

## Article

# YOLOv8n-DDA-SAM: Accurate Cutting-Point Estimation for Robotic Cherry-Tomato Harvesting

Gengming Zhang <sup>1,†</sup> , Hao Cao <sup>1,†</sup>, Yangwen Jin <sup>1</sup> , Yi Zhong <sup>1</sup>, Anbang Zhao <sup>1</sup>, Xiangjun Zou <sup>2</sup>   
and Hongjun Wang <sup>1,\*</sup>

<sup>1</sup> College of Engineering, South China Agricultural University, Guangzhou 510642, China; gengming\_zhang@stu.scau.edu.cn (G.Z.); caohao@stu.scau.edu.cn (H.C.); yangtsejin@stu.scau.edu.cn (Y.J.); zy0328@stu.scau.edu.cn (Y.Z.); 20223171033@stu.scau.edu.cn (A.Z.)

<sup>2</sup> Xinjiang Agricultural and Pastoral Robotics and High-End Equipment Engineering Research Center, Xinjiang University, Urumqi 830046, China; xjzou@scau.edu.cn

\* Correspondence: xtwhj@scau.edu.cn

† These authors contributed equally to this work.

**Abstract:** Accurately identifying cherry-tomato picking points and obtaining their coordinate locations is critical to the success of cherry-tomato picking robots. However, previous methods for semantic segmentation alone or combining object detection with traditional image processing have struggled to accurately determine the cherry-tomato picking point due to challenges such as leaves as well as targets that are too small. In this study, we propose a YOLOv8n-DDA-SAM model that adds a semantic segmentation branch to target detection to achieve the desired detection and compute the picking point. To be specific, YOLOv8n is used as the initial model, and a dynamic snake convolutional layer (DySnakeConv) that is more suitable for the detection of the stems of cherry-tomato is used in neck of the model. In addition, the dynamic large convolutional kernel attention mechanism adopted in backbone and the use of ADown convolution resulted in a better fusion of the stem features with the neck features and a certain decrease in the number of model parameters without loss of accuracy. Combined with semantic branch SAM, the mask of picking points is effectively obtained and then the accurate picking point is obtained by simple shape-centering calculation. As suggested by the experimental results, the proposed YOLOv8n-DDA-SAM model is significantly improved from previous models not only in detecting stems but also in obtaining stem's masks. In the mAP@0.5 and F1-score, the YOLOv8n-DDA-SAM achieved 85.90% and 86.13% respectively. Compared with the original YOLOv8n, YOLOv7, RT-DETR-L and YOLOv9c, the mAP@0.5 has improved by 24.7%, 21.85%, 19.76%, 15.99% respectively. F1-score has increased by 16.34%, 12.11%, 10.09%, 8.07% respectively, and the number of parameters is only 6.37M. In the semantic segmentation branch, not only does it not need to produce relevant datasets, but also improved its mIOU by 11.43%, 6.94%, 5.53%, 4.22% and mAP@0.5 by 12.33%, 7.49%, 6.4%, 5.99% compared to Deeplabv3+, Mask2former, DDRNet and SAN respectively. In summary, the model can well satisfy the requirements of high-precision detection and provides a strategy for the detection system of the cherry-tomato.

**Keywords:** cherry-tomato; cutting point; DSConv; SAM; DLKA; YOLOv8n



**Citation:** Zhang, G.; Cao, H.; Jin, Y.; Zhong, Y.; Zhao, A.; Zou, X.; Wang, H. YOLOv8n-DDA-SAM: Accurate Cutting-Point Estimation for Robotic Cherry-Tomato Harvesting.

*Agriculture* **2024**, *14*, 1011. <https://doi.org/10.3390/agriculture14071011>

Academic Editor: Yanbo Huang

Received: 7 June 2024

Revised: 20 June 2024

Accepted: 22 June 2024

Published: 26 June 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Cherry-tomato, is a fruit and vegetable with a fresh taste and rich nutrition [1,2]. As a very important fruit food, cherry-tomato is one of the main vegetables for people in most parts of the world, and China is the largest producer and consumer of cherry-tomato in the world [3]. From the process of planting and ripening to harvesting and transportation for delivery, the link that most affects the development of the huckleberry industry is the harvesting link [4]. Cherry-tomato harvesting is a labor-intensive task, mainly relying on manual picking and transportation, due to the small size of the fruit, usually only 1–2 cm, the distribution of the fruit in bunches, and the thin skin of the fruit, so the picking efficiency

is low, the breakage rate is high, and the labor cost is high, which restricts the development of the cherry-tomato industry [5,6]. To realize machine harvesting, the most critical link is the identification and localization of the point of cutting of cherry-tomato, and only by obtaining the location of the point of picking of cherry-tomato can subsequent steps such as automatic harvesting be realized.

With the development of artificial intelligence technology, fruit identification and localization techniques have become more and more increasingly important in the fruit harvesting process. Traditional image detection methods have been the main method for early detection of fruits [7]. Khoshroo et al. [8] developed an algorithm based on image processing techniques for detecting red cherry-tomato on plants in a greenhouse and guiding a robot to pick ripe red cherry-tomato. The image background is removed by subtractive operations (R-G and R-B) and then the noise is eliminated using the open operation algorithm to ensure that only the front facing cherry-tomato are selected and the red cherry-tomato are detected using the watershed algorithm and the region growing algorithm. However, the extraction of red cherry-tomato from the image was not effective under insufficient lighting. Furthermore, there may be limitations in the performance of the algorithm under different environmental conditions. Additionally, failure to separate overlapping cherry-tomato may affect the detection accuracy. Omidi-Arjenaki et al. [9] developed a machine vision-based cherry-tomato sorting system that first converts cherry-tomato images into HSI space (hue, saturation, and luminance), and separates cherry-tomato from the background by filtering out colors outside the specified HSI range. The average of the color components is then used as a benchmark for identifying the state of ripe cherry-tomato in the algorithm, and the shape index of the cherry-tomato is calculated using the eccentricity, which can be regarded as a measure of the degree of deviation from circularity of a 2-D object (cross-section). The shape of a cherry-tomato is determined by its curvature, which categorises the shape as either circular or elliptical. However, the identification of different types of cherry-tomato or specific defects may require further optimisation of the algorithm and parameter settings to improve the applicability and generalisation of the system.

Traditional image processing algorithms possess their own limitations, such as a single recognition mode, inability to apply to complex environments, and low accuracy. The utilisation of machine vision algorithms is becoming increasingly prevalent among researchers engaged in the recognition and localisation of the cherry-tomato. These algorithms are capable of simultaneously extracting multiple features, including color, texture, shape, and so forth, which enables a more comprehensive characterisation of the cherry-tomato and enhances the accuracy of recognition [10,11]. Furthermore, machine vision algorithms are able to adapt to varying lighting conditions, angles, and backgrounds, and are capable of effective recognition even when the fruit is partially obscured or deformed [12–15].

Pavithra et al. [16] proposed a color-based ripeness estimation algorithm utilising the concept of change in external color at different ripening stages. Furthermore, a novel color segmentation algorithm based on euclidean distance was developed for extracting the surface of cherry-tomato from the background and leaves. Finally, the extracted features were fed into a K-nearest neighbour support vector machine-based (SVM) classifier to classify the ripe fruits into three categories. The use of machine vision techniques enables the rapid and precise scoring and categorisation of the cherry-tomato. However, the reliability of the system may be influenced by external factors, such as light and environmental conditions, and requires further optimisation and improvement. Liu et al. [17] utilizes histogram gradient orientation (HOG) descriptors to train support vector machine classifiers while combining coarse-to-fine scanning methods, false color removal (FCR), and non-maximal suppression (NMS) for accurate cherry-tomato detection. However, the accuracy of the algorithm for occluded and overlapped cherry-tomato detection is not satisfactory, particularly when the occluded region is more than 50%.

Deep learning algorithms represent an advanced technique in the field of machine vision. They are able to automatically learn complex feature representations from large

amounts of data without the need for manually designing feature extractors, in contrast to the above vision algorithms [18–22]. This enables the algorithm to better understand and recognise patterns and structures in images. Furthermore, deep learning algorithms demonstrate strong generalisation capabilities and can show good generalisation on unseen data, i.e., they can maintain a high recognition accuracy in new scenarios [23–25]. Concurrently, deep learning algorithms are capable of addressing more intricate visual tasks, such as object recognition in variable lighting conditions, diverse viewing angles, and complex backgrounds [26–28]. Consequently, the research and application of deep learning algorithms has seen significant growth in recent years, becoming a key area in artificial intelligence.

Yuan et al. [29] developed a robust cherry-tomato detection algorithm using deep learning techniques in a greenhouse scenario using a modified SSD algorithm based on the Inception V2 network for recognition of cherry-tomato. The capacity to accurately identify cherry-tomato was markedly enhanced and exhibited greater generalizability than traditional methodologies. Chen et al. [30] used a dual-path network as a feature extraction network to extract richer small target features, and used an improved K-means clustering algorithm to compute the anchor box size. The capacity to accurately identify cherry-tomato was markedly enhanced and exhibited greater generalizability than conventional methodologies. Fuentes-Peñailillo et al. [31] presents a novel approach for automating seedling counts in greenhouse environments using computer vision and AI. By leveraging a combination of image processing techniques and a SSD MobileNet object detection model, the system achieves high accuracy in counting seedlings across various crops and tray types, reducing manual labor and improving efficiency in the horticulture industry. Kim et al. [32] presents an autonomous harvesting robot system for tomatoes using a deep learning network called Deep-ToMaToS, which classifies tomato maturity and estimates the 6D poses of fruits and side-stems simultaneously. The system employs a Peck-and-Pull harvesting method and achieves an 84.5% success rate in a smart farm environment. However, the accuracy of the approach remains suboptimal for laterally growing cherry-tomato. The harvesting method for cherry-tomato, which are soft fruit and fruit bunches with longer fruit, differs from that of similar fruits such as apples. Accurate judgement of the location of the picking is essential to ensure that the automated equipment can separate the fruit and fruit branches [33–35]. Nowadays, the hottest deep learning algorithms in the field of target detection and recognition are the YOLO series, which started from the initial YOLOv1, YOLOv2 to today's YOLOv8, etc. The algorithms of the YOLO series are constantly evolving and improving, and the detection accuracy and speed of the models are constantly increasing, which is making a big splash in various visual recognition applications. In [36], a new method for detecting small target tomatoes in a large tomato production field is presented. Traditional fruit detection methods have low robustness and are time consuming, so deep learning techniques are used to improve the accuracy and robustness of detection. The authors proposed an improved YOLOv8 network called RSR-YOLO, which uses partial swarm convolution and an innovative FasterNet module for better feature extraction. The Gather and Distribute mechanism is combined with a redesigned feature fusion to extract and fuse different levels of tomato features. In addition, Repulsion Loss is used to check the effect of fruit overlap and leaf occlusion on the detection results. RSR-YOLO achieves high precision in recall, F1 score and average precision, which outperforms the original YOLOv8 network. In [37], the YOLOv8 deep learning model was used to optimise tomato plant phenotype detection, a novel data balancing method was proposed, and a Squeeze-and-Excitation (SE) attention module was integrated into the YOLOv8 architecture to improve the detection performance. The results show that the data balancing approach significantly improves model performance, especially when pre-training weights are balanced on the data. Yan et al. [38] proposed a Si-YOLO-based deep learning algorithm for identifying and locating cherry-tomato picking points in unstructured environments. The method enhances the dataset by combining a target detection algorithm and an attention mechanism, and uses both GAN and traditional image data enhancement methods to locate

