiment

To further evaluate the practical applicability of the proposed SkySeg-Net model for autonomous orchard navigation, we conducted row-tail discrimination experiments using the validation set of trellis grape orchards at different growth stages, including the fruit-setting, expansion, and maturity stages. The test images were grouped by growth stage, with five repetitions per group, resulting in a total of 15 independent datasets (corresponding to e in Equation (9)). The empirical value for the number of frames m used in the threshold calibration (Equation (8)) was set to 200. Based on Equations (8) and (9), a universal pixel-area threshold ($M = 30821$) that is robust across varying canopy densities and lighting conditions was obtained, as shown in Table 5.

Table 5. The pixel area value of the end of the line in different growth periods.

Growth Phase	Tree Row	Pixel Area	Average Pixel Value
Fruit-setting period	1	33,602	30,821
	2	34,803	
	3	29,368	
	4	31,802	
	5	32,098	
Expansion period	1	28,419	30,821
	2	30,156	
	3	31,598	
	4	29,542	
	5	30,509	
Maturity period	1	27,907	30,821
	2	33,608	
	3	28,937	
	4	39,874	
	5	30,102	

Using this threshold, we used the predicted masks of the front and rear cameras to measure the sky-region pixel areas in real time and compare them with M to determine the robot's relative position within the row. As illustrated in Figure 14 and combined with the observations in Section 2.1.1, the pixel values in the blank area of the rear camera were 15,361, 11,798, and 18,122 for the fruit-setting, expansion, and maturity stages, respectively, and all values were below M . Conversely, the pixel values in the blank area of the front camera were 31,950, 51,177, and 42,344, respectively (all above M), indicating that the robot was exiting the row-scaffold area.

The recognition accuracy is summarized in Table 6. The overall average recognition rate reached 98.17%, confirming the reliability of the proposed strategy. During fruit setting, rates were slightly lower (96.42%, 97.88%, and 98.22%) due to uneven canopy density blurring sky-vegetation boundaries. During expansion, recognition improved (97.36%, 98.80%, and 97.89%) as growth became more uniform. During maturity, rates peaked (99.40%, 98.02%, and 99.50%), with dense, stable canopies providing rich, distinctive features.

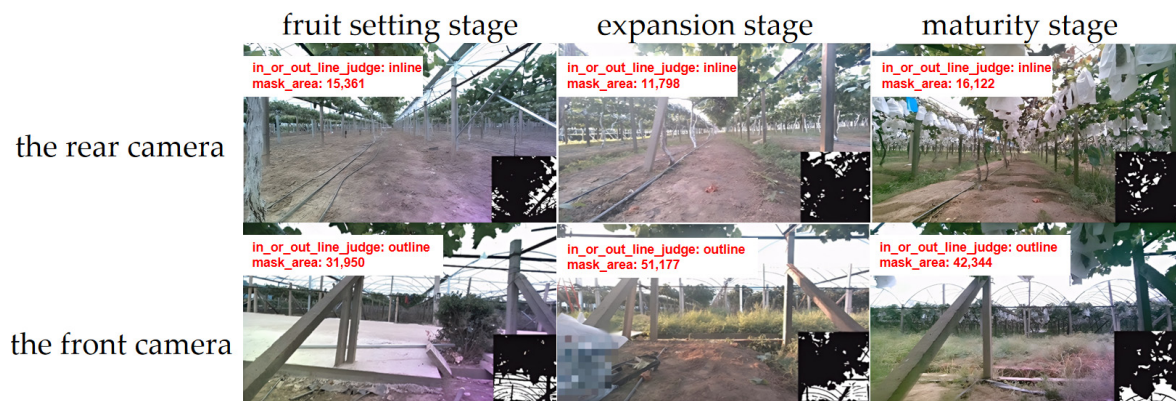


Figure 14. Row-tail discriminant across different growth cycles.

Table 6. Comparison of recognition rates in different growth periods.

