Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Method for 3D Morphological Measurement and Cutting Point Localization of Green Asparagus Based on Improved YOLOv8n-seg and RGB-D Data

qingqi lu
Jiangxi Academy of Agricultural Sciences
yuliang zhong
Jiangxi Academy of Agricultural Sciences
jinsong zhou
Jiangxi Academy of Agricultural Sciences
xiaobo su
m177791165700163.com

Jiangxi Academy of Agricultural Sciences
jian ding
Jiangxi Academy of Agricultural Sciences
ruopeng du
Chinese Academy of Agricultural Sciences
hui hong
Jiangxi Academy of Agricultural Sciences

Article

Keywords: Green Asparagus, Automated Harvesting, YOLOv8n-seg, Lightweight, Cutting Point Localization

Posted Date: November 4th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11311204/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A Method for 3D Morphological Measurement and Cutting Point Localization of Green Asparagus Based on Improved YOLOv8n-seg and RGB-D Data

Qingqi Lu[1], Yuliang Zhong[1], Jinsong Zhou[2], Xiaobo Su[1,*], Jian Ding[1], Ruopeng Du[3,*], Hui Hong[1]

[1]Institute of Agricultural Economics and Information, Jiangxi Academy of Agricultural Sciences, Nanchang 330200, China

[2]Institute of Vegetables and Flowers, Jiangxi Academy of Agricultural Sciences, Nanchang 330200, China

[3]Agricultural information Institute, Chinese Academy of Agricultural Sciences /Key Laboratory of Knowledge Mining and Knowledge Services in Agricultural Converging Publishing, National Press and Publication Administration, Beijing 100081, China

Abstract

To address the difficulties in recognition and localization caused by the curved and irregular growth forms of green asparagus in natural environments, with parts of the rootstalk hidden in soil, and the challenge of balancing detection accuracy and speed with traditional instance segmentation algorithms during automated harvesting operations, a lightweight green asparagus instance segmentation model named SES-YOLO based on improved YOLOv8n-seg is proposed. First, the StarNet lightweight backbone network is used to replace the original backbone structure, reducing the computational burden of the model. Second, the C2fDWConv_Strip module is designed by integrating depthwise separable convolution and strip convolution to enhance the feature extraction capability for slender targets. Then, the EGAP module is embedded into the SPPF layer to improve edge detail perception and anti-interference capability, and the MPDIoU loss function is adopted to optimize bounding box regression accuracy. Finally, a method combining image masks with depth information to calculate the length and diameter of green asparagus and locate the cutting point is employed. Experimental results show that SES-YOLO achieves precision, recall, and mAP@50 of 98.6%, 97.3%, and 99.3%, respectively. Compared to the baseline model, the parameter count is reduced by 38.1%, and the computational load is decreased by 22.3%. The average relative measurement errors for asparagus length and diameter are 1.97% and 2.07%, respectively, and the average cutting point localization error is 0.34 cm. The model's inference speed on the Jetson Orin NX edge device is 38.69 FPS, which is 58.4% higher than the baseline model. This model maintains high accuracy while possessing good lightweight and real-time performance, making it suitable for embedded deployment scenarios such as green asparagus harvesting robots, providing reliable technical support for achieving automated selective harvesting.

Keywords: Green Asparagus; Automated Harvesting; YOLOv8n-seg; Lightweight; Cutting Point Localization

Introduction

Asparagus is a perennial erect herbaceous plant rich in various trace elements and amino acids essential for the human body [1-2]. China, as the world's largest producer, processor, and exporter of asparagus, faces challenges in its harvesting. Currently, asparagus harvesting primarily relies on manual labor. However, due to the growth cycle characteristics of asparagus, multiple harvests are required during the maturity period, resulting in high labor intensity and elevated manual harvesting costs. With the continuous development and mature application of automated robotics technology in agriculture, many researchers have combined deep learning detection models with automated robotics to replace human labor in agricultural activities, giving rise to automated asparagus harvesting robots[3]. However, in natural growth environments, asparagus grows densely with overlapping stalks, exhibits irregular shapes, and has parts of its rootstalk concealed within the soil. Therefore, the core challenge for an asparagus harvesting robot lies in achieving rapid, accurate detection of mature asparagus shape features and precise localization of the harvesting point.

With the advancement of computer vision and deep learning technologies, computer vision has been increasingly applied by scholars to address the challenges of acquiring key information from crops [4-6]. In recent years, due to their low cost and high efficiency, RGB-D sensors have been widely adopted by researchers for collecting key crop information [7-9]. In the measurement of crop length and volume, although deep learning-based object detection algorithms can identify regions of interest in crops within complex environments [10-11], they still rely on image processing techniques to further extract crop contour information. This leads to increased algorithm complexity, potential introduction of errors, and consequently a reduction in overall detection efficiency and accuracy. Deep learning-based instance segmentation algorithms can simultaneously achieve precise extraction of both crop regions of interest and contour information, significantly reducing algorithm complexity compared to the former and facilitating subsequent analytical work. Gené-Mola et al. used RGB-D data combined with a convolutional neural network for instance segmentation of apples to estimate diameter [12]. Rong et al. utilized an instance segmentation algorithm to demonstrate the effectiveness of accurately identifying and matching tomato organs in real-world scenarios [13]. Kim et al. performed size estimation of leaf-occluded oriental melons in vertical cultivation systems based on instance segmentation and de-occlusion models [14].

Image semantic segmentation can accurately segment pixels, separate different parts of the target and background, and obtain the specific contour of the target [15]. Unlike semantic segmentation, image instance segmentation can accurately segment the pixels of each individual object, not only identifying targets in the image but also precisely obtaining the contour boundaries of each target. Wang et al. proposed an improved Mask-RCNN-based instance segmentation method, integrating deformable convolution with a Transformer attention module in the backbone network to achieve accurate segmentation of apples under different states. This method achieved excellent performance with a recall of 97.1%, precision of 95.8%, and a segmentation mAP of 0.917 [16]. Yu et al., based on Mask-RCNN, utilized ResNet50 as the backbone network combined with a Feature Pyramid Network (FPN) for feature extraction to generate mask images of mature strawberry fruits. Prediction results for 573 mature strawberry fruit picking points showed an average localization error of ±1.2 mm, demonstrating stronger generalization and robustness [17]. The aforementioned algorithms

are two-stage methods, possessing high accuracy but with the drawback of slower speed. The YOLO series algorithms are single-stage methods, achieving an excellent balance between accuracy and speed, thus attracting significant attention from researchers [18-19]. YOLACT innovatively decomposed the instance segmentation task into two parallel sub-tasks, achieving high-quality object detection and segmentation, and is an excellent single-stage algorithm [20]. Inspired by YOLACT, similar segmentation methods have been integrated into YOLO [21], making its speed close to YOLACT and much faster than Mask R-CNN, while its accuracy is very close to Mask R-CNN, achieving an optimal balance.

To date, an increasing number of researchers have improved and optimized YOLO-based instance segmentation models for specific application scenarios, conducting extensive work. Mu et al. used an RGB-D camera and an improved YOLOv8 model to identify and localize green asparagus, achieving relative errors of less than 2.98% and 2.15% for asparagus length and diameter measurements, respectively [22]. Chen et al., based on a YOLOv9 model and point cloud data acquired from a ZED camera, performed recognition, detection, and instance segmentation of asparagus, proposing a dynamic scaling factor method to calculate the actual asparagus length and harvesting point position. The mean absolute error for length measurement was 0.9 cm, and the mean absolute error for the harvesting point in the x-axis direction was 0.15 cm [23]. Li et al. used an improved YOLOv8n model for object detection of mango fruits and segmentation of the stem region of interest (ROI), developing a picking point localization algorithm to determine the coordinates of the optimal picking point, achieving an average localization success rate of 92.01% [24]. Qi et al., based on an improved YOLOv8-seg algorithm combined with RGB-D data, obtained masks and detection boxes for tomato buds and stems, using the instance segmentation results and depth information to calculate the main stem diameter of tomatoes. The measurement results showed an RMSE of 1.5 mm and a MAPE of 12.37% [25]. Shen, based on an improved lightweight YOLOv8 instance segmentation model, segmented the region of interest (ROI) of grape pedicels, achieving a mAP@50 of 99.5%, significantly reducing GFLOPs, with a model parameter count of only 2.3M [26]. Paul et al. used a RealSense D455 camera and a YOLOv8s model for detection and segmentation of chili peduncles, achieving box and mask mAP values of 0.790 and 0.771, respectively [27].

The aforementioned studies have promoted the development of instance segmentation technology in specific scenarios, but the following problems remain: detection models have not achieved a balance between detection accuracy and speed, resulting in high-accuracy methods being slow and fast methods having unsatisfactory accuracy. For the task of automated green asparagus harvesting, the growth morphology of green asparagus varies, planting density is high, and overlap and occlusion problems exist. Furthermore, the model needs to be deployed on edge devices with limited performance. Real-time and accurate detection of green asparagus in complex planting environments requires the model to achieve both high accuracy and low complexity; otherwise, harvesting operations are difficult to perform.

To solve the aforementioned problems, this study, based on the YOLOv8n-seg model, proposes a lightweight green asparagus instance segmentation model named SES-YOLO. Through improvements to the backbone network, SPPF layer, C2F layer, and loss function, the model's complexity is reduced, achieving lightweight design while improving detection

accuracy for green asparagus. This achieves a balance between detection accuracy and speed, enabling deployment on edge devices for precise segmentation and cutting point localization of green asparagus.

Methodology

This study proposes a method for stem length and diameter measurement and harvest point localization for the selective harvesting of green asparagus. The specific workflow is illustrated in Fig. 1. First, images of green asparagus are acquired. The SES-YOLO instance segmentation algorithm is then employed to obtain the corresponding mask images. Subsequently, image morphological processing and skeleton fitting are performed. Combined with the two-dimensional parameters derived from the mask images, the pixel length and diameter of the green asparagus are calculated. Finally, the coordinate system of the Intel RealSense D435i depth camera is utilized to compute the true stem length and diameter and to localize the harvest point for pickable green asparagus. The specific details of each process are described in the following sections.