the picking points of cherry-tomato more accurately and to improve the generalisation of the model. However, there is still room for improvement in the precision and recall of the model, and further research is still needed on how to fuse the depth information and color features of the image to obtain accurate coordinates of the picking point. Yang et al. [39] improved YOLOv8s to achieve a 2% increase in accuracy and a 28% decrease in the number of parameters to meet the requirements of a maidenhair fruit picking robot. Zhang et al. [40] proposed a multi-class detection method using an improved YOLOv4-Tiny model to categorize cherry-tomato images into four classes based on the occlusion of cherry-tomato, and finally achieved a high average accuracy.

In addition, although many scholars have used YOLO, an efficient target detection algorithm, to identify and localize the picking point of cherry-tomato, the accuracy of small target object detection is still lacking due to the defects of YOLO itself, so many scholars have proposed to obtain the localization of the cherry-tomato by instance segmentation. In [41], a method for detecting and segmenting ripe green tomatoes using Mask R-CNN networks and automatic image acquisition via a mobile robot in a greenhouse environment. The authors compared their method with previous studies using traditional machine learning techniques and deep learning convolutional neural networks for fruit detection and yield estimation, and the proposed method possessed higher accuracy. Fawzia Rahim et al. [42] describes the use of deep instance segmentation, data synthesis and color analysis to accurately identify and count tomatoes at different growth stages. By training a Mask R-CNN neural network using a synthetic dataset, the researchers were able to efficiently segment tomato instances in an image and determine their growth stages through color-based thresholding. The synthetic data generation preserved the physical structure of the objects and provided realistic growing scenarios for training the model. The method demonstrated high accuracy in tomato segmentation and counting with an average precision of 92.1% and a recall of 91.4%. However, the performance of the Mask R-CNN network used by the authors on the real dataset was slightly degraded compared to the synthetic dataset, indicating that there are still some challenges in real agricultural environments. In addition, the research on the effects of factors such as illumination and angle is relatively lacking, and there is not enough detailed discussion on how to deal with overlapping and occlusion, as well as strategies for dealing with anomalies. Yoshida et al. [43] presents a method for detecting tomato peduncles using 3D point clouds, enabling harvesting robots to identify the optimal cutting points for tomato harvesting. The proposed approach utilizes voxelization, classification, clustering, maturity estimation, and an energy function to accurately identify peduncles and determine cutting points, even for short peduncles.

In order to solve the above problems, we proposed a network model YOLOv8n-DDA-SAM to accurately obtain the cutting points of the cherry-tomato in this research. Taking the YOLOv8n model as a benchmark, our method introduced appropriate dynamic snake convolution and large kernel convolutional attention mechanism with specific data enhancement for the features of the stem of the cherry-tomato. A semantic segmentation branch is added to obtain the stem mask more accurately for the accuracy and simplicity of the subsequent localisation information computation. This branch not only saves the trouble of creating datasets, but also achieves excellent performance in segmenting the information of stem. The contributions of this study are summarized as follows:

- The YOLOv8n-DDA-SAM model is proposed based on the introduction of a large model branch SAM, which can accurately obtain the semantically segmented image of the cherry-tomato stem part and accurately compute it with the cutting point.
- DySnakeConv, which is sensitive to elongated topology, is introduced based on the original YOLOv8n, which greatly improving the accuracy of cherry-tomato stem detection.
- In order to better fuse the features obtained from the main stem at the neck, DLKA, a deformable large kernel attention mechanism, is introduced, which plays a crucial role in the detection of the stems of the cherry-tomato.

- The introduced branch not only avoids the creation of semantic segmentation datasets but also significantly improves the accuracy of semantic segmentation.

## 2. Materials and Methods

### 2.1. Dataset

#### 2.1.1. Data Acquisition

The image dataset of the cherry-tomato used in this study was collected from the Litchi Culture Expo Park, Conghua District, Guangzhou City, Guangdong Province, China (Longitude 23°59' E, Latitude 113°62' N). The greenhouse of this cherry-tomato greenhouse adopts a standardized planting mode and management mode. The images used were taken on 25 April 2024 from 1–3 p.m. The equipment used for the acquisition was a cell phone (with a 64 megapixel rear camera using 720 p resolution) and the shooting distance was about 10–60 cm. We collected 2000 original images of cherry-tomato with a resolution of 640 × 480 pixels and saved them in “.jpg” format. In order to increase the diversity of the model training samples and reduce the risk of model overfitting, this dataset consists of images taken from different angles, growth positions and different occlusion situations. Figure 1 illustrates the different situations captured, including (a) single cluster; (b) multiple clusters; (c) occlusion, where the stem is mainly lightly shaded by leaves; and (d) overlap. The acquired images were subjected to data screening in order to remove poor quality images such as transition exposures, blurring, and heavily occluded images. A final set of 1624 initial images was produced.



**Figure 1.** The main environments of cherry-tomato greenhouse, including: (a) Single cluster. (b) Multiple clusters. (c) Cover. (d) Overlap.

#### 2.1.2. Data Augmentation

In order to be able to improve the robustness of the model to some extent, this research adopts data augmentation for the original images. The diversity of the data can help the model to improve its robustness and generalization [44]. In this research, we use a variety of data enhancement methods, including horizontal flipping, panning, luminance adjustment, notching, noise addition, and barrel distortion. We randomly selected 600 images for random enhancement. Each enhancement is a random combination of the above methods and each method has a probability of 10%. As a result, 600 new samples were generated, expanding the total number of samples in the cherry-tomato dataset to 2224. As demonstrated in Figure 2, for the various data enhancement methods in a randomized combination of strategies presented in the cherry-tomato dataset.

#### 2.1.3. Dataset Details

In this research, we use LabelImg software (<https://github.com/HumanSignal/labelImg>, accessed on 20 June 2024) to label 2224 images and generate corresponding “.txt” documents, including classification information and coordinate information. There is only one class of objects, namely the stems of the cherry-tomato (labelled stem). The situation after augmenting the dataset is shown in Table 1. The dataset is divided into a training set, validation set and a test set, whose numbers are 1556, 445 and 223 sheets respectively, according to the ratio of 7:2:1. For semantic segmentation, LabelImg software is also used for the dataset to generate the corresponding masked images.

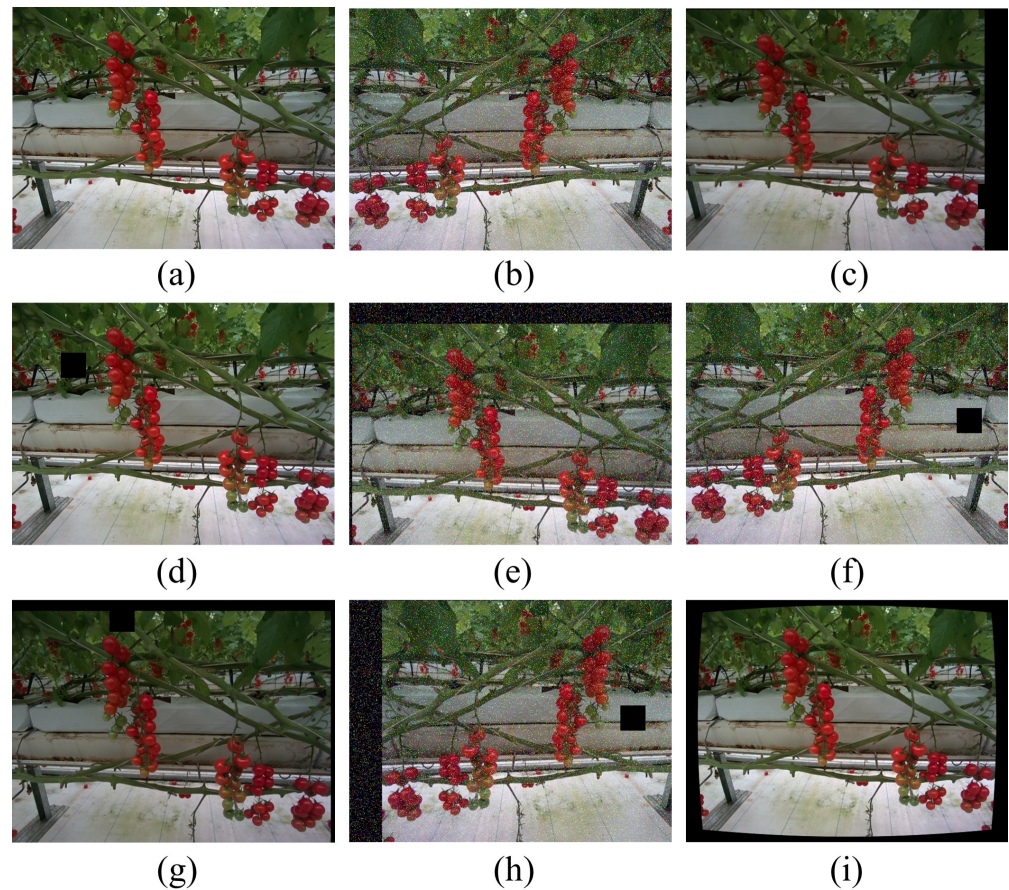


**Figure 2.** Effect of using different data augmentation methods on cherry-tomato images. (a) Original image. (b) Flip and noise-addition. (c) Shift and brightness darkening. (d) Cutout. (e) Shift. (f) Flip and cutout. (g) Shift, brightness darkening and cutout. (h) Flip, shift, brightness darkening and cutout. (i) Barrel distortion.