Growth Phase	Tree Row	Number of Pictures at the End of the Line	Number of Accurate Identifications	Number of Error Identification	Recognition Rate/%
fruit setting period	1	587	566	21	96.42
	2	630	617	13	97.88
	3	614	603	11	98.22
expansion period	1	587	572	15	97.36
	2	630	622	8	98.8
	3	614	601	13	97.89
maturity period	1	587	583	4	99.40
	2	630	618	12	98.02
	3	614	611	3	99.50

These results demonstrate that SkySeg-Net combined with pixel-area thresholding enables robust and accurate row-tail recognition, effectively distinguishing in-line and out-of-line conditions and directly supporting autonomous navigation and seamless switching between algorithms in trellised orchard environments.

4. Discussion

This study developed and validated a SkySeg-Net-based trellis orchard row-terminal identification method, addressing the practical challenge of insufficient robustness of conventional perception approaches in complex orchard environments, including dense branch and leaf occlusion, varying illumination levels, and different growth stages. By leveraging multi-scale feature fusion and attention mechanisms, the proposed method enhances the segmentation accuracy of row-terminal edges—targets that are particularly vulnerable to environmental interference. This improvement strengthens feature extraction and edge localization under challenging visual conditions, outperforming traditional geometric, depth-based, and GNSS-dependent approaches.

Extensive evaluations across multiple grapevine growth stages demonstrated that average row-terminal discrimination accuracy consistently exceeds 98.17%, confirming high precision, strong adaptability, and stable performance in real trellised orchard scenarios. These results highlight that SkySeg-Net poses a robust and reliable perception capability, even in conditions where conventional methods often fail.

Although a full navigation-switching closed loop was not implemented, accurate row-terminal recognition is essential for practical autonomous orchard navigation. Specifically, it addresses the termination judgment problem in visual navigation, providing spatial

cues for coordinating multiple navigation sources. The end-of-row position indicates the transition between the in-row environment—where visual guidance dominates—and the open inter-row area—where GNSS/RTK guidance is more effective—supporting precise end-of-line steering and path switching.

Despite the promising results, several limitations of the proposed method should be acknowledged. First, SkySeg-Net assumes a relatively stable and forward-facing camera configuration with a slight upward field of view to ensure sufficient sky visibility. Extreme camera pitch angles or unstable mounting may affect the consistency of sky-region perception, although this issue can be mitigated through standardized camera installation and basic calibration procedures commonly adopted in agricultural robots. Second, in rare scenarios where the canopy is completely closed and the sky region is nearly invisible, the discriminative power of sky-based cues may be reduced. However, such conditions are uncommon in trellised orchards, especially near row terminals, which typically exhibit a clear structural opening. Third, missing plants or large gaps within a row may introduce abnormal sky-region expansion, potentially leading to false row-terminal indications. In practical systems, this limitation can be alleviated by integrating temporal consistency checks or complementary perception cues, such as row continuity or motion information, which are beyond the scope of the present study. Importantly, SkySeg-Net is not intended to serve as a complete navigation or control solution. Instead, it is designed as a robust and structurally grounded complementary perception cue for row-terminal recognition in trellised orchards. By providing reliable end-of-row awareness under canopy occlusion and illumination variation, the proposed method can support downstream navigation, decision-making, or multi-sensor fusion frameworks, while leaving full navigation strategy design to future work. By clearly acknowledging these constraints, the practical applicability of the proposed method is transparently communicated to readers, providing a balanced view of its strengths and limitations.

Overall, the proposed method enhances environmental perception in orchard automation systems, supports seamless transitions from in-row guidance to row-terminal decisions, and provides a practical foundation for future intelligent navigation strategies, such as sky-perception-informed visual-RTK mode switching. By explicitly addressing the limitations of traditional methods under dense canopy occlusion and complex visual conditions, this work contributes to improving the robustness and reliability of perception systems for autonomous agricultural robots.