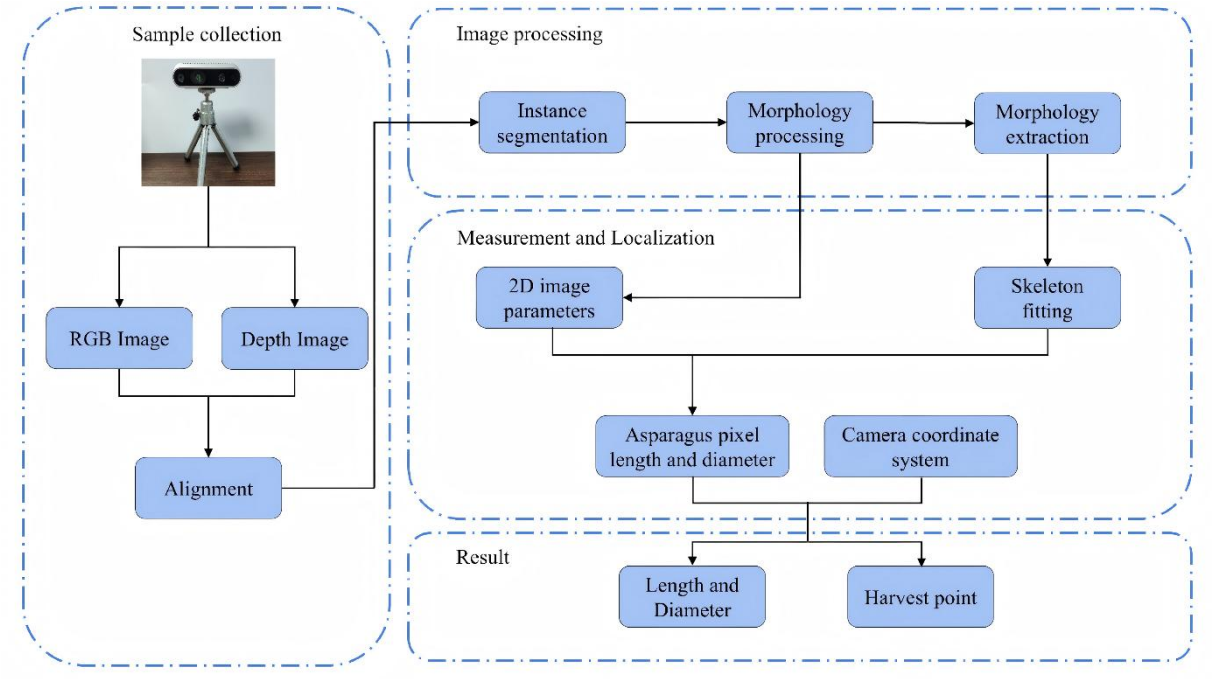


Fig. 1. Visual recognition flowchart for green asparagus.

Data Acquisition

The green asparagus samples used in this study were collected from Mengfei Pastoral Farm (28° 37′ N, 115° 39′ E) in Shibu Town, Xinjian District, Nanchang City, Jiangxi Province, China. An Intel RealSense D435i depth camera was employed as the acquisition device, with an RGB resolution of 1920×1080 at 30 FPS and a depth resolution of 640×480 at 30 FPS. During data collection, the depth camera was mounted on one side of the asparagus planting ridge, approximately 30 cm above the ridge surface, with an imaging distance ranging from 400 mm to 800 mm. Shooting was conducted under varying natural light

intensities in the morning and afternoon. The lens was adjusted under natural lighting conditions to capture images from multiple angles, including level and top-down perspectives. A total of 1058 raw images of green asparagus were acquired, as shown in Fig. 2a and Fig. 2b.



Fig. 2. Sample images of green asparagus. (a) An image captured under sufficient morning light. (b) An image captured under weaker afternoon light.

Dataset Construction and Data Augmentation

The green asparagus image data were annotated using LabelMe to form the original green asparagus image dataset, which was used for model training to generate asparagus masks. The original dataset was split into training and validation sets in a 9:1 ratio, with 953 images allocated for training and 105 images for validation.

To enhance the robustness and generalization capability of the model in green asparagus recognition and segmentation, offline and online data augmentation methods were applied to expand the original dataset. Offline data augmentation primarily involved seven techniques, including random brightness adjustment, horizontal flipping, motion blur, random rotation, and noise addition, collectively generating 7406 images. The augmentation effects are illustrated in Fig. 3. Online data augmentation mainly employed the Mosaic method, with a probability set to 1.0 per epoch, and its application was discontinued during the final 10 epochs of the total epoch count.

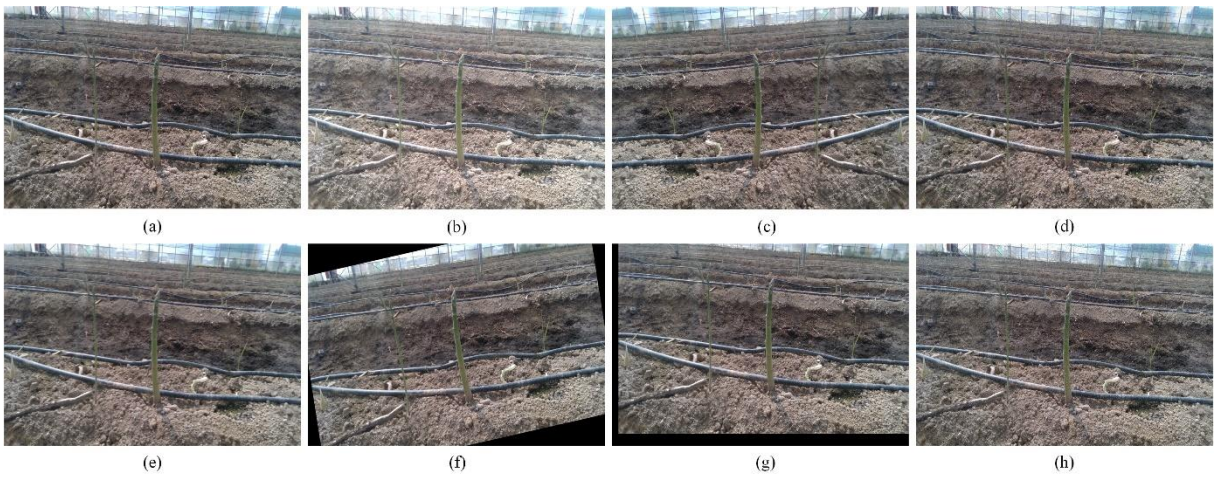


Fig. 3. Image augmentation effects: (a) original image, (b) random brightness adjustment, (c) horizontal flipping, (d) Gaussian noise, (e) motion blur, (f) random rotation, (g) random translation, (h) salt and pepper

noise.

Improved YOLOv8n Instance Segmentation Method

YOLOv8n is an efficient object detection neural network model developed by Ultralytics. Built upon a single-stage detector with the addition of an instance segmentation branch, it can simultaneously accomplish object detection and instance segmentation tasks with high performance and low latency, and has been widely adopted in industrial and agricultural applications. The overall architecture of YOLOv8n consists of a backbone network, a neck network, and a head network. Compared to YOLOv5, the C3 module in the backbone and neck networks is replaced by the C2F module, the anchor-based detector is modified to anchor-free, and a DFL module is incorporated to enable faster and more accurate bounding box prediction. The head network is redesigned with a decoupled structure, enhancing the model's capability for feature representation learning and stability.

Although YOLOv8n possesses strong generalization capabilities, it remains inadequate for specific application scenarios such as green asparagus recognition. For instance, its feature extraction ability struggles with the complex growing environment of green asparagus. Furthermore, when deployed on automated harvesting robots or other resource-constrained edge devices, its lightweight level and inference speed fail to meet deployment demands and real-time requirements. To address these issues, this study selects YOLOv8n-seg as the baseline model for improvement and performance comparison. Enhancements are made in the following four aspects. The architecture of the improved model is shown in Fig. 4.

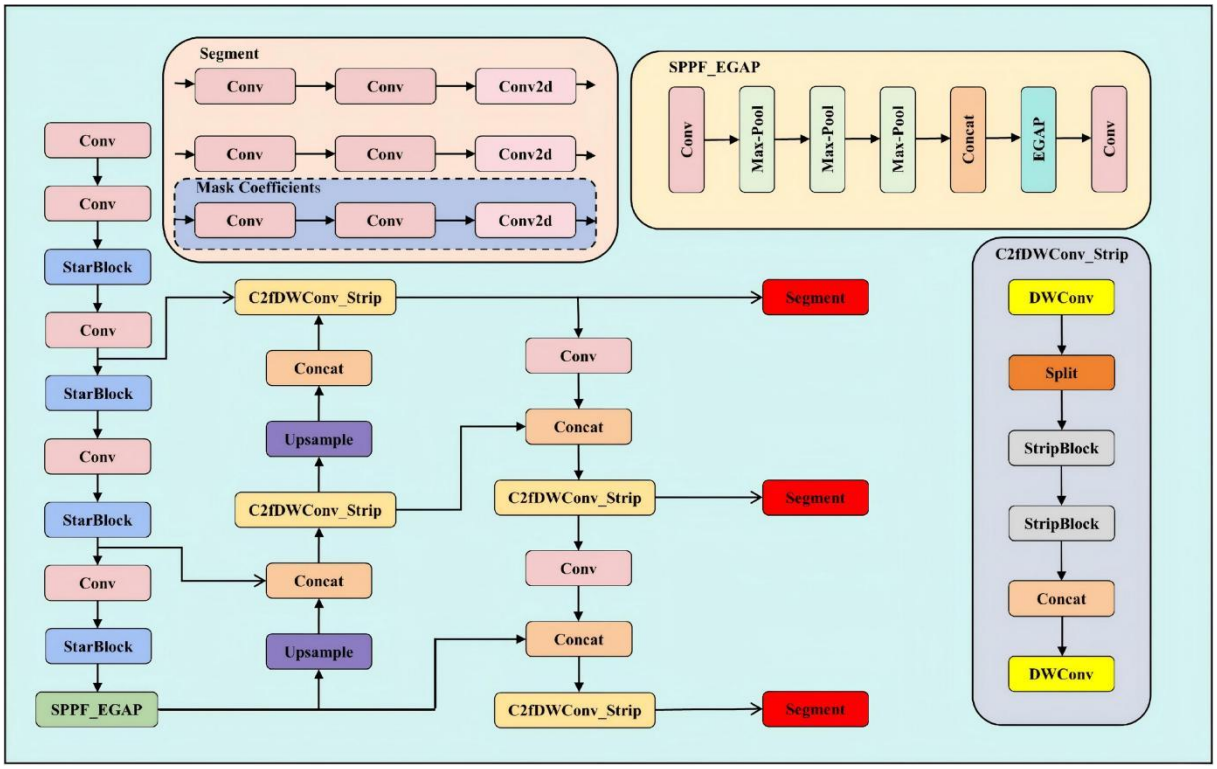


Fig. 4. The model architecture of SES-YOLO.

Improvement of the Backbone Network

This study introduces the more efficient backbone network StarNet [28] to replace the original backbone in the baseline YOLOv8n model. StarNet contains multiple feature extraction stages. Each stage employs a convolutional layer to progressively reduce the resolution of feature maps and double the number of channels, and repetitively uses multiple StarBlocks for feature extraction, enabling efficient feature extraction performance. By combining the star operation (element-wise multiplication) and depthwise separable convolution (DWConv), StarNet maps inputs into a high-dimensional nonlinear feature space, achieving efficient feature representation under a compact network structure and lower energy consumption without increasing computational complexity. This design grants StarNet low-latency characteristics, providing potential for application in scenarios with high real-time requirements.