**Table 1.** Details dataset for the classification information of the training samples. The “stems” represent the cut points.

|            | Number of Images | Number of the Picking Point (Stems) |
|------------|------------------|-------------------------------------|
| Train      | 1556             | 6176                                |
| Validation | 445              | 2173                                |
| Test       | 223              | 1286                                |

## 2.2. Methodology

### 2.2.1. System Overview

We have established a system framework for accurately sensing the cutting point of the cherry-tomato based on the proposed YOLOv8n-DDA-SAM model. As shown in Figure 3, the framework can be regarded as two parts: the visual perception module YOLOv8n-DDA-SAM and the post-detection processing module. The visual perception module can acquire the prompt of a specific object (the stem of cherry-tomato) and obtain the corresponding mask for the post-processing module to obtain the accurate spatial location of the cherry-tomato cutting point.

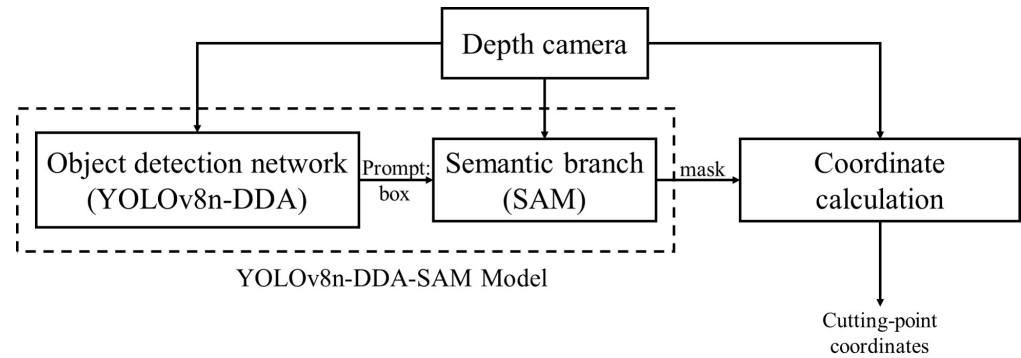


**Figure 3.** Framework of the proposed accurate cutting point detection.

### 2.2.2. YOLOv8n-DDA-SAM Algorithm Model

In our study, cherry-tomato stems are taken as the object, and the YOLOv8n-DDA-SAM network is proposed to address the characteristics of cherry-tomato stems that are slender and not easy to be detected, the difficulty of making semantic segmentation datasets, and the poor effect of semantic segmentation. To ensure that the model has a good performance, the YOLOv8n is chosen as the baseline network in this work. YOLOv8n, an anchorless frame model proposed by Jocher et al. [45] in 2023, is a lightweight parametric structure known for its fast detection speed and high accuracy [46,47]. However, the detection accuracy is not as expected for the elongated and relatively small size of the stem of the cherry-tomato. Therefore, the dynamic snake convolution was incorporated into the neck to form Bottleneck\_DySnakeConv and eventually C2f\_DSConv, making the network more focused on the slender and continuous features of the tubular structure. At the same time, deformable large kernel convolution (DLKA) was introduced in backbone in order to focus more on small targets, extend the sensory field and improve the adaptability of the model to target shape and size. Finally, we introduce the downsampling operation ADown [48] in backbone, which not only improves the detection accuracy, but also reduces the number of parameters by hundreds of thousands. As shown in Figure 4, the YOLOv8n-DDA-SAM model proposed in this research is demonstrated.

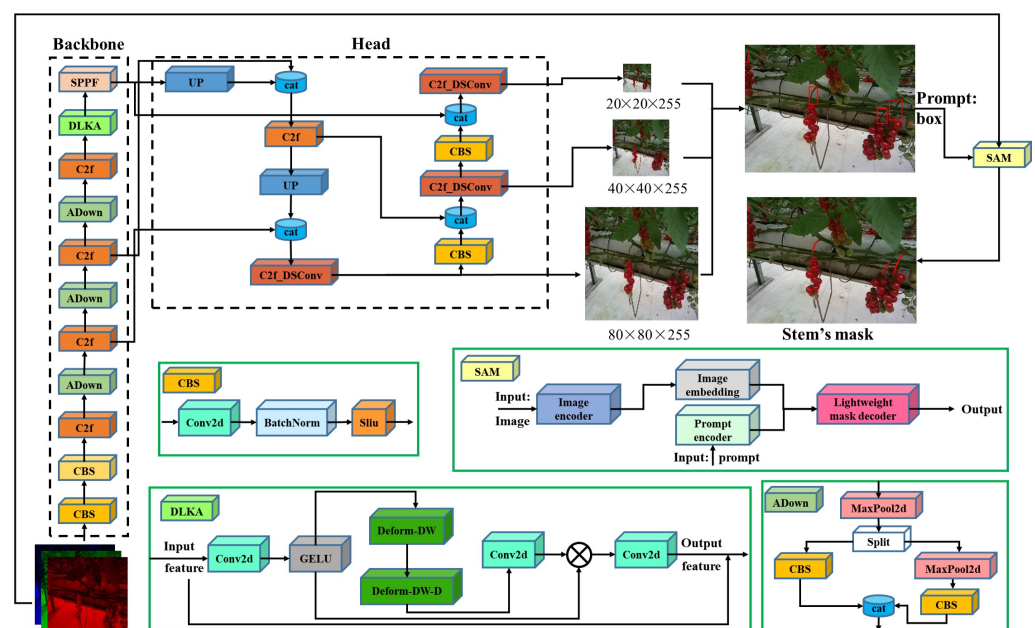


**Figure 4.** Overall structure of YOLOv8n-DDA-SAM. The CBS module consists of a 2D Convolution, a 2D BatchNorm and an SiLU activation function. ADown is a downsampling module. DLKA represents the variable large kernel attention mechanism.

### 2.2.3. Deformable Large Kernel Attention Mechanism

The Deformable Large Kernel Attention module (DLKA) is a streamlined attention mechanism that employs a large convolutional kernel to fully understand the volumetric context [49], and can flexibly distort the sampling grid, enabling the model to appropriately use different data patterns, as shown in Figure 5. It specializes in capturing far-reaching context and global context information. It combines a large convolutional kernel with deformable convolution by employing a large convolutional kernel to model a self-attention-like feeling field while avoiding the high computational cost of traditional self-attention mechanisms.

The large convolution kernel is a mechanism for capturing rich contextual information in images [50]. It imitates the sensory field of the self-attention mechanism. The large convolution kernel can be constructed efficiently by using depth-separable convolution (DW Conv) and depth-separable convolution with dilation (DW-D Conv). This allows the network to learn features within a large receptive field, while reducing the number of parameters to decrease the computational complexity. On this basis, deformable convolution is also introduced, which allows the sampling position to be adaptively adjusted according to different target scales, thus capturing features more accurately on objects of different scales [51].

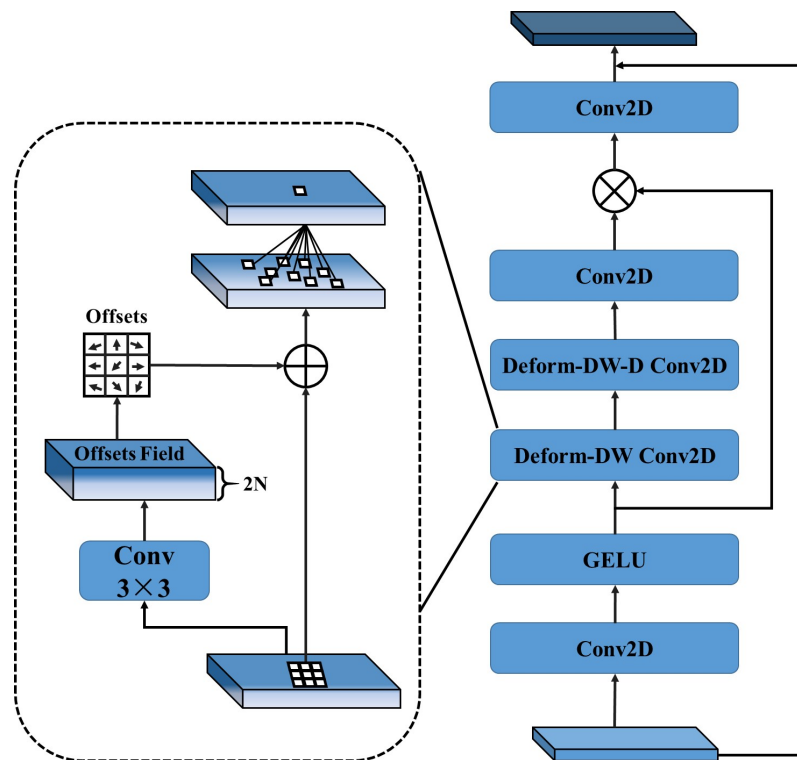


**Figure 5.** Architecture of the DLKA module. Deform-DW denotes depth divisible convolution and Deform-DW-D denotes depth divisible convolution with dilation.

### 2.2.4. Dynamic Snake Convolution

DySnakeConv [52] was first proposed in 2023, which is mainly applied to the segmentation and recognition of elongated blood vessels in the medical field; for the adaptability of non-rigid object shapes, deformable convolution kernels have higher feature extraction ability and better adaptability than traditional convolution kernels. In this research, for the research object of cherry-tomato, it's stem and blood vessels have similar characteristics both functionally and structurally, in function blood vessels and fruit stem are used to transport various substances; in structure blood vessels and fruit peduncle have the role of supporting structure. Therefore, we have added DySnakeConv to the deep learning

network for recognizing and predicting cherry-tomato in this research to extract the local features of cherry-tomato stems.

For the flexibility of the convolutional kernel to learn the complex geometric features of the target, DySnakeConv borrows the idea of DCN to introduce constrained offsets as learning parameters to let the model adaptively choose the appropriate feature extraction location for each target to be processed, so that it is not detached from the detection area due to large deformation offsets. The structure of the specific DySnakeConv convolutional kernel is shown in Figure 6.