5. Conclusions

This study built and manually labeled datasets of trellis vineyards covering different growth stages, enriching orchard field data and providing a foundation for developing adaptive navigation-switching strategies. To enable end-of-row detection, SkySeg-Net, a multi-scale U-Net, was proposed to segment the sky and background. The backbone's standard convolution module was replaced with a residual grouped convolution module to expand the receptive field and capture large-scale contextual information. The CBAM was added during downsampling to better extract key features from small receptive fields and strengthen global context awareness. During the upsampling process, a Pyramid Pooling Module was incorporated into the convolutional layer, improving segmentation accuracy, edge localization, and detail handling by fusing multi-resolution features. Experiments conducted across the fruit-setting, expansion, and maturity stages showed that the improved algorithm achieved an *mIoU* of 91.21%, an *mPA* of 94.82%, a precision of 95.80%, and a recall of 94.72%, outperforming other models and demonstrating robust sky segmentation under complex backgrounds. The average recognition rate at the end of the row reached 98.17%, confirming the method's feasibility across growth stages.

Challenges remain, including variations in orchard lighting, seasonal differences in vegetation density, and canopy occlusion. Future work will focus on two main areas: (1) improving model robustness to adapt to complex environments and exploring lightweight networks or optimization methods suitable for embedded devices and (2) leveraging the reliable sky segmentation results from this study to develop efficient and safe navigation mode-switching strategies. This will involve defining sky-area criteria (size, location, and continuity), designing smooth-switching algorithms, and conducting comprehensive validation on a robot platform integrating vision, RTK, and inertial navigation. These efforts aim to achieve a fully robust autonomous navigation system for orchards under diverse conditions.

Author Contributions: Conceptualization, H.G. and C.G.; methodology, H.G. and Y.W.; software, H.G.; validation, H.G., and Y.W.; formal analysis, H.G. and H.L.; investigation, H.G. and Y.W.; resources, C.G., Y.S. and T.T.; writing—original draft preparation, Y.W. and H.G.; writing—review and editing, Y.W. and H.L.; visualization, H.G.; supervision, C.G. and Y.S.; project administration, H.G. and T.T.; funding acquisition, Y.S. and T.T.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Innovation Program of the Chinese Academy of Agricultural Sciences (CAAS-CAE-202302 and CAAS-CAE-202301) and the Open Research Project of the Key Laboratory of Applied Research on Tropical Crop Information Technology of Hainan Province (ZDSYS-KFJJ-202503).

Data Availability Statement: Due to our laboratory’s policies or confidentiality agreements, we cannot provide raw data.

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

Abbreviations

The following abbreviations are used in this manuscript:

RTK	Real-time kinematic
BBCH	Biologische Bundesanstalt, Bundessortenamt und Chemical industry
U-Net	Convolutional Networks for Biomedical Image Segmentation
GNSS	Global Navigation Satellite System
LiDAR	Light Detection and Ranging
SLAM	Simultaneous Localization And Mapping
YOLOv5/7	You Only Look Once version 5/7
RT-DETR	Real-Time Detection Transformer
UAV	Unmanned Aerial Vehicle
DeepLabv3+	Encoder–Decoder with Atrous Separable Convolution for Semantic Image Segmentation
<i>mIoU</i>	Mean Intersection over Union
Mask R-CNN	Mask Region-based Convolutional Neural Network
ROI	Region of Interest
JSON	JavaScript Object Notation
ResNeSt	Split-Attention Networks
ResNet	Residual Network
MBAM	Multi-branch Attention Module
CBAM	Convolutional Block Attention Module
PPM	Spatial Pyramid Pooling
BCEWithLogitsLoss	Binary Cross Entropy With Logits Loss
CPU	Central Processing Unit
GPU	Graphics Processing Unit
RAM	Random Access Memory

CUDA	Compute Unified Device Architecture
mPA	Mean Pixel Accuracy
IoU	Intersection over Union
TP	The Proportion of True Cases
FP	False Positive Cases
FN	False Negative Cases
TN	True Negative Cases

