## Front Matter

### Title

Long title: SDC-DeepLabv3+: Lightweight and precise localization algorithm for safflower-harvesting robots
Short title: Algorithm for safflower-harvesting robot

### Authors

Zhenyu Xing[1, 3], Zhenguo Zhang[1, 2, *], Yunze Wang[1], Peng Xu[1], Quanfeng Guo[1], Chao Zeng[1], Ruimeng Shi[1]

### Affiliations

[1] College of Mechanical and Electrical Engineering, Xinjiang Agricultural University, Urumqi 830052, China

[2] Key Laboratory of Xinjiang Intelligent Agricultural Equipment, Urumqi 830052, China
[3] Key Laboratory of Intelligent Equipment and Robotics for Agriculture of Zhejiang Province, Hangzhou 310058, China
[*]Address correspondence to: zhangzhenguo@xjau.edu.cn

### Abstract

Harvesting robots can easily damage small safflower filaments and cannot accurately harvest filaments. In addition, harvesting robots experience difficulty in localizing accurately under the influence of a near-color background and blurred contour edge features in complex environments. Therefore, a method for detecting and locating filament picking points based on an improved DeepLabv3+ algorithm is proposed in this study. A lightweight network structure, ShuffletNetV2, was used to replace the backbone network Xception of the traditional DeepLabv3+. Convolutional branches for three different sampling rates were added to extract information on the safflower features under the receptive field and reduce the parameters. Convolutional block attention was incorporated into feature extraction at the coding and decoding layers to solve the interference problem of the near-color background in the feature-fusion process. Then, using the region of interest of the safflower branch obtained by the improved DeepLabv3+, an algorithm for filament picking-point localization was designed based on barycenter projection. The tests demonstrated that this method was capable of accurately localizing the filament. The mean pixel accuracy and mean intersection over union of the improved DeepLabv3+ were 95.84% and 96.87%, respectively. The detection rate and parameters required were superior to those of other algorithms. In the localization test, the depth-measurement distance between the depth camera and target safflower filament was 450–510 mm, which minimized the visual-localization error. The average localization and picking success rates were 92.50% and 90.83%, respectively. The results show that the proposed localization method offers a viable approach for accurate harvesting localization.

## MAIN TEXT

### 1. Introduction

Safflower is a specialty economic crop with a combination of medicinal herbs, dyes, oil, and fodder and is thus a key pillar of our society [1]. The fruit balls of safflowers contain multiple clusters of small, compact, and dense safflower filaments. The entire life cycle is harvested continuously for 3–5 crops [2, 3]. Filaments that are not immediately harvested from the fruiting bulb affect subsequent opening, resulting in reduced filament production [4, 5]. Current safflower-crop production relies on extensive labor, negatively impacting production and the lives of flower farmers. Agricultural robots have the potential to revolutionize standard practices [6]. Segmenting safflower filaments can effectively reduce field operations and improve filament yield through robotic identification [7, 8]. However, the near-color background and contour edge features of the filament harvesting points are blurred owing to environmental disturbances (weather and light), making their location difficult [9, 10]. Therefore, a safflower-filament localization method based on a segmentation algorithm is necessary to accurately localize and harvest filaments with high efficiency and minimal damage in complex environments.

Recently, deep-learning-based flower-classification methods have been applied to various types of flower-detection problems [11-13]. Most used convolutional neural networks containing multilayer stacked structures for flower color, shape, and appearance features to obtain better detection results. Dias et al. [14] proposed an improved end-to-end residual convolutional neural network to enhance color sensitivity. The flowers were accurately identified by applying DEEPLAB+RGR with an effective recall and accuracy higher than 90%. Tian et al. [15] proposed a method for introducing single-shot multiBox detector deep-learning techniques to color detection and recognition. The average accuracy on the flower dataset was 83.64%. Williams et al. [16] used Faster R-CNN for training on a kiwifruit-flower dataset. The average accuracy of the algorithm was 85.3%. Palacios et al. [17] proposed a field Mikania-recognition model based on deep convolutional neural networks (DCNN). The accuracy and computing time of the algorithm were improved by fusing the local response normalization function of AlexNet with the continuous convolutional structure of VGG-Net. The recognition accuracy was 94.50%. Zhou et al. [18] optimized the model structure of VGGNet and preferred an eight-layer network algorithm for tomato-flower shape feature extraction during different flower periods. The detection accuracy was 84.48%. Xiong et al. [19] recognized and segmented the shape features of litchi flowers that were densely aggregated in complex natural environments by constructing a 34-layer ResNet backbone network. The mean average precision was 87%. Zhao et al. [20] proposed a cascade convolutional neural network-based tomato-bouquet recognition method. To realize the extraction of the bouquet region of tomatoes and accurate recognition of the flowering period in the bouquet, the average detection accuracy reached 82.79%. Although the aforementioned methods address the detection problem in terms of flower features, they are only applicable to cases with obvious differences (color and brightness) between the target and background. Moreover, most studies have only focused on flower detection [21, 22]. Furthermore, few studies have developed simultaneous extraction and picking localization.