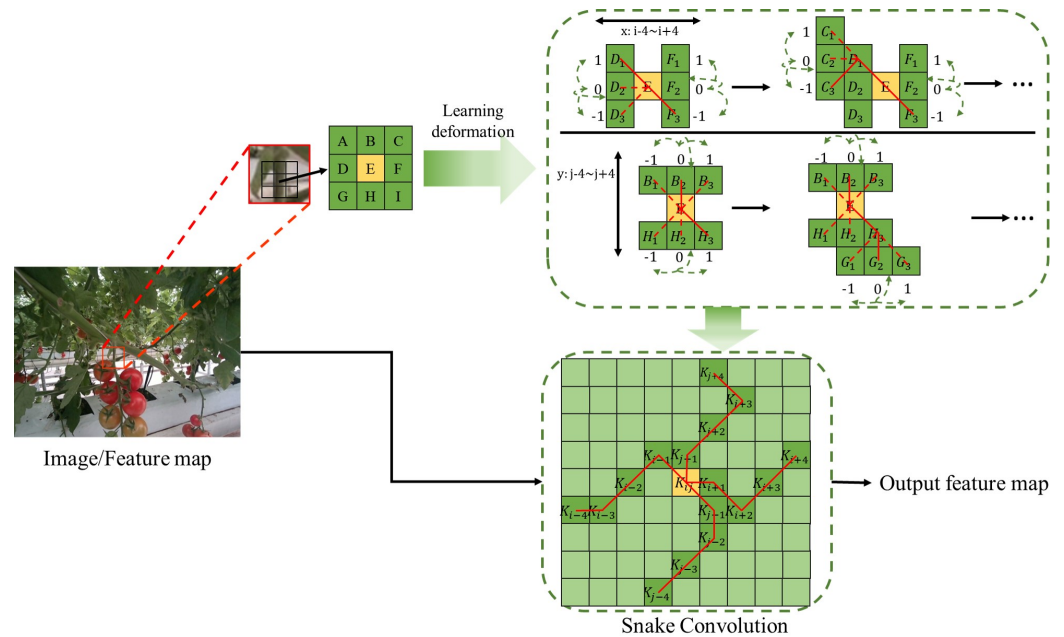


**Figure 6.** Structure of the Dynamic Serpentine Convolution. Top: The DSConv learns deformations based on the input feature mappings, and adaptively focuses on local features of slender bends. Bottom: The receptive field of the DSConv.

The setup of a typical two-dimensional  $3 \times 3$  convolution kernel  $K$  can be extended by Equation (1), where the centroid coordinates are  $K_i = (x_i, y_i)$  and the expansion coefficient is 1.

$$K = \{(x-1, y-1), (x-1, y), \dots, (x+1, y+1)\} \quad (1)$$

The standard convolution kernel  $x$  and  $y$  directions are elongated by Equations (2) and (3), thus realizing a form of deformed convolution.

In  $x$ -direction:

$$K_{i \pm c} = \begin{cases} (x_{i+c}, y_{i+c}) = (x_i + c, y_i + \sum_{i=c}^{i+c} \Delta y) \\ (x_{i-c}, y_{i-c}) = (x_i - c, y_i + \sum_{i=c}^{i-c} \Delta y) \end{cases} \quad (2)$$

In  $y$ -direction:

$$K_{j \pm c} = \begin{cases} (x_{i+c}, y_{i+c}) = (x_j + \sum_{j=c}^{j+c} \Delta x, y_i + c) \\ (x_{i-c}, y_{i-c}) = (x_j - \sum_{j=c}^{j-c} \Delta x, y_i - c) \end{cases} \quad (3)$$

where  $c$  denotes the range 0, 1, 2, 3, 4, representing the distance from the horizontal or vertical position away from the center position of the convolution kernel;  $\Delta$  denotes the offset each mesh selected position is a cumulative process. Compared to  $K_i$ ,  $K_{(i+1)}$  increases the offset  $\Delta = \{\beta \mid \beta \in (-1, 1)\}$ , and the offset summation ensures a linear morphological

structure of the convolution kernel. Since the offsets may be fractional, the convolution kernel is fitted by bilinear interpolation, implemented as follows.

$$K = \sum_{K'} B(K', K) \cdot K' \quad (4)$$

$$B(K', K) = b(K_x, K'_x) \cdot b(K_y, K'_y)$$

where  $K$  is the fractional order position of Equations (2) and (3),  $K'$  enumerates all integral space positions, and  $B$  is a bilinear interpolation kernel divided into two one-dimensional kernels.

The shape of the convolutional kernel is modified by deep learning of its offsets to make it more suitable for slender tubular structures, leading to better perception of key features of the fruit stem. It was introduced into YOLOv8n as a convolutional layer responsible for capturing deeper features and able to adaptively focus on the features of slender and curved structures. Finally, a new C2f layer is formed, called C2f\_DSConv in this research, as shown in Figure 7.

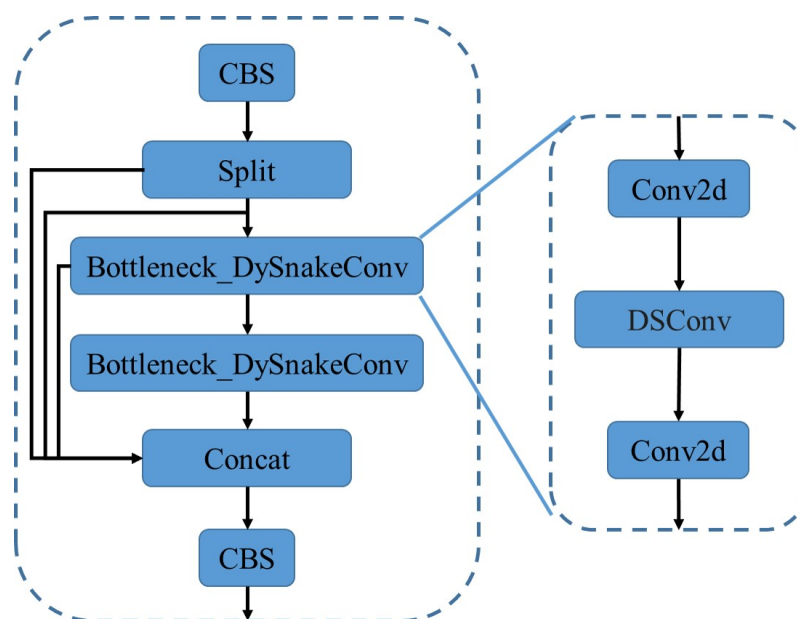


**Figure 7.** Structure of C2f\_DSConv. (Left) C2f\_DSConv. (Right) Bottleneck\_DySnakeConv.

### 2.3. Segment Branch SAM

The SAM big model was released by Meta's FAIR lab in 2023, and has received much attention in various fields since its release [53]. The SAM big model belongs to the basic model, which has been trained on millions of images and more than one billion mask (SA-1B) datasets in the early stage, and can be generalized to segment all common objects, focusing on the fast completion of the segmentation task. As shown in Figure 8, the structure of SAM has three components, namely: image encoder, prompt decoder, mask decoder, each of which has its own role, in which the image encoder is to map the image to be segmented into the image feature space; the prompt encoder maps the user input prompts into the feature space of the prompts; and the mask encoder is responsible for integrating the image encoder and the prompt encoder into the feature map and finally decoding the masks. SAM is pre-trained through a large amount of data, fusion and then use the model to label the dataset, so that the iterative cycle of pattern training can make the model has a strong generalization performance, but because SAM belongs to the pre-training AI model, so the model will not be targeted at intelligent segmentation of the cherry-tomato stem, so this research for the research object of cherry-tomato stem, target detection network prediction layer information into the SAM model prompt as a hint information to let SAM complete the segmentation for the cherry-tomato stem information.

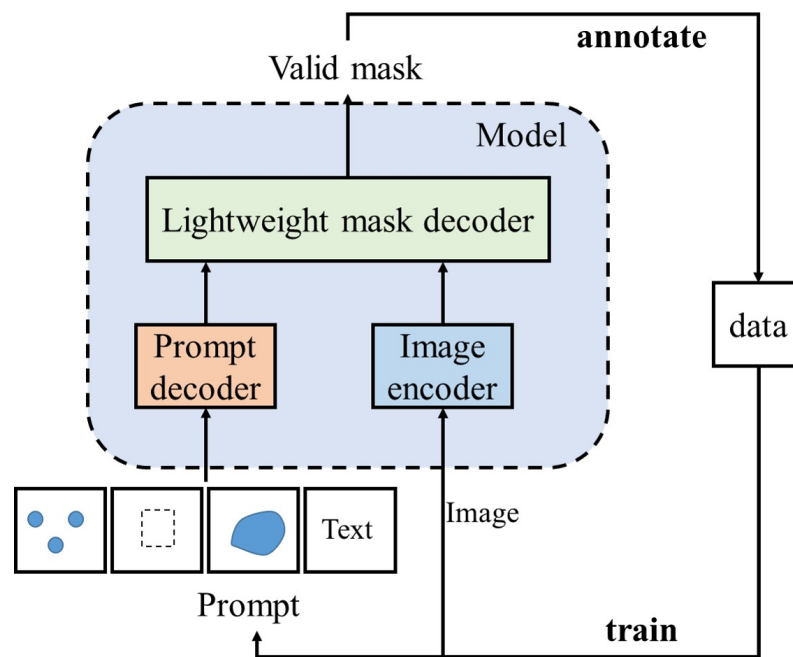


**Figure 8.** Structure of SAM.

#### 2.4. Acquire Cutting Points Based on Accurate Semantic Segmentation

In the case of the stem of the cherry-tomato, its shape is not ideally strictly perpendicular to the ground. Due to the nature of the growing environment and the influence of variable factors such as light, temperature, and water, the stem of cherry-tomato may show a certain degree of curvature. Therefore, in this subsection, the shape center of the stem is calculated based on the stem mask as the cutting point, which effectively improves the accuracy of the cutting point.