As the core module of StarNet, StarBlock first adopts a DWConv layer with a kernel size of 7×7. The large 7×7 kernel ensures a sufficient spatial receptive field, effectively capturing broader contextual information and enhancing feature representation capability. The design of DWConv reduces computational complexity. The combination of both allows for efficient acquisition of local spatial features from the input feature map. The specific process is shown in Equation (1).

$$y_{dw} = d(y) \tag{1}$$

where y represents the input feature map, $d(\cdot)$denotes the DWConv computation function, and y_{dw} represents the feature map obtained after DWConv computation.

The feature map output from the DWConv is passed through two fully connected (FC) layers, which expand the number of feature channels and map the input features into two different feature spaces. The ReLU6 activation function is then applied to control the output, and element-wise multiplication is used for nonlinear interaction, thereby enhancing feature expression capability and suppressing redundant information. The specific process is shown in the following equations.

$$f_1 = FC1(y_{dw}) \tag{2}$$

$$f_2 = FC2(y_{dw}) \tag{3}$$

$$z = ReLU6(f_1) \odot f_2 \tag{4}$$

where $FC1$ and $FC2$ are two fully connected layers, f_1 and f_2 are the feature maps after passing through the two fully connected layers, ReLU6 represents the computation process of the activation function, $\odot$ denotes the element-wise multiplication operation, and z represents the feature map after feature fusion via the activation function.

A 1×1 convolution is then used to reduce the number of feature channels back to the original number, followed by a second 7×7 DWConv to further optimize spatial feature representation. To enhance the generalization capability of the model, StarBlock employs a random drop path (DropPath) mechanism, which reduces overfitting by randomly dropping residual branches. This mechanism prevents the model from over-relying on specific paths, ensuring high generalization ability and robustness. The specific process is shown in the following equations.

$$y_1 = Conv_{1\times1}(z) \tag{5}$$

$$y_2 = d(y_1) \tag{6}$$

$$y_{out} = y + DropPath(y_2) \quad (7)$$

where $Conv_{1\times1}$ is the computation process of the 1×1 convolution, y_1 is the feature map obtained after the 1×1 convolution, y_2 is the feature map obtained after the 7×7 DWConv, and $DropPath$ represents the random drop path mechanism.

StarBlock enhances the modeling accuracy of local features of green asparagus while maintaining low computational overhead, enabling the network to retain strong representational learning capability under lightweight conditions. The structure of the StarBlock module is shown in Fig. 5.

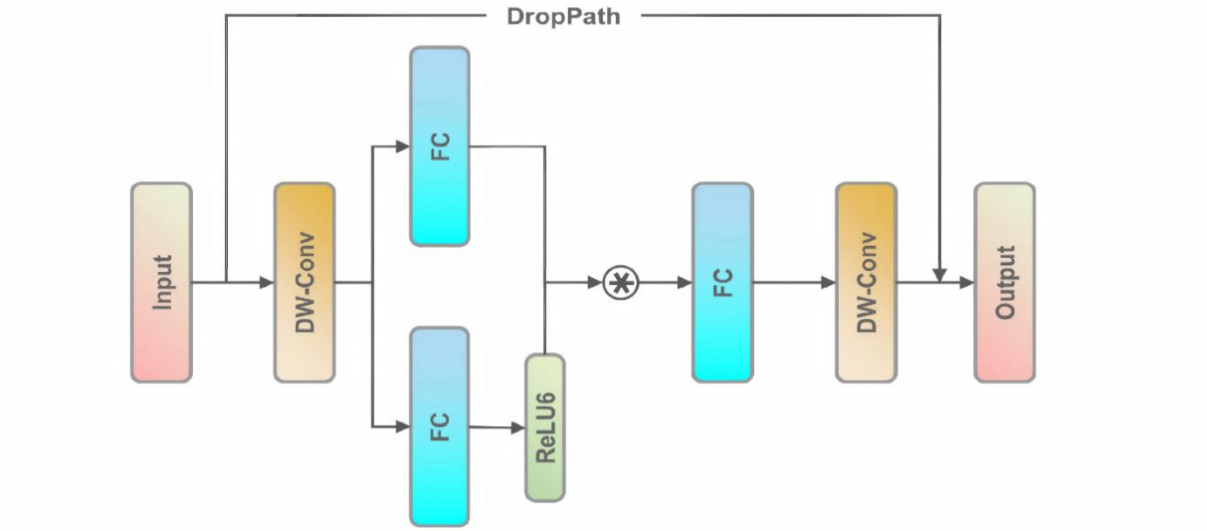


Fig. 5. StarBlock module.

C2fDWConv_Strip

The core component C2f module in the YOLOv8 model effectively enhances the model's feature representation capability by introducing rich gradient flow and feature reuse. However, the C2f module contains a substantial number of standard convolution computations, which increases redundant calculations in the model, hindering its lightweight design and deployment on edge computing platforms. Furthermore, the traditional Bottleneck structure has limitations in capturing global contextual information, especially for targets like green asparagus that exhibit a significant aspect ratio, as the spatial receptive field of the Bottleneck is constrained. To address these issues, this study introduces DWConv [29] to replace the standard convolution in the C2f module. Additionally, based on the Strip Module [30], an improved StripBlock is proposed to replace the Bottleneck. Specific details are described below.

Traditional standard convolution involves computational redundancy when processing spatial and channel dimensions, limiting its application in resource-constrained environments. To improve model efficiency, this study employs DWConv to replace the standard convolution in C2fDWConv significantly reduces computational complexity while maintaining the model's representational capacity, achieving a better balance between accuracy and efficiency. As illustrated in Fig. 6(a) and Fig. 6(b), DWConv decomposes the standard convolution into two independent operations: depthwise convolution and pointwise convolution. Depthwise convolution independently processes spatial information for each channel, preserving the integrity of local features. Pointwise convolution is specifically

responsible for information interaction between channels, enabling more efficient feature combination.

Although the use of DWConv reduces computational redundancy and decreases the model's parameter count to some extent, there remains room for improvement in the model's lightweight design. Moreover, the use of DWConv may lead to a slight loss in accuracy. To further optimize the lightweight design of the model and improve the accuracy of asparagus-specific recognition, this study employs StripBlock to replace the Bottleneck. First, StripBlock adopts a 5×5 DWConv as the feature extractor, balancing receptive field size and computational efficiency. It then uses strip convolutions with kernel sizes of 1×19 and 19×1 to capture the rich spatial information of green asparagus. For targets like green asparagus with a significant aspect ratio, strip convolutions can better capture their spatial features, demonstrating strong sensitivity to the edges and contours of green asparagus. Furthermore, under similar receptive fields, strip convolutions have fewer parameters than standard large-kernel convolutions, significantly improving computational efficiency. The first strip convolution with a kernel size of 1×19 focuses on capturing long-range dependencies in the horizontal direction, while the second strip convolution with a kernel size of 19×1 focuses on capturing long-range dependencies in the horizontal direction. Subsequently, a 1×1 convolutional layer is used to enable feature interaction between channels and adjust dimensions. Finally, the $ReLU6$ activation function is applied to control the output, and element-wise multiplication is utilized for nonlinear interaction, enhancing the expression of important features while suppressing irrelevant ones. The relevant computational process is as follows.

$$x_{temp} = conv_{1\times1}(conv_{(19,1)}(conv_{(1,19)}(conv_{5\times5}^{dw}(x)))) \tag{8}$$

$$y = ReLU6(x_{temp}) \odot x \tag{9}$$

Where x is the feature input, $conv_{5\times5}^{dw}$ denotes the computation process of the 5×5 DWConv, $conv_{(1,19)}$ and $conv_{(19,1)}$ represent the computation processes of strip convolutions with kernel sizes of 1×19 and 19×1, respectively, $conv_{1\times1}$ denotes the computation process of the 1×1 convolution, $ReLU6$ represents the computation process of the activation function, and $\odot$ denotes the element-wise multiplication operation. The structure of the StripBlock model is shown in Fig. 6(c).

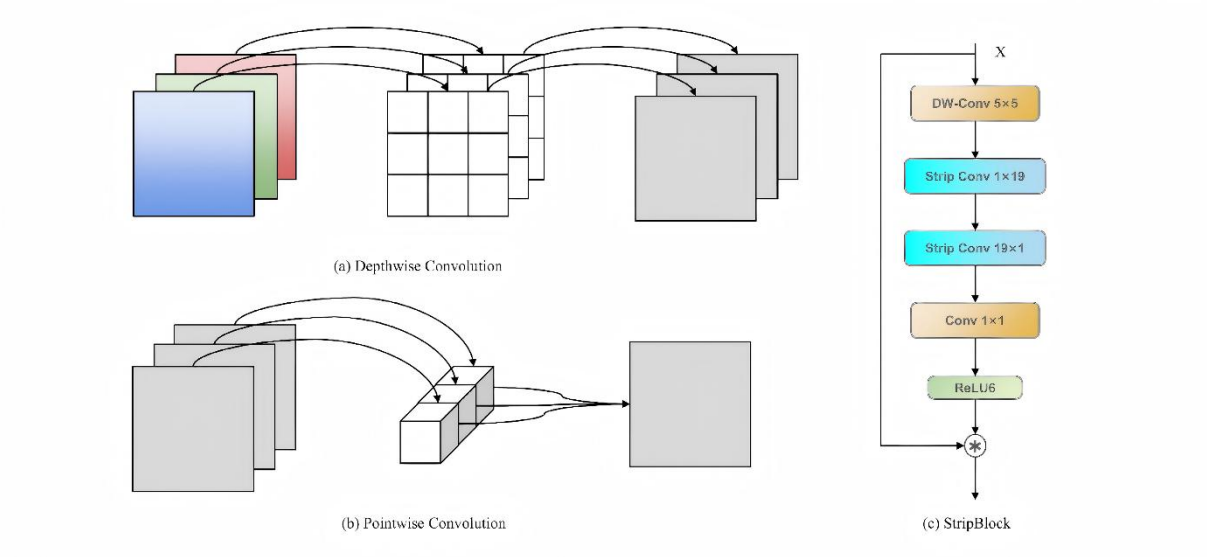


Fig. 6. (a) Depthwise convolution operation. (b) Pointwise convolution operation. (c) Structure of the StripBlock.

EGAP Module

The actual planting and growing environment of green asparagus is often complex, involving issues such as low environmental light intensity and occlusion. Furthermore, in automated asparagus harvesting scenarios, factors like terrain, temperature, and noise interference may cause RGB cameras to capture low-quality images, often resulting in blurred features and unclear boundaries of green asparagus. To address these problems, this study employs the Edge-Gaussian Aggregation Precision (EGAP) module, which enables the model to more accurately capture the edge features of green asparagus and guides the network to achieve better image segmentation results. The structure of the EGAP module is shown in Fig. 7.