Several studies have shown that the most important feature of DeepLabV3+ is the introduction of null convolution compared with the latest improved PSPNet [23], SegNet [24], U-Net [25], MDE-UNet [26], CDMS [27], and CondInst [28] algorithms. The sensory field is increased without loss of information. Each convolution output contains a large range of information, which is beneficial for extracting multiscale information [29, 30]. DeepLabV3+ achieved superior segmentation results for various targets. Zhu et al. [31]

proposed a two-stage model, LD-DeepLabv3+, with adaptive loss, to address the problems of hard samples and pixel-proportion imbalance in diseased apple-leaf image segmentation. Peng et al. [32] focused on the semantic segmentation of branches to enable robotic litchi-fruit harvesting by clamping/shearing bearing branches. Therefore, to improve the recognition and localization accuracy of safflower-harvesting robots, this study uses the DeepLabV3+ segmentation algorithm, which has a high accuracy and lightweight structure. A method for detecting and localizing filament picking points based on an improved DeepLabv3+ algorithm is proposed. The improved DeepLabv3+ adopts the lightweight network ShuffletNetV2 to replace the backbone network, improves atrous spatial pyramid pooling (ASPP), and incorporates the convolutional block attention module (CBAM), reducing background interference and enhancing target features. An improved DeepLabv3+ algorithm was used to segment the filament, fruit ball, and branches. The line features of the branch and minimum distance constraint from the barycenter of the filament and fruit ball to the line of the branch were searched and solved to locate the picking-point position. The main contributions of the proposed approach are as follows:

(1) DeepLabV3+ image segmentation with high accuracy and a lightweight structure is used as the basis for the algorithm to reduce the interference of background regions and contour edges on the filament, fruit bulb, and backbone segmentation. A lightweight network, ShuffletNetV2, is used as the backbone network. The input safflower features are operated during channel separation and a channel-blending operation was performed to ensure that the branch safflower-feature information is fully fused. Subsequently, the downsampling unit directly increases the number of network channels and network width to enhance the capability of the network to extract safflower features.

(2) According to the geometrical characteristics of the filaments, the dilated depth-separable convolution replaces the original ASPP module convolution. Three branches of convolution at different sampling rates are added to extract information on the safflower features under the receptive field and reduce the parameters. The detection effect significantly improves after the introduction of the Convolutional Block Attention Module (CBAM). The problems of missed and incorrect segmentation are effectively solved using light intensity and crop shading.

(3) Combined with obtaining the regions of interest (ROIs) of safflower branches, the line segments are solved using the Hough straight-line detection algorithm. Concurrently, a picking-point localization algorithm is designed to locate filaments based on barycenter projection with respect to safflower features. Recognition accuracy and real-time performance are effectively improved. It is proposed to provide technical support for safflower-harvesting robots to locate and harvest safflowers with high efficiency and minimal damage.

The remainder of this paper is organized as follows. Section 2 explains the relevant materials and methods. Section 3 describes the localization test of the filament picking point in this study. Section 4 presents the experimental results, Section 5 presents a discussion, and Section 6 presents the conclusions and lists future work.

## 2. Materials and Methods

### *2.1. Image data acquisition*

The safflower-image data used in this study were collected in late July 2023 from a safflower planting base in Yili, Xinjiang, China. Safflower grows naturally upward with filaments opening on the fruiting ball. The area connecting the filament and fruit ball is the necking (green boxed area in Fig. 1), which is the best harvesting site for the filament picking point. A safflower branch is typically located directly below the filament, fruit ball, and necking, as shown in Fig. 1.