The shape center is a geometric concept that refers to the center of mass or the center of a planar image; for a planar figure, the shape center is the average position of all points within the figure. In the previous process, a high-quality mask of the fruit stem is obtained, which provides a reliable basis for obtaining the optimal cutting point later. The area of the stalk mask is calculated using Opencv technology, and the following formula is used to obtain the position of the center of the shape:

$$X_c = \frac{\sum_{i=1}^n x_i}{n} \quad (5)$$

$$Y_c = \frac{\sum_{i=1}^n y_i}{n} \quad (6)$$

The coordinates of the shaped center are  $(X_c, Y_c)$ ,  $n$  is the number of points on the contour, and  $(X_i, Y_i)$  are the coordinates on the contour of the stem. The shape center of the stem is obtained using its position as the cutting point as shown in Figure 9.

#### 2.5. Evaluation Metrics

In order to evaluate the performance of the model, metrics such as precision (P), recall (R), and mean accuracy (mAP@0.5) are used in target detection, and mean intersection and merger ratio (mIOU) is also used in semantic segmentation. The above metrics are calculated as follows:

$$P = \frac{TP}{TP + FP} \quad (7)$$

$$R = \frac{TP}{TP + FN} \quad (8)$$

$$AP = \int_0^1 P(R) dR \quad (9)$$

$$mAP = \frac{\sum_{i=1}^n AP_i}{n} \quad (10)$$

$$mIOU = \frac{1}{k+1} \sum_{i=0}^k \frac{p_{ii}}{\sum_{i=0}^k p_{ij} + \sum_{i=0}^k p_{ji} - p_{ii}} \quad (11)$$

where,  $TP$  is the number of positive samples predicted as positive categories,  $FP$  is the number of negative samples predicted as positive categories,  $FN$  is the number of positive samples predicted as negative categories,  $AP$  is the area under the PR curve, and  $mAP@0.5$  is the average of the  $AP$  of each category.  $n$  denotes the number of categories detected by the object. In semantic segmentation metrics,  $mIOU$  is the average of the ratio of the intersection and concatenation ratio of all categories computed.  $p_{ii}$  is the prediction of  $i$  as  $i$  that is true.  $p_{ij}$  is the prediction of  $i$  as  $j$  that is false-negative.  $p_{ji}$  is the prediction of  $j$  as  $i$  that is false-positive. In this research,  $k = n = 1$ .



**Figure 9.** Calculation of cutting points for clusters of multiple bunches of cherry-tomato. The red areas are masks for semantic segmentation, and the small green boxes are computed picking points.

## 2.6. Training Parameters and Experimental Environment

The experimental setup in this research is based on the deep learning framework PyTorch, with the version of 1.12.1+cu116, Python3.8, and was conducted on a Windows 10 operating system. The processor used was Intel(R) Core(TM) i7-10700 CPU@2.90 GHz, with 32 GB of memory. The graphics card model was NVIDIA GeForce RTX 3090Ti and CUDA 11.6 and CUDNN 8.6.0 were used to accelerate GPU computation. The specific configuration is organized in Table 2. The training parameters used in the experiments are detailed in Table 3.

**Table 2.** Experimental environment settings.

| Parameter               | Configuration                        |
|-------------------------|--------------------------------------|
| Operating system        | Windows 10                           |
| Deep learning framework | Torch1.12.1+cu116                    |
| Programming language    | Python3.8                            |
| GPU                     | NVIDIA GeForce RTX 3090Ti            |
| CPU                     | Intel(R) Core(TM) i7-10700 @2.90 GHz |

**Table 3.** The parameters in the training process.

| Parameter             | Configuration                            |
|-----------------------|------------------------------------------|
| Epoch                 | 200                                      |
| Initial learning rate | 0.01                                     |
| Batch size            | 16                                       |
| momentum              | 0.98 (YOLOv7 and YOLOv9c)/0.937 (others) |
| Weight decay          | 0.0005                                   |

### 3. Results

#### 3.1. Comparison of Reception Fields and the Heat Map

In this subsection we will investigate whether the DKLA attention mechanism has an effect on expanding the receptive field in the model, and whether the dynamic snake convolution enables the model to pay more attention to the elongated stem part of the cherry-tomato. As shown in Figure 10, the receptive fields and heat maps before and after the improvement are demonstrated.

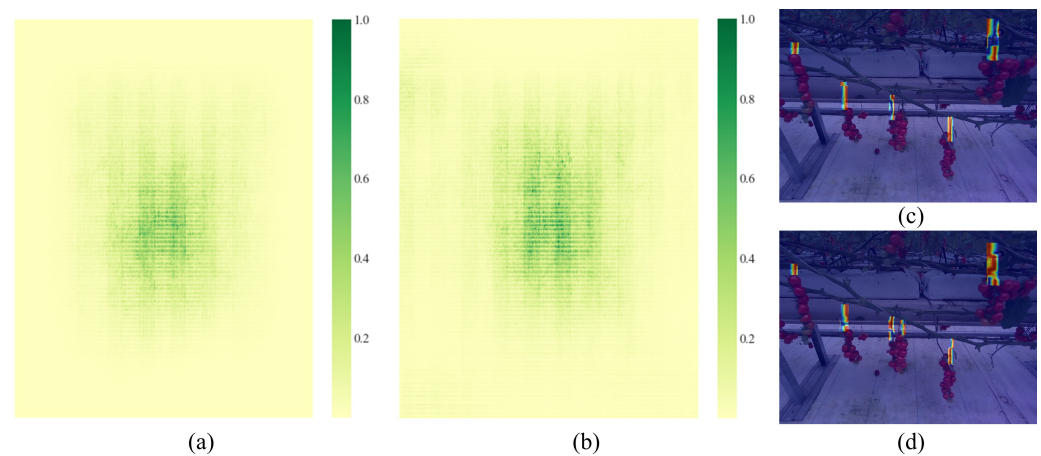


**Figure 10.** Receptive fields and heat maps before and after model improvement. (a) Receptive field region before improvement. (b) Receptive field region after introduction of DLKA attention mechanism. (c) Heat map before improvement (without more focus on stem area). (d) Heat map after introduction of dynamic snake convolution (with more focus on peduncle area). The colors in (c,d) represent the model paying more attention to this region, with redder representing more attention.

We can clearly see that the model receptive field is indeed improved after the introduction of the DKLA attention mechanism, according to [54], we datamined the receptive field as shown in Table 4. The smoother the distribution of high contribution pixels after introducing it. From the heat map analysis, it can be seen that the dynamic snake convolution makes the model pay more attention to the information of the slender fruit stems, focuses the attention on the effective region, and accurately extracts the feature information so that it can improve the detection efficiency of the model to a certain extent.

**Table 4.** Quantitative analysis of receptive fields. Higher ratios indicate larger receptive fields.

|              | t = 20% | t = 30% | t = 50% | t = 99% |
|--------------|---------|---------|---------|---------|
| YOLOv8n      | 3.01%   | 4.99%   | 10.26%  | 63.25%  |
| YOLOv8n-DLKA | 3.94%   | 6.33%   | 13.48%  | 95.98%  |

### 3.2. Experiments Were Conducted to Evaluate the Performance of YOLOv8n-DDA-SAM

To illustrate the detection and semantic segmentation performance of the YOLOv8n-DDA-SAM model, we perform a series of comparative experiments in this subsection.

#### 3.2.1. Object Detection Branch

In the target detection branch, we compare several mainstream neural network models, such as the original YOLOv8n, YOLOv7, RT-DETR-l, and YOLOv9c, and finally we will compare their performance on metrics such as Precision, Recall, mAP@0.5, and F1-score. The results are shown in Table 5.

**Table 5.** Comparison of different detection algorithms.

| Model       | Precision | Recall | mAP@0.5 | F1-Score | Processing Time per Photo (ms) | Param (M) |
|-------------|-----------|--------|---------|----------|--------------------------------|-----------|
| YOLOv7      | 76.30%    | 63.92% | 61.43%  | 69.56%   | 36.0                           | 37.20     |
| YOLOv8n     | 80.43%    | 68.17% | 64.28%  | 73.79%   | 20.00                          | 3.01      |
| RT-DETR-l   | 82.42%    | 70.18% | 66.37%  | 75.81%   | 44.0                           | 31.99     |
| YOLOv9c     | 84.09%    | 72.43% | 70.14%  | 77.83%   | 51.5                           | 50.70     |
| YOLOv8n-DDA | 90.64%    | 81.64% | 86.13%  | 85.90%   | 35.17                          | 6.37      |

As shown in Table 5, the YOLOv8n-DDA-SAM proposed in this research demonstrates its excellent performance in the target detection when compared to several more mainstream network models. In terms of accuracy, YOLOv8n-DDA improves by 14.34%, 10.21%, 8.22%, and 6.55%, respectively. Compared to YOLOv7, YOLOv8n, RT-DETR-l, and YOLOv9c network models. In terms of checking completeness, it was improved by 17.72%, 13.47%, 11.46%, and 9.21%, respectively. In terms of average precision, it improved by 24.7%, 21.85%, 19.76, and 15.99%, respectively. In terms of F1 score, there is an improvement of 16.34%, 12.11%, 10.09%, and 8.07%, respectively. As shown in Figure 11, this research also gives the results of the experiment with 200 epochs of training, including the mAP@0.5, inference time and F1-score metrics. As shown in Figure 12, the results of YOLOv7, YOLOv8n, RT-DETR-l, YOLOv9c, and YOLOv8n-DDA-SAM models for the same image are demonstrated.

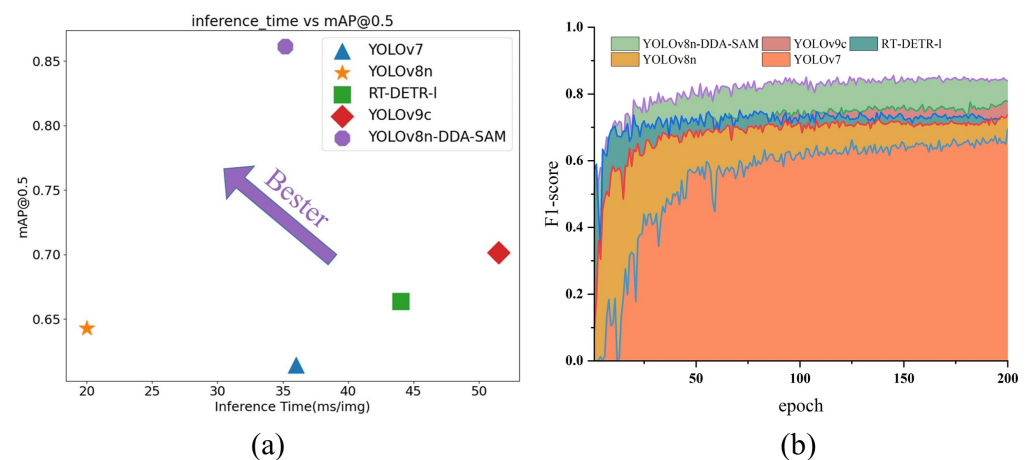


**Figure 11.** F1-score, mAP@0.5 and inference time achieved by five models YOLOv7, YOLOv8n, RT-DETR-l, YOLOv9c, and YOLOv8n-DDA-SAM model after 200 epochs training. (a) mAP@0.5 and inference time graph. (b) F1-score graph.