The EGAP module first utilizes a Gaussian prior-based feature extraction component to model blurred features. A 5×5 fixed-parameter Gaussian convolution kernel is applied to smooth each channel, helping to suppress irrelevant background noise interference. Meanwhile, leveraging the property of Gaussian distribution to emphasize central pixels, it highlights the salient regions of green asparagus objects and enhances edge details of green asparagus, addressing the feature response challenges posed by blur and noise. The specific process is shown in the following equation.

$$F_{gauss} = G_{1.0}^{5\times5}(F_{in}) \tag{10}$$

where F_{in} is the input feature, $G_{1.0}^{5\times5}$ represents the processing with a fixed 5×5 Gaussian convolution kernel, and F_{gauss} is the feature output after Gaussian processing.

After the EGAP module, the ReLU activation function is applied to control the output. Then, an efficient channel attention (ECA) mechanism[31] incorporating lightweight 1D convolution is adopted to adaptively adjust channel response intensity. This enhances important edge feature channels, reduces responses from noisy or unimportant channels, and improves overall semantic discriminative capability, achieving significant performance gains at minimal computational cost. Subsequently, the initial input features are fused with the features processed by ECA. Finally, the fused features are passed through a three-layer convolutional block to enhance feature representation. The specific process is shown in the

following equations.

$$F_{relu} = ReLU(F_{gauss}) \tag{11}$$

$$F_{eca} = ECA(F_{relu}) \tag{12}$$

$$F_{temp} = F_{eca} + F_{in} \tag{13}$$

$$F_{egap} = Conv_{3\times3}(F_{temp}) \tag{14}$$

where $ReLU$ denotes the computation process of the activation function, ECA represents the computation process of the efficient channel attention mechanism, and $Conv_{3\times3}$ denotes the computation process of the three-layer convolution.

This study proposes an improved SPPF layer, termed SPPF_EGAP, placed at the end of the backbone network. Its structure is shown in the fig. 4. SPPF_EGAP combines the advantages of SPPF and EGAP, demonstrating stronger capability in extracting features of green asparagus. It helps the model more clearly identify the edges of green asparagus, improves the segmentation accuracy of the model for green asparagus, and maintains high computational efficiency.

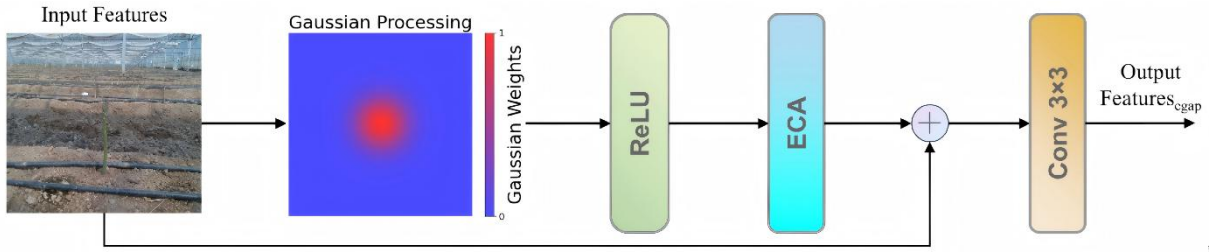


Fig. 7. EGAP module.

Improvement of the Loss Function

The design of the bounding box regression loss function is crucial for the recognition, localization, and segmentation of green asparagus. The CIoU loss function is used in the original YOLOv8 model. It achieves higher prediction accuracy than the IoU, GIoU, and DIoU loss functions by comprehensively optimizing three key factors: the overlap area between the predicted and ground truth boxes, the distance between their centers, and their aspect ratio. However, since green asparagus has a large aspect ratio, the aspect ratio penalty term in CIoU struggles to accurately describe this extreme proportional relationship. Additionally, the curved growth of green asparagus leads to variable bounding box orientations. The center-point distance metric in CIoU cannot fully reflect the actual positional deviation, which can reduce model convergence efficiency and cause issues such as missed detections and frame distortion. To address these problems, this study introduces the MPDIoU loss function [32] to replace CIoU. The MPDIoU loss function fully considers factors such as the target's size, degree of overlap, center-point distance, and deviations in height and width, while simultaneously optimizing the distance between the two key points of the bounding box: the top-left and bottom-right corners. Since the MPDIoU loss function directly controls these two endpoints, it can more accurately describe the axial characteristics of green asparagus. Moreover, it does not rely on aspect ratio assumptions, making it suitable for green asparagus in various curved growth states. Therefore, the MPDIoU loss function is particularly well-suited for green asparagus detection and segmentation tasks.

For a given predicted box $B_{pred} = (x_1^{pred}, y_1^{pred}, x_2^{pred}, y_2^{pred})$ and ground truth box $B_{gt} = (x_1^{gt}, y_1^{gt}, x_2^{gt}, y_2^{gt})$, where (x_1, y_1) are the coordinates of the top-left corner and (x_2, y_2) are the coordinates of the bottom-right corner, the distances between the two key points (top-left and bottom-right corners) of the bounding boxes are respectively expressed as:

$$d_1^2 = \left(x_1^{pred} - x_1^{gt}\right)^2 + \left(y_1^{pred} - y_1^{gt}\right)^2 \tag{15}$$

$$d_2^2 = \left(x_2^{pred} - x_2^{gt}\right)^2 + \left(y_2^{pred} - y_2^{gt}\right)^2 \tag{16}$$

Therefore, the MPDIoU calculation expression is as follows:

$$MPDIoU = IoU - \frac{d_1^2}{c^2} - \frac{d_2^2}{c^2} \tag{17}$$

$$L_{MPDIoU} = 1 - MPDIoU \tag{18}$$

where IoU represents the intersection over union between the predicted and ground truth boxes, c denotes the diagonal length of the smallest enclosing box, $\frac{d_1^2}{c^2}$ and $\frac{d_2^2}{c^2}$ represent the normalized distance penalty terms, and L_{MPDIoU} denotes the MPDIoU loss function.

Morphological Measurement and Localization of Green Asparagus

This study integrates RGB images and depth images to differentiate individual green asparagus stalks, measure their morphology, and simultaneously localize the harvest point. First, the image data acquired by the Intel RealSense D435i depth camera is input into the proposed model. Point-to-point alignment is performed between the RGB and depth images to obtain the depth camera's intrinsic parameters and the depth frame within the aligned frames. The segmented green asparagus mask image is acquired and subjected to morphological processing. The contour of the green asparagus is then extracted, and skeleton fitting is performed. Subsequently, relevant initial points are extracted from the asparagus contour to calculate its length and diameter. Finally, after identifying harvestable green asparagus objects, the final harvest point is localized. The relevant processing steps are elaborated in detail below.

Morphological Processing of Segmented Images

After obtaining the region-of-interest (ROI) image of green asparagus through instance segmentation, it is converted into a binary image to facilitate subsequent analysis and processing, as shown in Fig. 8(a). However, due to interference from factors such as model training and complex environments, relying solely on the segmentation algorithm often fails to yield ideal results. The edges of the acquired green asparagus mask image may still contain noise, which can affect subsequent experimental outcomes. Therefore, to effectively improve segmentation accuracy and reliability, this study employs morphological analysis to process the green asparagus mask. First, an opening operation using a rectangular structural element is applied to the green asparagus mask to eliminate isolated noise points and small connected regions. The processing result is shown in Fig. 8(b). Then, an opening operation using an elliptical structural element is performed to further smooth the edges of the green asparagus mask. The processing result is shown in Fig. 8(c).

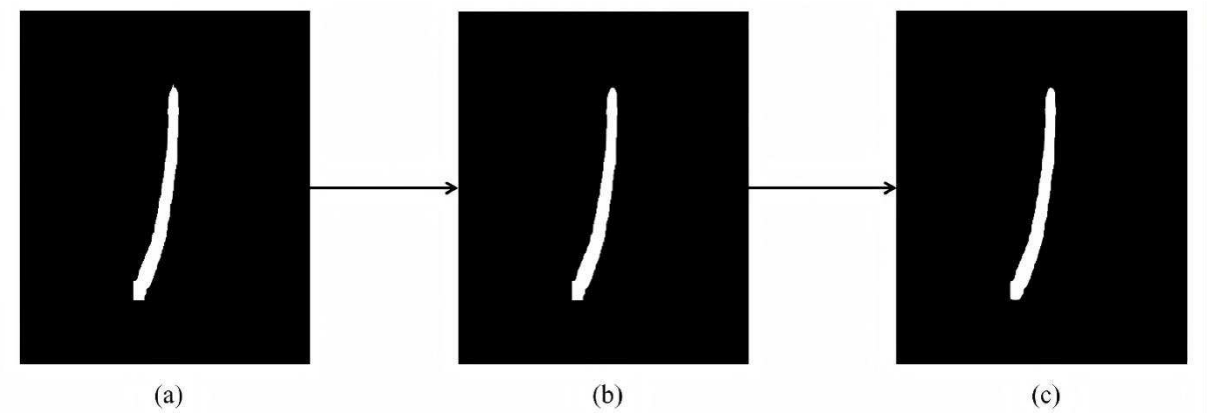


Fig. 8. Morphological processing of the green asparagus mask: (a) asparagus mask with noise, (b) denoised asparagus mask, (c) smoothed asparagus mask boundaries.

Green Asparagus Stem Skeleton Fitting and Length and Diameter Calculation

After locating and obtaining the green asparagus mask in the image, the Canny edge operator is used to extract the asparagus contour, as shown in Fig. 9(a). By traversing all pixel points on the contour, the highest and lowest points of the green asparagus mask projected onto the v-axis of the pixel coordinate system can be obtained. The green asparagus contour is then projected onto the u-axis and divided into k equal segments parallel to the u-axis, as illustrated in Fig. 9(b). Subsequently, based on the highest and lowest points of the v-axis projection and each dividing line along the boundary of the green asparagus contour, the pixel coordinates at the adjacent boundaries of each segment can be determined. The relevant calculation process is as follows.

$$v_{top}^{i} = min(v^{i}) \tag{19}$$

$$v_{bottom}^{i} = max(v^{i}) \tag{20}$$

$$v_{j}^{i} = v_{bottom}^{i} - \frac{v_{bottom}^{i} - v_{top}^{i}}{k} \tag{21}$$

In the equations, $i = 1,2,3,\cdots$ represents the number of asparagus stalks in the image, v^{i} denotes the coordinate values of the asparagus contour projected onto the v-axis, v_{top}^{i} is the coordinate value of the lowest point of the asparagus projection on the v-axis, v_{bottom}^{i} is the coordinate value of the highest point of the asparagus projection on the v-axis, and v_{j}^{i} represents the coordinate value at the boundary between segment j and segment $j+1$ in the v-axis projection of the $i-th$ asparagus stalk.