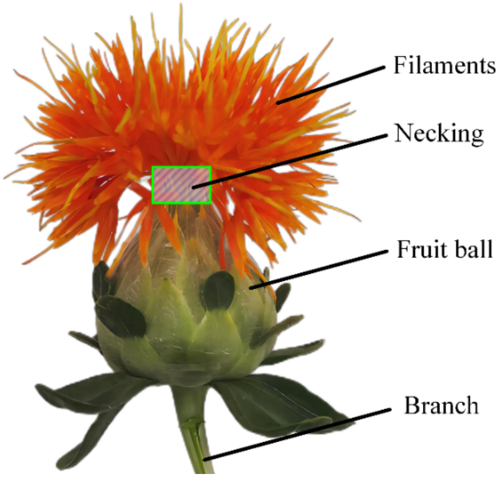

**Fig. 1.** Structure diagram of the opening safflower.

A D435i depth camera (Intel Corp., USA) was used as the image-acquisition equipment. The image size was 640 × 480 pixels and the format of the acquired images was .png. The acquired images included different weather, light, and shade conditions. A total of 525 safflower-filament images were obtained, as shown in Fig. 2.

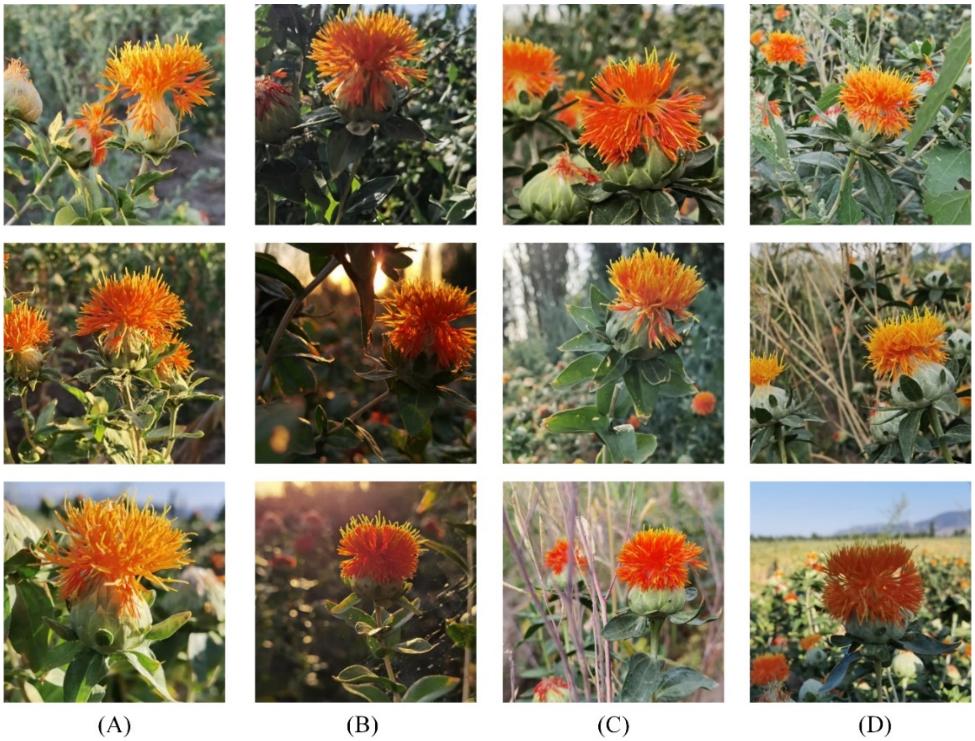

**Fig. 2.** Examples of safflower images. (A) Sunny day with light, (B) sunny day with backlight, (C) overcast day with light, and (D) cloudy day with light.

### 2.2 Dataset production

The data were enhanced using geometric (translation and rotation) and color transformations (contrast and brightness) to expand the dataset to 1050 pictures. The data was divided into training and test sets in a ratio of 8:2, including 840 pictures for the training set and 210 pictures for the validation set. The training set was used to train the deep network model-parameter weights, and the test set was used to evaluate the generalization ability of the final model [33, 34]. The filament, fruit ball, and branch were labeled separately using the LabelIme tool. The category information is filament, fruit ball, and branch, which is stored in the .json format.