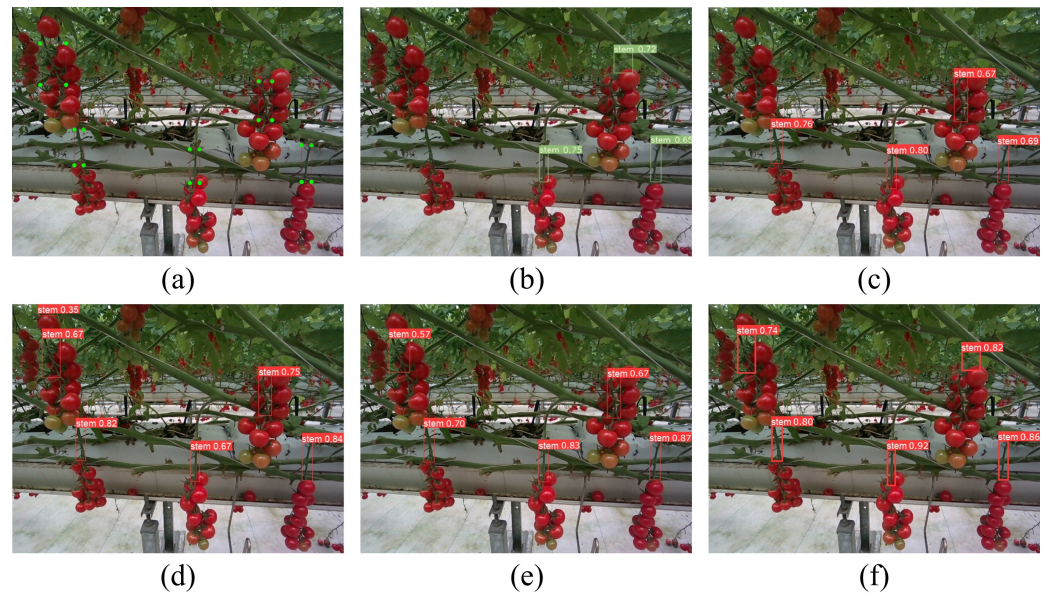


**Figure 12.** Results of YOLOv7, YOLOv8n, RT-DETR-l, YOLOv9c, and YOLOv8n-DDA-SAM models for the same image. (a) Ground truth. (b) YOLOv7. (c) YOLOv8n. (d) RT-DETR-l. (e) YOLOv9c. (f) YOLOv8n-DDA-SAM.

### 3.2.2. Semantic Segmentation Branch

Similarly, in the semantic segmentation branch, the performance of the mainstream algorithms Deeplabv3+, Mask2former, DDRNet, SAN in terms of P, R, mAP@0.5, and mIOU metrics were compared. The results are presented in Table 6.

**Table 6.** Comparison of different segmentation algorithms.

| Model       | Precision | Recall | mAP@0.5 | mIOU   | Process per Photo (s) |
|-------------|-----------|--------|---------|--------|-----------------------|
| Deeplabv3+  | 72.72%    | 61.65% | 80.79%  | 74.92% | 3.89                  |
| Mask2former | 76.41%    | 71.34% | 85.63%  | 79.41% | 3.25                  |
| DDRNet      | 77.82%    | 73.11% | 86.72%  | 80.82% | 3.26                  |
| SAN         | 80.14%    | 75.37% | 87.13%  | 82.13% | 3.84                  |
| Ours        | 89.95%    | 86.30% | 93.12%  | 86.35% | 5.68                  |

As shown in Table 6, the YOLOv8n-DDA-SAM proposed in this research demonstrates its excellent performance in the semantic segmentation branch comparing several more mainstream network models. In terms of accuracy, compared with Deeplabv3+, Mask2former, DDRNet and SAN network models, it is improved by 17.23%, 13.54%, 12.13% and 9.81% respectively. In terms of checking completeness, it is improved by 24.65%, 14.96%, 13.19% and 10.93% respectively. In average accuracy, it improved by 12.33%, 7.49%, 6.4% and 5.99% respectively. In mIOU, it was improved by 11.43%, 6.94%, 5.53% and 4.22% respectively. On processing each image, our semantic segmentation branches SAM, Deeplabv3+, Mask2former, DDRNet and SAN are 5.68s, 3.89s, 3.25s, 3.26s and 3.84s respectively. On this basis, we demonstrate its semantic segmentation effect on the stems of cherry-tomato, as shown in Figure 13.

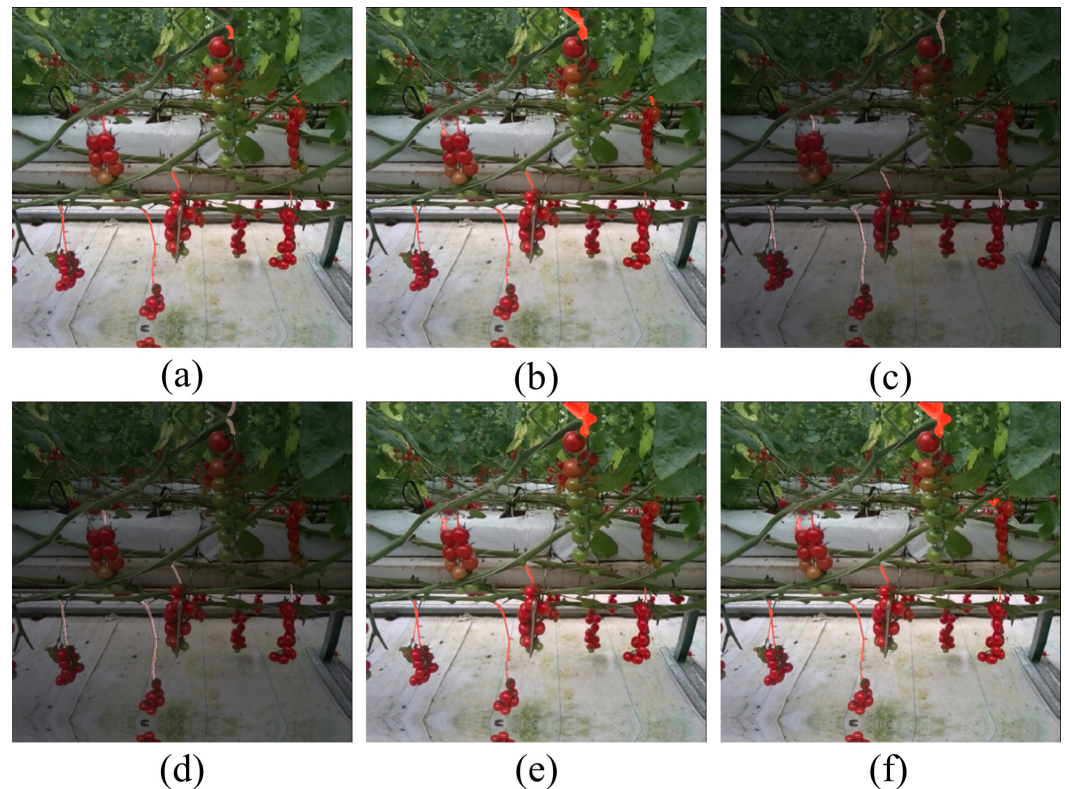


**Figure 13.** Demonstration of the effect of Deeplabv3+, Mask2former, and SAM on semantic segmentation. (a) Ground truth. (b) SAM. (c) Deeplabv3+. (d) Mask2former. (e) DDRNet. (f) SAN.

### 3.3. Ablation Experiments with YOLOv8n-DDA

To verify the effect of DLKA, DSConv, and ADown introduced in this research on the model, the ablation test is performed by incremental setting for the standard original YOLO. The effects are shown in Table 7.

**Table 7.** Ablation Study on DSConv, DLKA and ADown in YOLOv8n-DDA. Values represent the percentage (%) of Precision, Recall, mAP@0.5, F1-Score and Param. YOLOv8D represents YOLOv8n after the introduction of dynamic snake convolution, sequentially postfixes as DD's represent the former introducing the large kernel convolutional attention mechanism (DLKA).

| Model Abbreviation | Precision | R      | mAP@0.5 | F1-Score | Param (M) |
|--------------------|-----------|--------|---------|----------|-----------|
| YOLOv8n            | 80.43%    | 68.17% | 64.28%  | 73.79%   | 3.01      |
| YOLOv8n-D          | 85.27%    | 72.19% | 67.13%  | 78.19%   | 3.34      |
| YOLOv8n-DD         | 88.51%    | 77.93% | 71.62%  | 82.88%   | 6.65      |
| YOLOv8n-DDA        | 90.64%    | 81.64% | 86.13%  | 85.90%   | 6.37      |

From the above table, it can be seen that based on the YOLOv8n algorithm in which dynamic serpentine convolution is introduced into the neck to form C2f\_DSConv, and the algorithm YOLOv8n-D is formed, compared with the initial YOLOv8n, the detection of cherry-tomato stems improves by 4.84%, 4.02%, 2.85%, and 4.4% in the precision rate, recall rate, average precision, and F1-score, respectively. This proves that the use of dynamic serpentine convolution, which is sensitive to the elongated tubular topology, instead of the convolutional layer of the model's high-level feature extraction can more accurately detect the stems of cherry-tomato. We also introduce the deformable large kernel convolutional attention mechanism, which enables the model to perform feature extraction with a larger sense field, and the experiments show that the introduction of this mechanism can again improve the performance of the model, compared with YOLOv8n-D, YOLOv8n-DD improves the precision, recall, average precision, and F1-score by 3.24%,

5.74%, 4.49% and 4.69%, respectively. However, it can be clearly seen that the number of parameters has doubled compared to the initial YOLOv8n, for which we introduced a more efficient downsampling operation ADown, and finally formed the YOLOv8n-DDA detection model branch, and its precision, recall, mAP@0.5, and F1-score reached 90.64%, 81.64%, 86.13%, and 85.90%. And the number of parameters is also reduced by 280,000 compared to YOLOv8n-DD. In summary, the performance of the YOLOv8n-DDA target detection branch has been very obvious. As shown in Figure 14, the F1-score plot in the ablation experiment is shown.