References

- Li, Y.; Feng, Q.; Ji, C.; Sun, J.; Sun, Y. GNSS and LiDAR Integrated Navigation Method in Orchards with Intermittent GNSS Dropout. *Appl. Sci.* **2024**, *14*, 3231. [\[CrossRef\]](#)
- Liu, H.; Du, Z.; Shen, Y.; Du, W.; Zhang, X. Development and evaluation of an intelligent multivariable spraying robot for orchards and nurseries. *Comput. Electron. Agric.* **2024**, *222*, 109056. [\[CrossRef\]](#)
- Su, Z.; Zou, W.; Zhai, C.; Tan, H.; Yang, S.; Qin, X. Design of an Autonomous Orchard Navigation System Based on Multi-Sensor Fusion. *Agronomy* **2024**, *14*, 2825. [\[CrossRef\]](#)
- Winterhalter, W.; Fleckenstein, F.; Dornhege, C.; Burgard, W. Localization for precision navigation in agricultural fields—Beyond crop row following. *J. Field Robot.* **2021**, *38*, 429–451. [\[CrossRef\]](#)
- De Silva, R.; Cielniak, G.; Gao, J. Vision based crop row navigation under varying field conditions in arable fields. *Comput. Electron. Agric.* **2024**, *217*, 108581. [\[CrossRef\]](#)
- Shah, H.N.M.; Lafidi, N.S.; Abdollah, M.F.; Ahmad, A.; Ab Rashid, M.Z.; Arshad, M.A. Vision Based Row Guidance Approach for Navigation of Agricultural Mobile Robots in Orchards. *Mod. Appl. Sci.* **2024**, *18*, 1–60.
- Peng, C.; Fei, Z.; Vougioukas, S.G. Depth camera based row-end detection and headland maneuvering in orchard navigation without GNSS. In Proceedings of the 2022 30th Mediterranean Conference on control and Automation (MED), Vouliagmeni, Greece, 28 June–1 July 2022; pp. 538–544.
- Peng, C.; Fei, Z.; Vougioukas, S.G. GNSS-free end-of-row detection and headland maneuvering for orchard navigation using a depth camera. *Machines* **2023**, *11*, 84. [\[CrossRef\]](#)
- 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. [\[CrossRef\]](#)
- Xia, Y.; Lei, X.; Pan, J.; Chen, L.; Zhang, Z.; Lyu, X. Research on orchard navigation method based on fusion of 3D SLAM and point cloud positioning. *Front. Plant Sci.* **2023**, *14*, 1207742. [\[CrossRef\]](#)
- Anagnostis, A.; Tagarakis, A.C.; Kateris, D.; Moysiadis, V.; Sørensen, C.G.; Pearson, S.; Bochtis, D. Orchard mapping with deep learning semantic segmentation. *Sensors* **2021**, *21*, 3813. [\[CrossRef\]](#)
- Gao, Y.; Tian, G.; Gu, B.; Zhao, J.; Liu, Q.; Qiu, C.; Xue, J. A Study on the Rapid Detection of Steering Markers in Orchard Management Robots Based on Improved YOLOv7. *Electronics* **2023**, *12*, 3614. [\[CrossRef\]](#)
- Gu, B.; Liu, Q.; Gao, Y.; Tian, G.; Zhang, B.; Wang, H.; Li, H. Research on the Relative Position Detection Method between Orchard Robots and Fruit Tree Rows. *Sensors* **2023**, *23*, 8807. [\[CrossRef\]](#) [\[PubMed\]](#)
- Kang, H.; Chen, C.J.C. Fruit detection, segmentation and 3D visualisation of environments in apple orchards. *Comput. Electron. Agric.* **2020**, *171*, 105302. [\[CrossRef\]](#)
- Wu, T.; Guo, H.; Wang, X.; Zhou, W.; Gao, G.; Yang, C.; Zhou, W. Real-Time Navigation Line Extraction for Safflower Harvesting Robots Using an Improved Detection Transformer. *IEEE Sens. J.* **2025**, *25*, 22941–22955. [\[CrossRef\]](#)
- Xu, Y.; Xue, X.; Sun, Z.; Gu, W.; Cui, L.; Jin, Y.; Lan, Y. Global path planning for navigating orchard vehicle based on fruit tree positioning and planting rows detection from UAV imagery. *Comput. Electron. Agric.* **2025**, *236*, 110446. [\[CrossRef\]](#)
- Xue, X.; Luo, Q.; Bu, M.; Li, Z.; Lyu, S.; Song, S. Citrus tree canopy segmentation of orchard spraying robot based on RGB-D image and the improved DeepLabv3+. *Agronomy* **2023**, *13*, 2059. [\[CrossRef\]](#)
- Zhang, J.; Tian, M.; Yang, Z.; Li, J.; Zhao, L. An improved target detection method based on YOLOv5 in natural orchard environments. *Comput. Electron. Agric.* **2024**, *219*, 108780. [\[CrossRef\]](#)
- Zhang, W.; Chen, X.; Qi, J.; Yang, S. Automatic instance segmentation of orchard canopy in unmanned aerial vehicle imagery using deep learning. *Front. Plant Sci.* **2022**, *13*, 1041791. [\[CrossRef\]](#)
- Häni, N.; Roy, P.; Isler, V. MinneApple: A benchmark dataset for apple detection and segmentation. *IEEE Robot. Autom. Lett.* **2020**, *5*, 852–858. [\[CrossRef\]](#)
- Wenzel, P.; Yang, N.; Wang, R.; Zeller, N.; Cremers, D. 4seasons: Benchmarking visual slam and long-term localization for autonomous driving in challenging conditions. *Int. J. Comput. Vis.* **2025**, *133*, 1564–1586. [\[CrossRef\]](#)
- Kusrini, K.; Suputa, S.; Setyanto, A.; Agastya, I.M.A.; Priantoro, H.; Chandramouli, K.; Izquierdo, E. Data augmentation for automated pest classification in Mango farms. *Comput. Electron. Agric.* **2020**, *179*, 105842. [\[CrossRef\]](#)

23. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 60. [\[CrossRef\]](#)
24. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
25. Zhang, H.; Wu, C.; Zhang, Z.; Zhu, Y.; Lin, H.; Zhang, Z.; Sun, Y.; He, T.; Mueller, J.; Manmatha, R. Resnest: Split-attention networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 2736–2746.
26. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
27. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
28. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [\[CrossRef\]](#)
29. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
30. Alom, M.Z.; Hasan, M.; Yakopcic, C.; Taha, T.M.; Asari, V.K. Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation. *arXiv* **2018**, arXiv:1802.06955.
31. Oktay, O.; Schlemper, J.; Folgoc, L.L.; Lee, M.; Heinrich, M.; Misawa, K.; Mori, K.; McDonagh, S.; Hammerla, N.Y.; Kainz, B.; et al. Attention u-net: Learning where to look for the pancreas. *arXiv* **2018**, arXiv:1804.03999. [\[CrossRef\]](#)
32. Zuo, Q.; Chen, S.; Wang, Z. R2AU-Net: Attention recurrent residual convolutional neural network for multimodal medical image segmentation. *Secur. Commun. Netw.* **2021**, *2021*, 6625688. [\[CrossRef\]](#)
33. Zhou, Z.; Rahman Siddiquee, M.M.; Tajbakhsh, N.; Liang, J. Unet++: A nested u-net architecture for medical image segmentation. In Proceedings of the International Workshop on Deep Learning in Medical Image Analysis, Granada, Spain, 20 September 2018; pp. 3–11.
34. Wang, J.; Sun, K.; Cheng, T.; Jiang, B.; Deng, C.; Zhao, Y.; Liu, D.; Mu, Y.; Tan, M.; Wang, X.; et al. Deep high-resolution representation learning for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 3349–3364. [\[CrossRef\]](#)
35. Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 801–818.

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.