### 2.3 DeepLabv3+ algorithm

DeeplabV3+ is a semantic segmentation algorithm based on a DCNN that mainly consists of an encoder [35], decoder [36], DCNN [37], and ASPP [38]. First, the image input to the encoder is fed into the backbone feature-extraction network, Xception. Second, the parallel ASPP module is passed in to incorporate the pooled features at the image level to merge and extract multiscale contextual information. Then, the number of channels is reduced by a 1×1 convolution and output to the decoder for four-fold upsampling. Fusion is performed using shallow features extracted from the DCNN module. Finally, the fused features are passed through a 3×3 convolution and four-fold upsampling to obtain the predicted image, which has the same size as the original image [39].

The high-accuracy DeepLabv3+ algorithm obtains rich image-boundary information and multiscale features. It is suitable for solving problems with no obvious differences and blurred contour-edge features between the target to be segmented and background of small-volume safflower filaments in complex environments [32, 40]. However, its network parameters are large, and its prediction speed is slow. It also produces missed and incorrect segmentation problems, resulting in a large deviation in the localization accuracy. Therefore, the detection algorithm must be further optimized.

### 2.4 Improved DeepLabv3+

To reduce the number of algorithm parameters, further capture the global and contextual information, and avoid the localization accuracy deviation caused by missed segmentation and mis-segmentation, a method for safflower filament picking-point localization with improved DeepLabv3+ is proposed (SDC-DeepLabv3+), which fuses the modules of ShuffletNetV2 [32], the ASPP of dilated depth-separable convolution (DDSC-ASPP), and CBAM [41], as shown in Fig. 3. To distinguish DeepLabv3+, the SDC-DeepLabv3+ algorithm is reordered with the initials of ShuffletNetV2, DSC-ASPP, and CBAM as content combinations. Lightweight ShuffleNetV2 is used as the backbone network, and the ASPP is improved to DDSC-ASPP. The number of algorithmic parameters is reduced, extracting more safflower features from different receptive fields. Moreover, CBAM is added so that the fusion of deep and shallow safflower features is allocated to the channel and spatial-attention resources. The feature information is maximized. SDC-DeepLabv3+ not only reduces the computing cost, but also considers the segmentation accuracy. This improves the capability of the algorithm to predict the localization accuracy of the safflower-filament picking points.