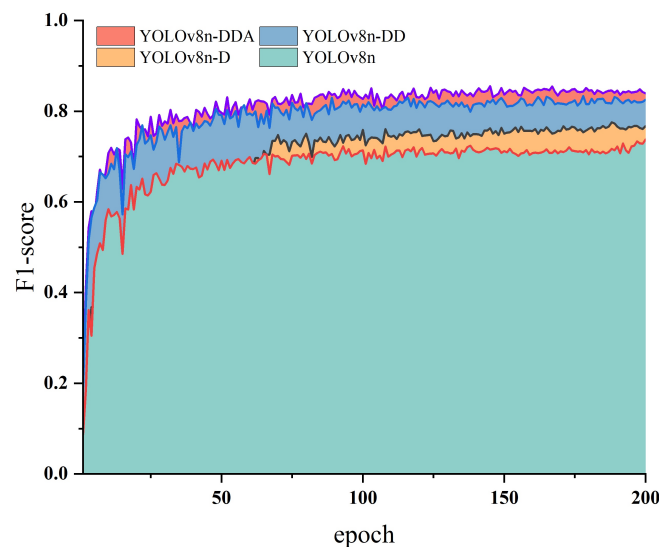


**Figure 14.** F1-score performance curve of YOLOv8n-DDA for each improvement point in the model.

#### 4. Discussion

In this research, for the problems such as the slender and small volume of the stem of the cherry-tomato, we introduce a large kernel convolutional attention mechanism and dynamic snake convolution with a lightweight downsampling operation ADown. In addition, we add a semantic segmentation branch on the YOLOv8n base network, fuse the target detection branch, and finally form the network model YOLOv8n-DDA-SAM network model. It is shown that our proposed YOLOv8n-DDA-SAM model can effectively detect and segment the stems of cherry-tomato, and obtain the localization information with simple shape centers under the premise of accuracy. This lays a solid foundation for subsequently guiding the picking robot to perform picking maneuvers.

In order to verify the performance of the target detection branch, we compare it with YOLOv7, YOLOv8n, RT-DETR-l, and YOLOv9c deep learning networks for target detection, and the experimental results show that the model achieves a balance between precision and speed with a significant improvement in detection precision and recall. In this, we also compare the semantic segmentation branch with several mainstream networks, and in terms of precision, compared with Deeplabv3+, Mask2former, DDRNet and SAN network models, it improves by 17.23%, 13.54%, 12.13% and 9.81% respectively. In terms of checking completeness, it is improved by 24.65%, 14.96%, 13.19% and 10.93% respectively. In average accuracy, it improved by 12.33%, 7.49%, 6.4% and 5.99% respectively. In terms of mIOU, it has been improved by 11.43%, 6.94%, 5.53% and 4.22% respectively. Most pleasingly, the branch does not require the production of semantic segmentation dataset.

Despite some significant achievements, it must be recognized that our research has some limitations. For example, the performance of semantic segmentation is very much dependent on the effectiveness of target detection, and due to the picking environment, our method may suffer from problems such as omissions. And our computation of localization information is also very much dependent on the performance of detection and segmentation, for example, the recognition and segmentation of a fruit stalk may have

misdetec-tion, leading to the duplication of picking information and loss of picking time. Therefore, in future work, the algorithm needs to be further optimized to be more compatible with unstructured environments and to optimize its localization algorithm, especially in dense environments.

## 5. Conclusions and Future Work

In this study, we propose a cherry-tomato stem detection method named YOLOv8n-DDA-SAM, which has a target detection branch and a semantic segmentation branch, aiming to accurately recognizing the stem of cherry-tomato and their masks and 3D locations. We employed various data enhancement techniques in the collected images including horizontal flipping, panning, brightness adjustment, notching, noise addition and barrel distortion. These enhancement methods contribute to the diversity of the dataset, which improves the effectiveness of the learning process and the robustness of the model. Compared to the original YOLOv8n, YOLOv7, RT-DETR-L, and YOLOv9c, YOLOv8n-DDA-SAM improves the average accuracy by 24.7%, 21.85%, 19.76% and 15.99% respectively, and in terms of the F1 scores, it is even better by 16.34%, 12.11%, 10.09% and 8.07% respectively. The introduction of DLKA allows the model to perform feature extraction over a larger receptive field, and the introduction of dynamic snake convolution allows the network to be more sensitive to the slender stems of the cherry-tomato. The ablation experiments further emphasize the impact of DLKA and dynamic snake convolution on the model, which allows YOLOv8n-DDA-SAM to demonstrate its excellent performance on various metrics. Then on this basis, our semantic segmentation branch also achieves ideal results, compared with Deeplabv3+, Mask2former, DDRNet and SAN, its mIOU improves by 11.43%, 6.94%, 5.54%, 4.22%, mAP@0.5 improves by 12.33% and 7.49%, 6.4%, 5.99% respectively. For the semantic segmentation branch, which does not need to produce a dataset and has excellent segmentation performance, complemented by the simple algorithm of Opencv, the three-dimensional localization information of the fruit stem of the cherry-tomato is pretty accurate.

Future work will involve algorithm optimization for inference time and the production of high quality datasets. More data sets will be collected and more segmentation networks tested. In addition, robotic arms will be installed on mobile devices suitable for the cherry-tomato growing environment and corresponding picking experiments conducted in real orchard environments.

**Author Contributions:** Conceptualization, G.Z. and H.W.; Methodology, G.Z., H.C., Y.J. and X.Z.; Software, G.Z. and H.C.; Validation, G.Z., H.C., Y.J., Y.Z. and A.Z.; Formal analysis, Y.Z. and A.Z.; Data curation, G.Z.; Writing-original draft, G.Z.; Writing review & editing, H.W.; Supervision, H.W.; Project administration, H.W.; funding acquisition, H.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported in part by the National Natural Science Foundation of China under Grant 32372001 and Guangzhou Science and Technology Project (2023B01J0046).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author.

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

## References

1. Tang, Q.; Zhu, F.; Cao, X.; Zheng, X.; Yu, T.; Lu, L. *Cryptococcus laurentii* controls gray mold of cherry tomato fruit via modulation of ethylene-associated immune responses. *Food Chem.* **2019**, *278*, 240–247. [[CrossRef](#)] [[PubMed](#)]
2. Zhang, L.; Chen, F.; Zhang, P.; Lai, S.; Yang, H. Influence of Rice Bran Wax Coating on the Physicochemical Properties and Pectin Nanostructure of Cherry Tomatoes. *Food Bioprocess Technol.* **2017**, *10*, 349–357. [[CrossRef](#)]