The value v_{j}^{i} is then used to traverse and obtain the initial point coordinates on the green asparagus contour, i.e., the pixel coordinates on both the inner and outer sides of each segment: $(u_{j1}^{i}, v_{j}^{i}), (u_{j2}^{i}, v_{j}^{i})$. Based on these initial point coordinates, the key coordinates of the green asparagus skeleton are calculated, specifically the central coordinate between the two initial coordinates: (u_{j}^{i}, v_{j}^{i}). The relevant calculation process is as follows.

$$u_{j}^{i} = \frac{u_{j1}^{i} + u_{j2}^{i}}{2} \tag{22}$$

Due to the predominantly irregular shapes of asparagus grown under natural conditions, calculating the stem length of green asparagus can introduce significant errors. Therefore, this study employs a segmented calculation method, which can effectively reduce errors and improve the accuracy of green asparagus stem length calculation. Using the coordinates of the

highest point, lowest point, and key points, the two adjacent points within each segment are connected with straight lines to form the skeleton of the green asparagus contour, as shown in the fig. 9(c). Functions from the pyrealsense2 library are utilized to convert the highest point, lowest point, initial points, and skeleton key points from the pixel coordinate system to their corresponding coordinates in the camera coordinate system. The length of each asparagus segment is then calculated, and the total length of the asparagus is the sum of the k segment lengths. The relevant calculation formula is as follows.

$$L^i = \sum_{n=0}^{k} \sqrt{\left(x^i_{c(n+1)} - x^i_{cn}\right) + \left(y^i_{c(n+1)} - y^i_{cn}\right) + \left(z^i_{c(n+1)} - z^i_{cn}\right)} \tag{23}$$

In the equation, x^i_{cn}、y^i_{cn}、z^i_{cn} represent the corresponding coordinate values of the highest point, lowest point, initial points, and skeleton key points in the camera coordinate system, and L^i denotes the total length of the asparagus stem.

Manual measurement of green asparagus diameter typically selects the thickest part for measurement. Therefore, within acceptable error limits, this study similarly divides the asparagus into k equal segments and calculates the maximum value within each segment as the green asparagus diameter. Unlike the calculation of green asparagus length, diameter calculation can be performed directly on the mask image in a two-dimensional coordinate system. As shown in Fig. 9(d), one segment of the k equal divisions is enlarged, and a point (x_{ci}, y_{ci}) on the stem is selected to calculate the diameter d_{ki}. The key points at both ends of this segment are connected to form a line segment denoted as l_l. A line L_D is drawn through the point (x_{ci}, y_{ci}) perpendicular to l_l. Representing the line segments l_l and l_D as vectors, let $l_l = (v^l_x, v^l_y)$ and $l_d = (v^d_x, v^d_y)$, where (v_x, v_y) denotes the components in the x-axis and y-axis directions. Thus, the following equation can be derived:

$$v^d_x \cdot v^l_x + v^d_y \cdot v^l_y = 0 \tag{24}$$

Setting $v^d_x = x - x_{ci}$ and $v^d_y = y - y_{ci}$, the following equation is obtained:

$$v^d_x \cdot v^l_x + v^d_y \cdot, v^l_y = 0 \tag{25}$$

By traversing points on the green asparagus mask, the diameter d_{ki} for each segment can be calculated. Finally, the diameter of the green asparagus is determined as $Max(d_{ki})$.

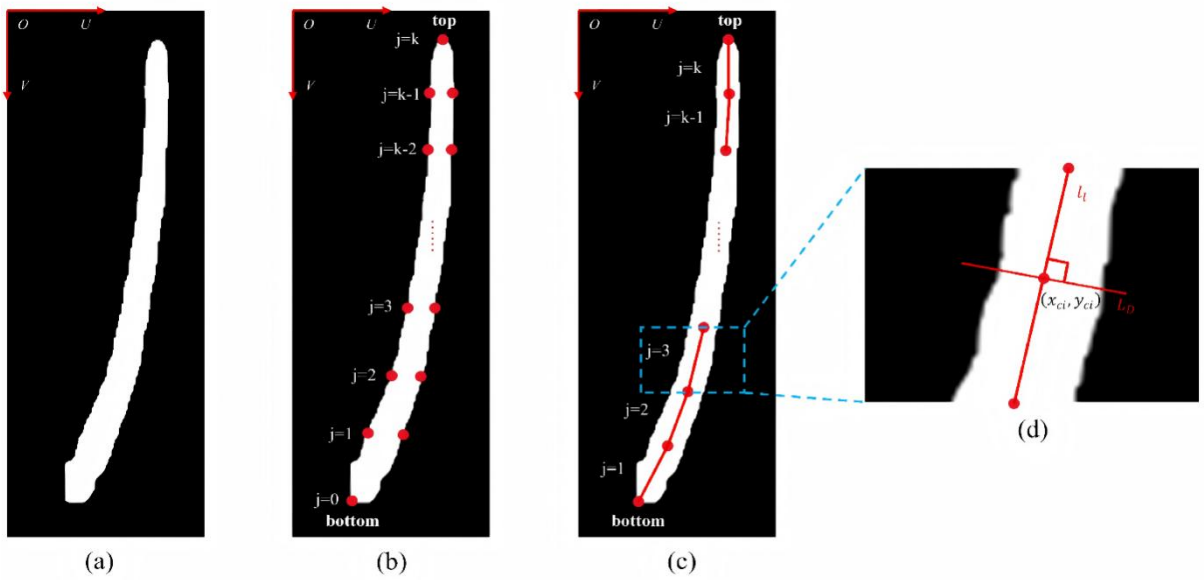


Fig. 9. (a) Contour image of green asparagus (b) Equally divided sections of the stem (c) Length calculation (d) Diameter calculation

Localization of Harvest Points for Green Asparagus

Unlike most other fruits and vegetables, green asparagus has no clear boundary between edible and inedible parts. The stem segment above the soil surface is generally green, tender and crisp, serving as the edible part. The root segment underground is typically white, with a short purple transition zone between the two parts—mainly located at the soil surface. Usually, the part below the purple zone is not consumed due to factors such as taste. Considering both the edible quality and maximum yield of green asparagus, this study defines the harvest point at the one-third segment of the first equally divided region at the bottom. Specifically, the first equally divided region at the bottom is evenly split into three segments, and the harvest point is the point on the mask skeleton closest to the lowest point of the mask. As shown in Fig. 10, in the pixel coordinate system, the two endpoints of the skeleton in the first segment are denoted as $p_{bottom}(u_1, v_1)$ and $p'(u_2, v_2)$, respectively. Thus, the harvest point $p(\frac{2u_1+u_2}{3}, \frac{2v_1+v_2}{3})$is derived. Subsequently, the coordinates of the harvest point in the pixel coordinate system are converted to the corresponding coordinates in the camera coordinate system using functions from the pyrealsense2 library, which serves as the final harvest point.

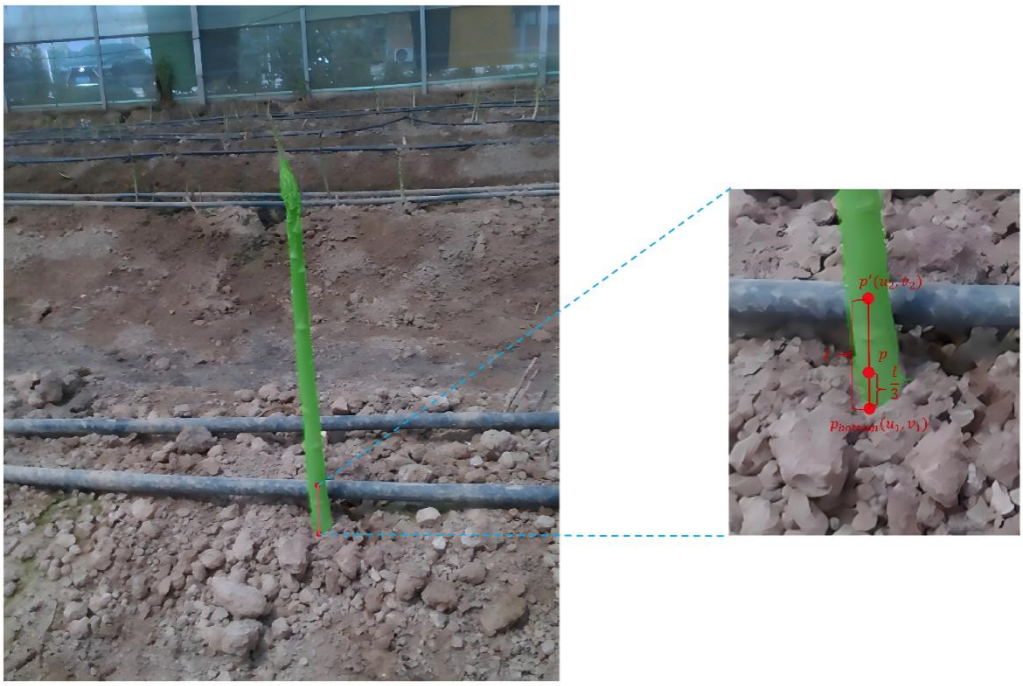


Fig. 10. Optimal cutting point localization for green asparagus

Model Training and Evaluation Metrics

Experimental Environment and Parameters

To ensure the validity and rigor of the experiments, all experiments in this study were conducted under the same hardware and software environments. The hardware environment

used includes an Intel I9-12900K CPU, an NVIDIA GeForce RTX 4090 (24G) GPU, and 64G of memory. The software environment consists of the Ubuntu 24.04 LTS operating system, Python 3.10.15, PyTorch 2.1.0, and CUDA 12.1.

The key hyperparameters of the model during training are as follows. The initial learning rate is 0.01, and the final learning rate is 0.0001. The batch size is set to 32, and the input image size is 640×640. The model is trained for 300 epochs using the SGD optimizer with a momentum of 0.937.

Evaluation Metrics

To verify the ability of the proposed green asparagus picking model to accurately segment each pixel of green asparagus in images and obtain the optimal cutting point position, Precision (P), Recall (R), mean Average Precision (mAP), F1-score, Parameters (Params), Floating-Point Operations (FLOPs), and Frames Per Second (FPS) are selected as key evaluation metrics. Among them, Params, FLOPs, and FPS reflect the model's size, computational complexity, and real-time performance, respectively. The specific definitions of the other evaluation metrics are as follows:

$$\mathrm{P} = \frac{TP}{TP + FP} \tag{26}$$

$$\mathrm{R} = \frac{TP}{TP + FN} \tag{27}$$

$$\mathrm{AP} = \int_0^1 P(R)dR \tag{28}$$

$$\mathrm{mAP} = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{29}$$

$$\mathrm{F1} = \frac{2 \times P \times R}{P + R} \tag{30}$$

Among them, TP (True Positive) denotes the successful identification and localization of green asparagus targets in images, FP (False Positive) denotes the successful identification and localization of non-green asparagus targets in images, and FN (False Negative) denotes the failure to successfully identify and localize green asparagus targets in images.