3. Li, Y.R.; Lien, W.Y.; Huang, Z.H.; Chen, C.T. Hybrid Visual Servo Control of a Robotic Manipulator for Cherry Tomato Harvesting. *Actuators* **2023**, *12*, 253. [\[CrossRef\]](#)
4. Hou, G.; Chen, H.; Ma, Y.; Jiang, M.; Hua, C.; Jiang, C.; Niu, R. An occluded cherry tomato recognition model based on improved YOLOv7. *Front. Plant Sci.* **2023**, *14*, 1260808. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Barnett, J.; Duke, M.; Au, C.K.; Lim, S.H. Work distribution of multiple Cartesian robot arms for kiwifruit harvesting. *Comput. Electron. Agric.* **2020**, *169*, 105202. [\[CrossRef\]](#)
6. Zahid, A.; Mahmud, M.S.; He, L.; Heinemann, P.; Choi, D.; Schupp, J. Technological advancements towards developing a robotic pruner for apple trees: A review. *Comput. Electron. Agric.* **2021**, *189*, 106383. [\[CrossRef\]](#)
7. Zheng, H.; Wang, G.; Li, X. YOLOX-Dense-CT: A detection algorithm for cherry tomatoes based on YOLOX and DenseNet. *J. Food Meas. Charact.* **2022**, *16*, 4788–4799. [\[CrossRef\]](#)
8. Khoshroo, A.; Arefi, A.; Khodaei, J. Detection of Red Tomato on Plants using Image Processing Techniques. *Agric. Commun.* **2014**, *2*, 9–15.
9. Omid-Arjenaki, O.; Moghaddam, P.; Motlagh, A. Online tomato sorting based on shape, maturity, size, and surface defects using machine vision. *Turk. J. Agric. For.* **2012**, *37*, 62–68.
10. Zhu, X.; Chen, F.; Zheng, Y.; Peng, X.; Chen, C. An efficient method for detecting Camellia oleifera fruit under complex orchard environment. *Sci. Hortic.* **2024**, *330*, 113091. [\[CrossRef\]](#)
11. Liu, X.; Jing, X.; Jiang, H.; Younas, S.; Wei, R.; Dang, H.; Wu, Z.; Fu, L. Performance evaluation of newly released cameras for fruit detection and localization in complex kiwifruit orchard environments. *J. Field Robot.* **2023**, *41*, 881–894. [\[CrossRef\]](#)
12. Situ, Z.; Teng, S.; Liao, X.; Chen, G.; Zhou, Q. Real-time sewer defect detection based on YOLO network, transfer learning, and channel pruning algorithm. *J. Civ. Struct. Health Monit.* **2023**, *14*, 41–57. [\[CrossRef\]](#)
13. Bello, R.; Oladipo, M. Mask YOLOv7-Based Drone Vision System for Automated Cattle Detection and Counting. *Artif. Intell. Appl.* **2024**, *2*, 129–139. [\[CrossRef\]](#)
14. Akkar, A.; Cregan, S.; Cassens, J.; Vander-Pallen, M.; Khan Mohd, T. Playing Blackjack Using Computer Vision. *Artif. Intell. Appl.* **2023**. [\[CrossRef\]](#)
15. Meng, F.; Li, J.; Zhang, Y.; Qi, S.; Tang, Y. Transforming unmanned pineapple picking with spatio-temporal convolutional neural networks. *Comput. Electron. Agric.* **2023**, *214*, 108298.
16. Pavithra, V.; Pounroja, R.; Bama, B.S. Machine vision based automatic sorting of cherry tomatoes. In Proceedings of the 2015 2nd International Conference on Electronics and Communication Systems (ICECS), Coimbatore, India, 26–27 February 2015; pp. 271–275.
17. Liu, G.; Mao, S.; Kim, J.H. A Mature-Tomato Detection Algorithm Using Machine Learning and Color Analysis. *Sensors* **2019**, *19*, 2023. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Fan, S.; Liang, X.; Huang, W.; Zhang, V.J.; Pang, Q.; He, X.; Li, L.; Zhang, C. Real-time defects detection for apple sorting using NIR cameras with pruning-based YOLOV4 network. *Comput. Electron. Agric.* **2022**, *193*, 106715. [\[CrossRef\]](#)
19. Mamat, N.; Othman, M.F.; Abdulghafor, R.; Alwan, A.A.; Gulzar, Y. Enhancing Image Annotation Technique of Fruit Classification Using a Deep Learning Approach. *Sustainability* **2023**, *15*, 901. [\[CrossRef\]](#)
20. Rebahi, Y.; Gharra, M.; Rizzi, L.; Zournatzis, I. Combining Computer Vision, Artificial Intelligence and 3D Printing in Wheelchair Design Customization: The Kyklos 4.0 Approach. *Artif. Intell. Appl.* **2023**. [\[CrossRef\]](#)
21. Tang, Y.; Qi, S.; Zhu, L.; Zhuo, X.; Zhang, Y.; Meng, F. Obstacle Avoidance Motion in Mobile Robotics. *J. Syst. Simul.* **2024**, *36*, 108453.
22. Lei, Y.; Wu, F.; Zou, X.; Li, J. Path planning for mobile robots in unstructured orchard environments: An improved kinematically constrained bi-directional RRT approach. *Comput. Electron. Agric.* **2023**, *215*, 108453.
23. Wang, C.; He, W.; Nie, Y.; Guo, J.; Liu, C.; Wang, Y.; Han, K. Gold-YOLO: Efficient Object Detector via Gather-and-Distribute Mechanism. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 10–16 December 2023; Curran Associates, Inc.: Red Hook, NY, USA, 2023; Volume 36, pp. 51094–51112.
24. Zhang, Y.; Zhang, H.; Huang, Q.; Han, Y.; Zhao, M. DsP-YOLO: An anchor-free network with DsPAN for small object detection of multiscale defects. *Expert Syst. Appl.* **2024**, *241*, 122669. [\[CrossRef\]](#)
25. Li, H.; Gu, Z.; He, D.; Wang, X.; Huang, J.; Mo, Y.; Li, P.; Huang, Z.; Wu, F. A lightweight improved YOLOv5s model and its deployment for detecting pitaya fruits in daytime and nighttime light-supplement environments. *Comput. Electron. Agric.* **2024**, *220*, 108914. [\[CrossRef\]](#)
26. Gulzar, Y. Fruit Image Classification Model Based on MobileNetV2 with Deep Transfer Learning Technique. *Sustainability* **2023**, *15*, 1906. [\[CrossRef\]](#)
27. Gao, F.; Fang, W.; Sun, X.; Wu, Z.; Zhao, G.; Li, G.; Li, R.; Fu, L.; Zhang, Q. A novel apple fruit detection and counting methodology based on deep learning and trunk tracking in modern orchard. *Comput. Electron. Agric.* **2022**, *197*, 107000. [\[CrossRef\]](#)
28. Phan, Q.H.; Nguyen, V.T.; Lien, C.H.; Duong, T.P.; Hou, M.T.K.; Le, N.B. Classification of Tomato Fruit Using Yolov5 and Convolutional Neural Network Models. *Plants* **2023**, *12*, 790. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Yuan, T.; Lv, L.; Zhang, F.; Fu, J.; Gao, J.; Zhang, J.; Li, W.; Zhang, C.; Zhang, W. Robust Cherry Tomatoes Detection Algorithm in Greenhouse Scene Based on SSD. *Agriculture* **2020**, *10*, 160. [\[CrossRef\]](#)
30. Chen, J.; Wang, Z.; Wu, J.; Hu, Q.; Zhao, C.; Tan, C.; Teng, L.; Luo, T. An improved Yolov3 based on dual path network for cherry tomatoes detection. *J. Food Process. Eng.* **2021**, *44*, e13803. [\[CrossRef\]](#)

31. Fuentes-Peñailillo, F.; Carrasco Silva, G.; Pérez Guzmán, R.; Burgos, I.; Ewertz, F. Automating Seedling Counts in Horticulture Using Computer Vision and AI. *Horticulturae* **2023**, *9*, 1134. [\[CrossRef\]](#)
32. Kim, J.; Pyo, H.; Jang, I.; Kang, J.; Ju, B.; Ko, K. Tomato harvesting robotic system based on Deep-ToMaToS: Deep learning network using transformation loss for 6D pose estimation of maturity classified tomatoes with side-stem. *Comput. Electron. Agric.* **2022**, *201*, 107300. [\[CrossRef\]](#)
33. Kang, H.; Chen, C. Fast implementation of real-time fruit detection in apple orchards using deep learning. *Comput. Electron. Agric.* **2020**, *168*, 105108. [\[CrossRef\]](#)
34. Wang, X.; Kang, H.; Zhou, H.; Au, W.; Chen, C. Geometry-aware fruit grasping estimation for robotic harvesting in apple orchards. *Comput. Electron. Agric.* **2022**, *193*, 106716. [\[CrossRef\]](#)
35. Tsai, F.T.; Nguyen, V.T.; Duong, T.P.; Phan, Q.H.; Lien, C.H. Tomato Fruit Detection Using Modified YOLOv5m Model with Convolutional Neural Networks. *Plants* **2023**, *12*, 3067. [\[CrossRef\]](#) [\[PubMed\]](#)
36. Solimani, F.; Cardellicchio, A.; Dimauro, G.; Petrozza, A.; Summerer, S.; Cellini, F.; Renò, V. Optimizing tomato plant phenotyping detection: Boosting YOLOv8 architecture to tackle data complexity. *Comput. Electron. Agric.* **2024**, *218*, 108728. [\[CrossRef\]](#)
37. Yue, X.; Qi, K.; Yang, F.; Na, X.; Liu, Y.; Liu, C. RSR-YOLO: A real-time method for small target tomato detection based on improved YOLOv8 network. *Discov. Appl. Sci.* **2024**, *6*, 268. [\[CrossRef\]](#)
38. Yan, Y.; Zhang, J.; Bi, Z.; Wang, P. Identification and Location Method of Cherry Tomato Picking Point Based on Si-YOLO. In Proceedings of the 2023 IEEE 13th International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Qinhuangdao, China, 11–14 July 2023; pp. 373–378.
39. Yang, G.; Wang, J.; Nie, Z.; Yang, H.; Yu, S. A Lightweight YOLOv8 Tomato Detection Algorithm Combining Feature Enhancement and Attention. *Agronomy* **2023**, *13*, 1824. [\[CrossRef\]](#)
40. Zhang, F.; Chen, Z.; Ali, S.; Yang, N.; Fu, S.; Zhang, Y. Multi-class detection of cherry tomatoes using improved YOLOv4-tiny model. *Int. J. Agric. Biol. Eng.* **2023**, *16*, 225–231.
41. Zu, L.; Zhao, Y.; Liu, J.; Su, F.; Zhang, Y.; Liu, P. Detection and Segmentation of Mature Green Tomatoes Based on Mask R-CNN with Automatic Image Acquisition Approach. *Sensors* **2021**, *21*, 7842. [\[CrossRef\]](#) [\[PubMed\]](#)
42. Fawzia Rahim, U.; Mineno, H. Highly Accurate Tomato Maturity Recognition: Combining Deep Instance Segmentation, Data Synthesis and Color Analysis. In Proceedings of the 2021 4th Artificial Intelligence and Cloud Computing Conference, Kyoto, Japan, 17–19 December 2021; Association for Computing Machinery: New York, NY, USA, 2022; pp. 16–23.
43. Yoshida, T.; Fukao, T.; Hasegawa, T. A Tomato Recognition Method for Harvesting with Robots using Point Clouds. In Proceedings of the 2019 IEEE/SICE International Symposium on System Integration (SII), Paris, France, 14–16 January 2019; pp. 456–461.
44. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 60. [\[CrossRef\]](#)
45. Jocher, G.; Chaurasia, A.; Qiu, J. Yolo by Ultralytics. Available online: <https://github.com/ultralytics/ultralytics> (accessed on 2 January 2023).
46. Wang, C.; Wang, C.; Wang, L.; Wang, J.; Liao, J.; Li, Y.; Lan, Y. A Lightweight Cherry Tomato Maturity Real-Time Detection Algorithm Based on Improved YOLOV5n. *Agronomy* **2023**, *13*, 2106. [\[CrossRef\]](#)
47. Gao, J.; Zhang, J.; Zhang, F.; Gao, J. LACTA: A lightweight and accurate algorithm for cherry tomato detection in unstructured environments. *Expert Syst. Appl.* **2024**, *238*, 122073. [\[CrossRef\]](#)
48. Wang, C.Y.; Yeh, I.H.; Liao, H.Y.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616.
49. Azad, R.; Niggemeier, L.; Hüttemann, M.; Kazerouni, A.; Aghdam, E.K.; Velichko, Y.; Bagci, U.; Merhof, D. Beyond Self-Attention: Deformable Large Kernel Attention for Medical Image Segmentation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 3–8 January 2024; pp. 1287–1297.
50. Sun, G.; Pan, Z.; Zhang, A.; Jia, X.; Ren, J.; Fu, H.; Yan, K. Large Kernel Spectral and Spatial Attention Networks for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–15. [\[CrossRef\]](#)
51. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 764–773.
52. Qi, Y.; He, Y.; Qi, X.; Zhang, Y.; Yang, G. Dynamic Snake Convolution Based on Topological Geometric Constraints for Tubular Structure Segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 2–6 October 2023; pp. 6070–6079.
53. Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A.C.; Lo, W.Y.; et al. Segment Anything. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 2–6 October 2023; pp. 4015–4026.
54. Ding, X.; Zhang, X.; Han, J.; Ding, G. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 11963–11975.

**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.