Results and Analysis

Ablation Experiments

To verify the effectiveness of the improved YOLOv8 method proposed in this study, multiple sets of ablation experiments were conducted on the collected green asparagus image dataset to comprehensively evaluate the function and effect of each improved component.

First, the YOLOv8n model was used to compare the impacts of two loss functions, CIoU and MPDIoU, on model performance. The experimental results are shown in Table 1. The model using MPDIoU as the loss function achieved better results in all metrics than the model using CIoU, demonstrating that MPDIoU can improve inference accuracy. Meanwhile, green asparagus images collected from the base were also used in this study to verify the detection performance of the two loss functions, and the results are shown in Fig. 11. It can be observed

from the figure that due to factors such as weeds, residual branches, and uneven land, the detection results may suffer from false detection, missed detection, and incomplete segmentation. However, the detection results using MPDIoU in this study are superior to those using CIoU. Therefore, based on the above results, adopting MPDIoU as the loss function in this study can effectively improve the model's detection performance for green asparagus.

Table 1 Results of ablation experiments on loss function improvement

No	Loss Function	Precision	Recall	$mAP_{0.5}$	F1
1	CIoU	0.972	0.961	0.990	0.971
2	MPDIoU	0.975	0.972	0.991	0.975

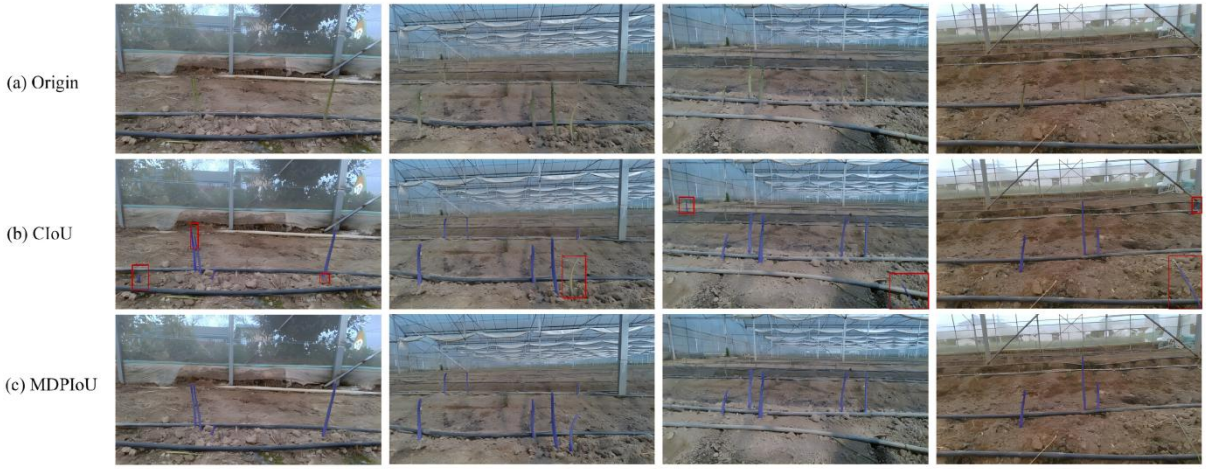


Fig. 11. Field green asparagus segmentation results with different loss functions

For the performance changes resulting from improvements to other modules of the model, the MPDIoU loss function was used in all ablation experiments for fairness, and the experimental results are shown in Table 2.The backbone network was replaced with StarNet. Due to its lightweight design and star-shaped structure, the model's detection accuracy and lightweight performance were improved. Compared with the baseline model, the accuracy increased by 0.3%, the number of model parameters decreased by 0.79M, and the model size reduced by 1.52MB. This improvement effectively lowers the computational cost and memory usage required for the model on edge devices.After replacing C2f with C2fDWConv_Strip, the recall rate increased by 0.6%, the number of model parameters decreased by 0.5M, and the model size reduced by 0.78MB. This indicates that the improvement of C2f enhances the model's expression of fine-grained features of green asparagus while achieving a certain level of lightweight enhancement.The addition of the EGAP attention module improved the model's precision and recall rate by 0.6% and 0.3% respectively compared with the baseline model, demonstrating that the introduction of EGAP can highlight the salient regions of green asparagus targets and enhance the edge details of green asparagus.

In the ablation experiments with pairwise combinations of improvements, the models of each combination achieved stable improvements in precision, with a slight increase in mAP@50, and further reductions in the number of model parameters and size. This proves that there is good synergy between the modules.

Combining all the above improvements, the finally proposed SES-YOLO model

achieved increases of 1.1%, 0.1%, and 0.2% in precision, recall rate, and mAP@50 respectively compared with the baseline model, while the number of parameters, floating-point operations (FLOPs), and model size decreased by 38.1%, 22.3%, and 36.5% respectively. It can be seen that under the synergy between modules, the model's precision is improved, and the computational complexity is significantly reduced, enabling the model to achieve an optimal balance between precision and efficiency.

Table 2 Results of ablation experiments on model improvements

No	StarNet	C2fDWConv_Strip	EGAP	Precision	Recall	$mAP_{0.5}$	Params	FLOPs	Weights
1	-	-	-	0.975	0.972	0.991	2.94M	5.43G	5.89MB
2	√	-	-	0.978	0.971	0.990	2.15M	4.59G	4.37MB
3	-	√	-	0.975	0.978	0.991	2.54M	5.03G	5.11MB
4	-	-	√	0.981	0.975	0.993	3.05M	5.47G	6.11MB
5	√	√	-	0.982	0.970	0.992	1.75M	4.19G	3.59MB
6	√	-	√	0.982	0.967	0.993	2.22M	4.62G	4.53MB
7	-	√	√	0.979	0.977	0.993	2.65M	5.08G	5.33MB
8	√	√	√	0.986	0.973	0.993	1.82M	4.22G	3.74MB

Note: √ Used, - Not used.

Performance Comparison of Instance Segmentation Models

To further validate the superiority of the proposed model in green asparagus detection and segmentation, this study compares several popular segmentation models, with experimental results shown in Table 3. As can be seen from Table 3, compared to other mainstream YOLO segmentation models—YOLOv5n-seg, YOLOv8n-seg, YOLOv11n-seg, and YOLOv12n-seg—the SES-YOLO model improves detection precision by 1.3, 1.4, 0.4, and 0.9 percentage points, respectively. Recall is increased by 1.4, 1.2, 1.1, and 1.8 percentage points, respectively. Floating-point operations (FLOPs) are reduced by 22.3%, 18.5%, and 13.7%, respectively. The number of parameters is decreased by 38.1%, 35.9%, and 30.8%, respectively. Model size is reduced by 36.5%, 34.8%, and 30.7%, respectively. In summary, SES-YOLO demonstrates its advantages across multiple metrics, effectively balancing accuracy and computational efficiency.

Table 3 Performance comparison of different models.

Model	Precision	Recall	$mAP_{0.5}$	F1	FLOPs	Params	Weights
YOLOv5n-seg	0.973	0.959	0.991	0.966	4.92G	2.44M	4.91MB
YOLOv8n-seg	0.972	0.961	0.990	0.966	5.43G	2.94M	5.89MB
YOLOv11n-seg	0.982	0.962	0.990	0.962	5.18G	2.84M	5.74MB
YOLOv12n-seg	0.977	0.955	0.987	0.966	4.89G	2.63M	5.40MB
SES-YOLO	0.986	0.973	0.993	0.979	4.22G	1.82M	3.74MB

To further compare the performance of different models in asparagus detection, Fig. 12 presents the visualization results of inference on the same image using the aforementioned models. As shown in the figure, YOLOv5n-seg exhibits issues of false detection and incomplete segmentation. YOLOv8n-seg and YOLOv11n-seg both suffer from missed detections, while YOLOv12n-seg shows incomplete segmentation. In contrast, the proposed SES-YOLO model successfully detects all asparagus instances, generates complete masks, and does not produce false detections, demonstrating its superior performance.

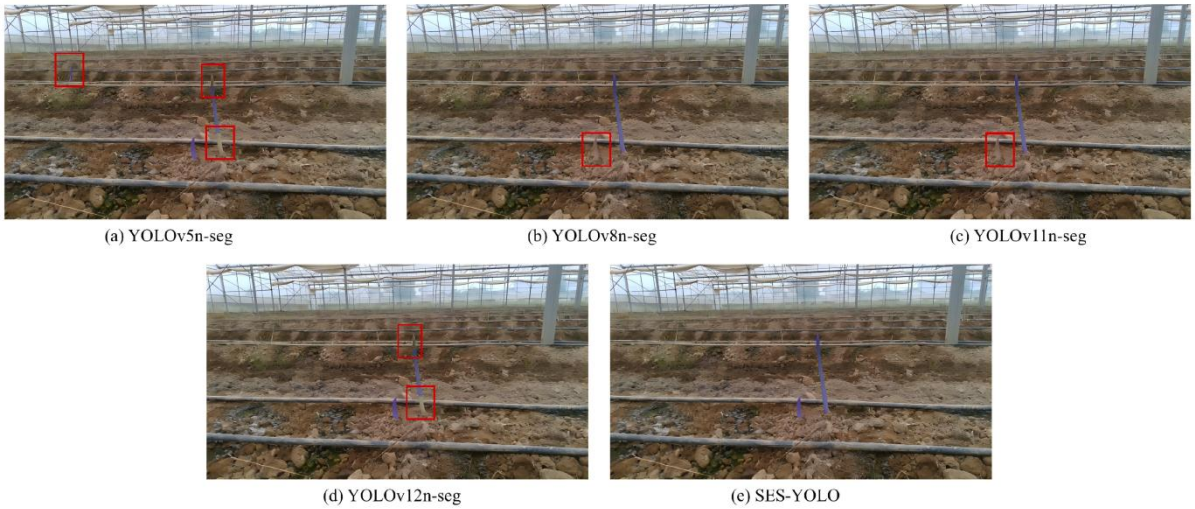


Fig. 12. Segmentation performance comparison of different models.

Experiments on Morphological Feature Calculation and Harvest Point Localization of Green Asparagus

To determine the value of the segmentation parameter *k*, this study conducted experiments indoors using four green asparagus stalks with different degrees of curvature, captured from a vertical angle relative to the ground. The values of *k* were set to 2, 4, 6, 8, and 10, respectively. For each *k*, the length of each segment of the green asparagus was calculated, and the total length was subsequently computed. Finally, the model-measured asparagus lengths under different *k* values were compared with the actual manually measured lengths to calculate the error rates. The results are presented in Table 4. Compared with other segmentation values, the overall error rate was minimized when the asparagus was divided into 8 segments. Therefore, the value of *k* was set to 8.

Table 4 Length error rates for different *k* segment values.

Degree	k=2	k=4	k=6	k=8	k=10
5°	1.238%	0.594%	0.201%	0.148%	0.574%
10°	1.535%	0.743%	0.232%	0.210%	0.688%
20°	2.178%	0.495%	0.337%	0.114%	0.717%
30°	2.574%	0.644%	0.495%	0.450%	0.847%

In the automated harvesting process of green asparagus, measuring its length and diameter is essential as they serve as key indicators for determining maturity and harvestability. Therefore, the model proposed in this study was applied at the planting base to

detect, segment, and calculate the length and diameter of green asparagus. The error rates were then computed by comparing these model-derived measurements with manually measured lengths and diameters. A total of 20 trials were conducted, and the results are presented in Table 5. As shown in Table 5, the length measurement errors for asparagus stalks No. 1, 16, and 20 were relatively large, all exceeding 4%. The smallest length measurement error was observed for stalk No. 10, at 0.20%. The average length measurement error was -0.11, with an average relative error of 1.97%. For diameter measurements, stalks No. 5 and 11 showed relatively large errors, both exceeding 4%, with the largest error being 5.88% for stalk No. 11. The smallest diameter measurement error was 0.69% for stalk No. 3. The average diameter measurement error was -0.02, with an average relative error of 2.07%. These results indicate that the model-measured values for asparagus length and diameter are mostly lower than the manual measurements, which is primarily related to the curvature and maturity of the asparagus. Greater curvature leads to larger errors, and compared to shorter asparagus, longer asparagus generally exhibits larger measurement errors, indicating that longer asparagus is more affected by curvature. The diameter of green asparagus is generally positively correlated with its growth stage; longer asparagus tends to have larger diameters, though this may vary slightly depending on specific growth and development conditions.

Table 5 Comparison of manual and model-measured green asparagus length and diameter.

No	Manual measurement length (cm)	Model measurement length (cm)	Length measurement error (cm)	Relative error Of length	Manual measurement diameter (cm)	Model measurement diameter (cm)	Diameter measurement error (cm)	Relative error of diameter
1	33.20	31.85	-1.35	4.07%	1.32	1.30	-0.02	1.52%
2	28.60	29.15	0.55	1.92%	1.35	1.32	-0.03	2.22%
3	27.50	26.74	-0.76	2.76%	1.45	1.46	0.01	0.69%
4	32.60	32.82	0.22	0.67%	1.35	1.33	-0.02	1.48%
5	22.10	21.63	-0.47	2.13%	1.20	1.15	-0.05	4.17%
6	28.10	29.12	1.02	3.63%	1.30	1.27	-0.03	2.31%
7	29.90	29.62	-0.28	0.94%	1.40	1.39	-0.01	0.71%
8	25.80	25.47	-0.33	1.30%	1.35	1.33	-0.02	1.48%
9	27.20	26.38	-0.82	3.01%	1.33	1.36	0.03	2.26%
10	20.20	20.24	0.04	0.20%	1.10	1.07	-0.03	2.73%
11	19.10	19.01	-0.09	0.47%	0.85	0.80	-0.05	5.88%
12	24.40	24.72	0.32	1.31%	1.23	1.21	-0.02	1.63%
13	18.50	18.34	-0.16	0.86%	1.20	1.22	0.02	1.67%
14	23.10	22.76	-0.34	1.47%	1.10	1.06	-0.04	3.64%

15	16.60	16.45	-0.15	0.90%	0.90	0.88	-0.02	2.22%
16	34.50	32.89	-1.61	4.67%	1.30	1.31	0.01	0.77%
17	29.10	30.02	0.92	3.16%	1.35	1.34	-0.01	0.74%
18	21.40	21.25	-0.15	0.70%	1.25	1.22	-0.03	2.40%
19	26.50	26.24	-0.26	0.98%	1.32	1.30	-0.02	1.52%
20	31.80	33.18	1.38	4.34%	1.50	1.52	0.02	1.33%
mean	26.01	25.89	-0.11	1.97%	1.26	1.24	-0.02	2.07%

To further validate the effectiveness and accuracy of the cutting point localization method proposed in this study, the ridge surface of the planting site was used as the horizontal reference plane. A comparative evaluation was conducted between manually measured cutting points and model-measured cutting points through a total of 15 trials. The results are presented in Table 6. The results indicate that the largest cutting point localization error was observed for green asparagus No. 1, measuring 0.090 cm, while the smallest error was for No. 13, at 0.002 cm. All other errors were less than 0.050 cm. The average horizontal localization error was 0.026 cm, the average depth localization error was 0.022 cm, and the average cutting point localization error was 0.034 cm.

These results demonstrate that the model and method proposed in this study exhibit high accuracy in green asparagus morphological measurement and cutting point localization.

Table 6 Comparison of cutting point localization for green asparagus.

No	Manual marking cutting-point coordinates	Model positioning cutting-point coordinates	Horizontal positioning error (cm)	Depth positioning error (cm)	cutting-point determination error (cm)
1	0.083,0.109, 0.525	0.031,0.118,0.452	0.053	0.073	0.090
2	0.057,0.122,0.457	0.023,0.138,0.433	0.038	0.024	0.045
3	0.113,0.126,0.486	0.096,0.105,0.467	0.027	0.019	0.033
4	0.075,0.134,0.397	0.069,0.124,0.416	0.012	0.014	0.018
5	-0.033,0.141,0.512	-0.045,0.152,0.511	0.016	0.001	0.016
6	0.026,0.099,0.455	-0.008,0.075,0.478	0.042	0.023	0.048
7	0.068,0.109,0.406	0.012,0.121,0.310	0.057	0.096	0.011
8	0.055,0.120,0.542	0.072,0.111,0.527	0.019	0.015	0.024
9	-0.021,0.098,0.379	-0.023,0.086,0.389	0.012	0.010	0.016
10	0.108,0.131,0.358	0.110,0.116,0.366	0.015	0.008	0.017
11	0.063,0.154,0.566	0.087,0.167,0.594	0.027	0.028	0.039
12	0.096,0.138,0.403	0.063,0.124,0.425	0.036	0.022	0.042

13	-0.052,0.127,0.369	-0.051,0.127,0.371	0.001	0.002	0.002
14	0.085,0.117,0.390	0.085,0.133,0.385	0.016	0.005	0.017
15	0.025,0.147,0.343	0.032,0.141,0.0.359	0.014	0.016	0.021
mean			0.026	0.022	0.034

Model Deployment and Testing

This study utilized the NVIDIA Jetson Orin NX as the edge computing platform for deployment testing, as shown in Fig. 13. This platform is equipped with a 1024-core NVIDIA Ampere architecture GPU with 32 Tensor Cores, a 6-core ARM architecture CPU, and 16 GB of memory. It offers a peak computing power of up to 100 TOPS and supports acceleration tools such as CUDA, cuDNN, and TensorRT, meeting the requirements for low-power, real-time model inference.

The improved SES-YOLO model and the baseline YOLOv8n-seg model were deployed on the NVIDIA Jetson Orin NX edge computing platform, using the TensorRT inference framework for acceleration. Simultaneously, 180 images of green asparagus not involved in model training were collected as a test set for performance inference experiments. The resolution of input images was uniformly adjusted to 640×640. The performance metrics after model deployment are shown in Table 7. As indicated in Table 7, the detection speed of SES-YOLO is 58.4% higher than that of the baseline model YOLOv8n-seg, and its detection accuracy is 1.4 percentage points higher. The results demonstrate that the performance of the improved SES-YOLO model meets the requirements for deployment on edge devices and real-time detection.

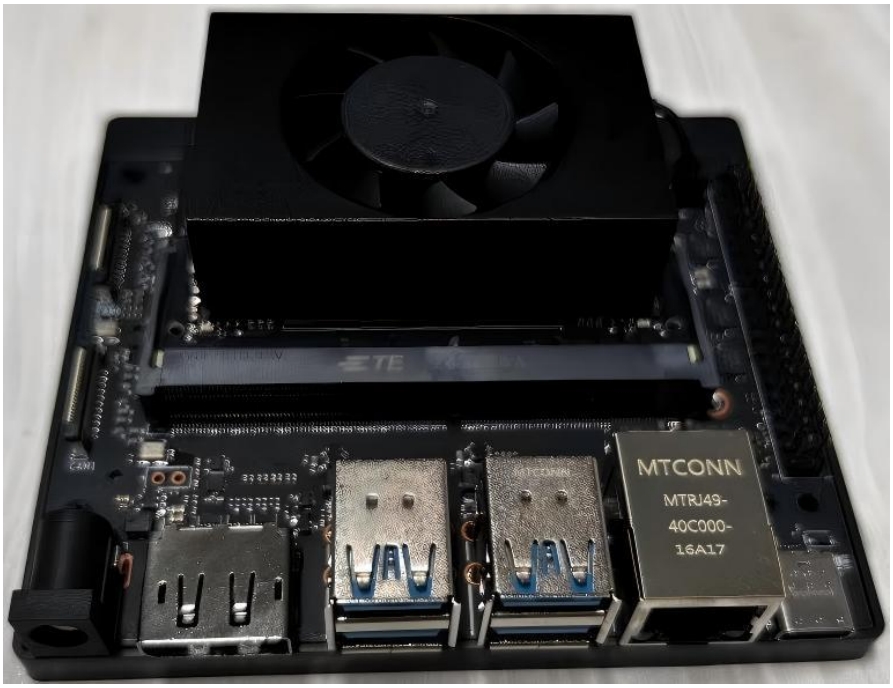


Fig. 13. NVIDIA Jetson Orin NX edge computing platform.

Table 7 Performance comparison of model deployment on edge devices.

Model	Precision	Recall	$mAP_{0.5}$	F1	FPS
YOLOv8n-seg	0.973	0.963	0.985	0.968	24.43
SES-YOLO	0.987	0.972	0.990	0.979	38.69

Conclusion

(1) To address the difficulties in recognition and localization caused by the curved growth morphology of green asparagus in natural environments and the partial burial of its rootstalks in soil, as well as the challenge of balancing accuracy and speed in instance segmentation algorithms, an improved lightweight instance segmentation model named SES-YOLO is proposed. Based on YOLOv8n-seg, this model introduces the lightweight StarNet backbone to replace the original structure, designs the C2fDWConv_Strip module integrating depthwise separable convolution and strip convolution, incorporates the EGAP edge enhancement module, and adopts the MPDIoU loss function to optimize bounding box regression. These enhancements significantly improve feature extraction capability and localization accuracy for slender, curved targets while substantially reducing the model's computational burden.

(2) To validate the effectiveness of each improved module, this study designed systematic ablation experiments, testing the impact on model performance from replacing the backbone network, modifying the C2f module, embedding the EGAP module, and substituting the loss function. Experimental results show that each module effectively improves the model's detection accuracy and lightweight characteristics, and the modules exhibit good synergistic effects. Ultimately, SES-YOLO maintains high segmentation accuracy while reducing the parameter count by 38.1% and computational load by 22.3%.

(3) Under identical experimental conditions, SES-YOLO was compared with current mainstream instance segmentation models, including YOLOv5n-seg, YOLOv8n-seg, YOLOv11n-seg, and YOLOv12n-seg. SES-YOLO demonstrated optimal performance across key metrics such as precision, recall, and mAP@50. The morphological measurement and cutting point localization method based on RGB-D data also achieved high accuracy, with average relative errors of 1.97% for length and 2.07% for diameter, and an average cutting point localization error of 0.34 cm. Furthermore, when deployed on a Jetson Orin NX edge device, SES-YOLO achieved a detection speed of 38.69 FPS, representing a 58.4% improvement over the baseline model.

Statements and Declarations

Competing interests

All authors certify that they have no affiliations with or involvement in any organizati on or entity with any financial interest or non-financial interest in the subject matter o r materials discussed in this manuscript.

Funding Statement

This research was funded by Open Research Projects of the Key Laboratory of Knowl edge Mining and Knowledge Services in Agricultural Convergence Publishing, Institute of Agricultural Information, Chinese Academy of Agricultural Sciences (2025KMKS0 4), the National Natural Science Foundation of China (Grant No. 32560736), the Basic Research and Talent Cultivation Project of Jiangxi Academy of Agricultural Sciences (Grant No. JXSNKYJCRC202309).

Author Contributions

The authors confirm contribution to the paper as follows: Conceptualization: Qingqi L u and Yuliang Zhong; methodology: Qingqi Lu and Xiaobo Su; software: Ruopeng Du and Hui Hong; validation: Jinsong Zhou and Jian Ding; formal analysis: Qingqi Lu a nd Yuliang Zhong; investigation: Xiaobo Su and Ruopeng Du; resources: Jian Ding an d Hui Hong; data curation: Qingqi Lu; writing—original draft preparation: Qingqi Lu and Jian Ding; writing—review and editing: Xiaobo Su and Yuliang Zhong; visualizati on: Ruopeng Du and Hui Hong; supervision: Jinsong Zhou; project administration: Xia obo Su; funding acquisition: Qingqi Lu, Jinsong Zhou, Xiaobo Su and Ruopeng Du. All authors reviewed and approved the final version of the manuscript.

Data availability statements

I used or generated research data in this study. Data sets generated during the current study are available from the corresponding author on reasonable request.

Ethics statements

This study did not involve human or animal subjects. Permission for the collection of green asparagus samples was obtained from Mengfei Pastoral Farm.

References

[1]. Bolya, D., Zhou, C., Xiao, F. & Lee, Y. J. YOLACT: real-time instance segmentation. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV) 9156-9165 (IEEE, 2019); 10.1109/ICCV.2019.00925.

[2]. Chen, C. et al. A robust vision system for measuring and positioning green asparagus based on YOLO-seg and 3D point cloud data. Comput. Electron. Agric. 230, 109937; 10.1016/j.compag.2025.109937 (2025).

[3]. Chitrakar, B., Zhang, M. & Adhikari, B. Asparagus (Asparagus officinalis): processing effect on nutritional and phytochemical composition of spear and hard-stem by products. Trends Food Sci. Technol. 93, 1-11; 10.1016/j.tifs.2019.08.020 (2019).

[4]. Gené-Mola, J. et al. Looking behind occlusions: a study on amodal segmentation for robust on-tree apple fruit size estimation. Comput. Electron. Agric. 209, 107854; 10.1016/j.compag.2023.107854 (2023).

[5]. Ghazal, S., Munir, A. & Qureshi, W. S. Computer vision in smart agriculture and precision farming: techniques and applications. Artif. Intell. Agric. 13, 64-83; 10.1016/j.aiia.2024.06.004 (2024).

[6]. Ghimire, H. et al. Wheat spike and spikelet detection and counting from high-resolution digital imagery using YOLO with oriented bounding boxes. Sci. Rep. (2026).

[7]. Keshavarzi, M., Mesarich, C., Bailey, D., Johnson, M. & Gupta, G. S. A review of semantic segmentation methods and their application in apple disease detection. Comput. Electron. Agric. 237, 110531; 10.1016/j.compag.2025.110531 (2025).

[8]. Kim, S. et al. De-occlusion models and diffusion-based data augmentation for size estimation of on-plant oriental melons. Plant Phenomics 7, 100097; 10.1016/j.plaphe.2025.100097 (2025).

[9]. Kuan, Y. N., Goh, K. M. & Lim, L. L. Systematic review on machine learning and computer vision in precision agriculture: applications, trends, and emerging techniques. Eng. Appl. Artif. Intell. 148, 110401; 10.1016/j.engappai.2025.110401 (2025).

[10]. Kurtser, P. & Lowry, S. RGB-D datasets for robotic perception in site-specific agricultural operations—a survey. Comput. Electron. Agric. 212, 108035; 10.1016/j.compag.2023.108035 (2023).

[11]. Li, H. et al. Positioning of mango picking point using an improved YOLOv8 architecture with object detection and instance segmentation. Biosyst. Eng. 247, 202-220; 10.1016/j.biosystemseng.2024.09.015 (2024).

[12]. Liu, H., Rivera Soto, R. A., Xiao, F. & Lee, Y. J. YolactEdge: real-time instance segmentation on the edge. In 2021 IEEE International Conference on Robotics and Automation (ICRA) 9579-9585 (IEEE, 2021); 10.1109/ICRA48506.2021.9561858.

[13]. Lu, Y. & Mohammadi, P. Automated asparagus harvesting technology: a review of research and developments since the 1950s in the United States and beyond. Comput. Electron. Agric. 237, 110744; 10.1016/j.compag.2025.110744 (2025).

[14]. Luqiang, Z. et al. Real-time recognition and dynamic positioning method for cotton terminal buds based on CottonBud-YOLOv5s algorithm and RGBD camera. Smart Agric. Technol. 11, 100975; 10.1016/j.atech.2025.100975 (2025).

[15]. Ma, S. & Xu, Y. MPDIoU: a loss for efficient and accurate bounding box regression. Preprint at https://doi.org/10.48550/ARXIV.2307.07662 (2023).

[16]. Ma, X., Dai, X., Bai, Y., Wang, Y. & Fu, Y. Rewrite the stars. Preprint at https://doi.org/10.48550/ARXIV.2403.19967 (2024).

[17]. Ma, Z. et al. STRAW-YOLO: a detection method for strawberry fruits targets and key points. Comput. Electron. Agric. 230, 109853; 10.1016/j.compag.2024.109853 (2025).

[18]. Mu, S. et al. S2CPL: a novel method of the harvest evaluation and subsoil 3D cutting-point location for selective harvesting of green asparagus. Comput. Electron. Agric. 225, 109316; 10.1016/j.compag.2024.109316 (2024).

[19]. Paul, A., Machavaram, R., Ambuj, Kumar, D. & Nagar, H. Smart solutions for capsicum harvesting: unleashing the power of YOLO for detection, segmentation, growth stage classification, counting, and real-time mobile identification. Comput. Electron. Agric. 219, 108832; 10.1016/j.compag.2024.108832 (2024).

[20]. Qi, Z. et al. A novel method for tomato stem diameter measurement based on i

mproved YOLOv8-seg and RGB-D data. Comput. Electron. Agric. 226, 109387; 10.1016/j.compag.2024.109387 (2024).

[21]. Qu, H., Du, H., Tang, X. & Zhai, S. Citrus fruit diameter estimation in the field using monocular camera. Biosyst. Eng. 252, 47-60; 10.1016/j.biosystemseng.2025.02.012 (2025).

[22]. Redondo-Cuenca, A. et al. Nutritional composition of green asparagus (Asparagus officinalis L.), edible part and by-products, and assessment of their effect on the growth of human gut-associated bacteria. Food Res. Int. 163, 112284; 10.1016/j.foodres.2022.112284 (2023).

[23]. Rong, J., Zheng, W., Qi, Z., Yuan, T. & Wang, P. RTMFusion: an enhanced dual-stream architecture algorithm fusing RGB and depth features for instance segmentation of tomato organs. Measurement 239, 115484; 10.1016/j.measurement.2024.115484 (2025).

[24]. Shen, Q., Zhang, X., Shen, M. & Xu, D. Multi-scale adaptive YOLO for instance segmentation of grape pedicels. Comput. Electron. Agric. 229, 109712; 10.1016/j.compag.2024.109712 (2025).

[25]. Sung, W., Tofik Isa, I. G. & Hsiao, S. Enhancing lightweight mango disease detection model performance through a combined attention module. Comput. Mater. Contin. 86, 1-31; 10.32604/cmc.2025.070922 (2026).

[26]. Wang, D. & He, D. Fusion of Mask RCNN and attention mechanism for instance segmentation of apples under complex background. Comput. Electron. Agric. 196, 106864; 10.1016/j.compag.2022.106864 (2022).

[27]. Wang, J. et al. PG-YOLO: an efficient detection algorithm for pomegranate before fruit thinning. Eng. Appl. Artif. Intell. 134, 108700; 10.1016/j.engappai.2024.108700 (2024).

[28]. Wang, Q. et al. ECA-Net: efficient channel attention for deep convolutional neural networks. Preprint at https://doi.org/10.48550/ARXIV.1910.03151 (2019).

[29]. Yi, W., Xia, S., Kuzmin, S., Gerasimov, I. & Cheng, X. RTFVE-YOLOv9: real-time fruit volume estimation model integrating YOLOv9 and binocular stereo vision. Comput. Electron. Agric. 236, 110401; 10.1016/j.compag.2025.110401 (2025).

[30]. Yin, H., Luo, S., Tang, J. & Hua, J. RGB camera-based monocular stereo vision applied in plant phenotype: a survey. Comput. Electron. Agric. 227, 109523; 10.1016/j.compag.2024.109523 (2024).

[31]. Yu, Y., Zhang, K., Yang, L. & Zhang, D. Fruit detection for strawberry harvesting robot in non-structural environment based on Mask-RCNN. Comput. Electron. Agric. 163, 104846; 10.1016/j.compag.2019.06.001 (2019).

[32]. Yuan, X. et al. Strip R-CNN: large strip convolution for remote sensing object detection. Preprint at https://doi.org/10.48550/ARXIV.2501.03775 (2025).