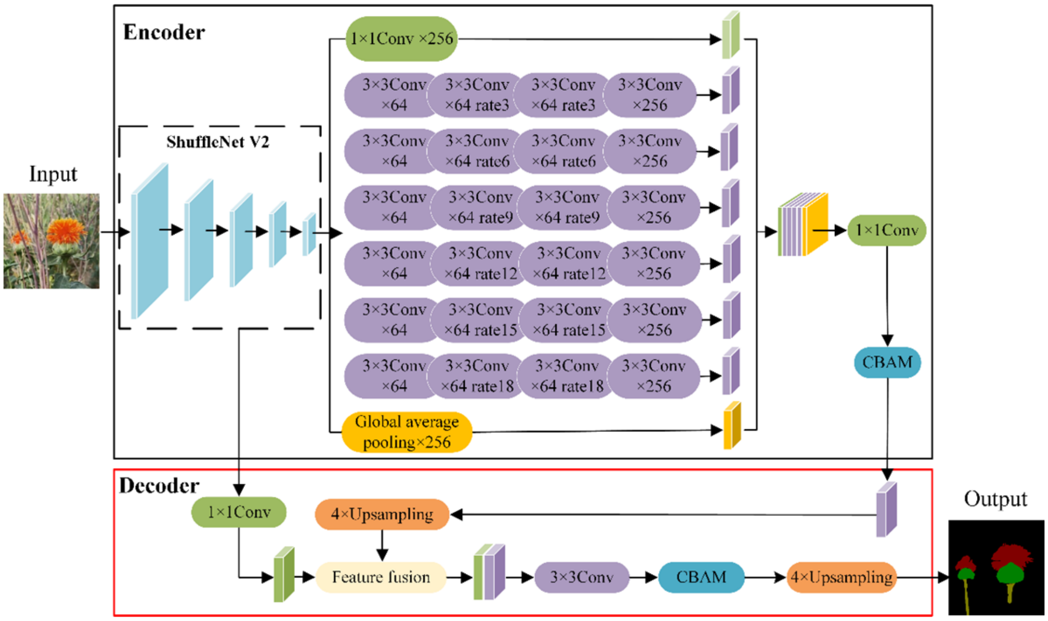

**Fig. 3.** Schematic of the SDC-DeepLabv3+ algorithm structure.

### *2.4.1 ShuffletNetV2 module*

To reduce algorithm computation and improve detection accuracy, the ShuffleNetV2 [31] structure is selected as the backbone network, as shown in Fig. 4. ShuffleNetV2 is based on ShuffleNetV1 [42] and retains the components of ShuffleNetV1, such as the channel shuffle and depth-separable convolution. A more efficient basic block is proposed to reduce the number of algorithm parameters and computations and improve detection accuracy.

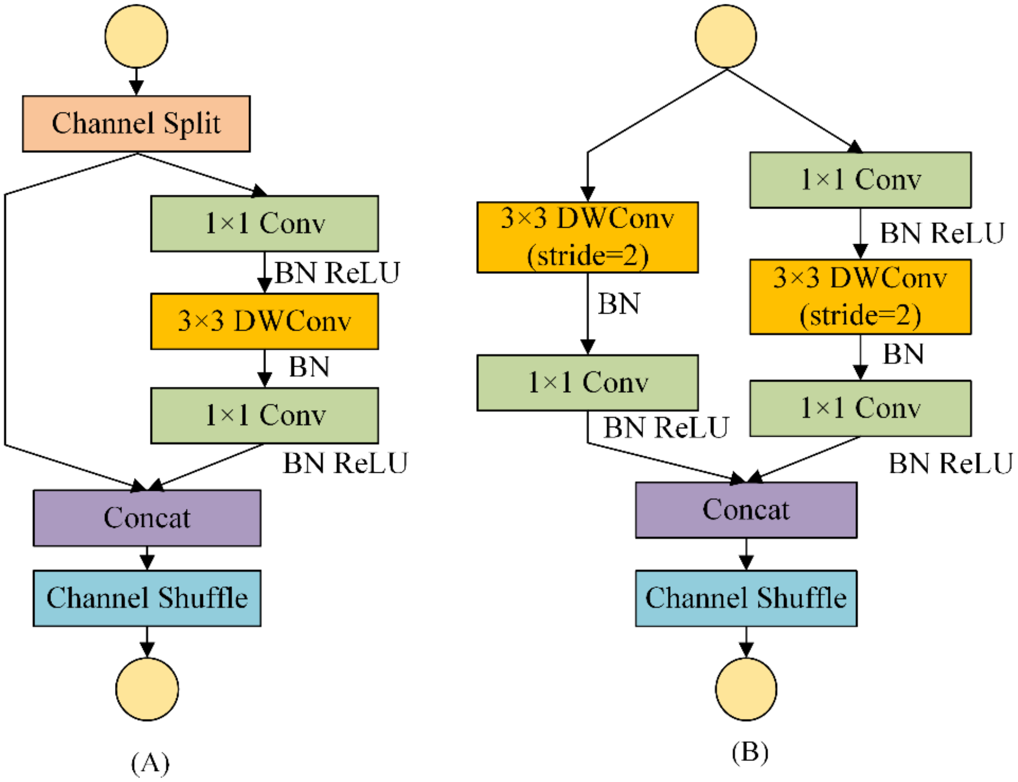

**Fig. 4.** Structure of ShuffleNetV2. (A) Stride=1 block and (B) stride=2 blocks.

The input safflower image was first subjected to 24 standard convolutions of size 3×3 and step 2 for feature extraction to obtain a new feature map. MaxPool was then used to maximize pooling for downsampling purposes to obtain a 24×56×56 feature map. Finally, three stage structures consisting of ShuffleNetV2 basic components with stride=1 and 2 blocks were connected [43]. The semantic information of the different layers of the target image was extracted.

Each unit feature channel input of the component stride=1 block was divided into two branches, one of which was changed, and the other comprised three convolutions. The two 1×1 convolutions were no longer group convolutions but ordinary 1×1 convolution operations. After convolution, the two branches were connected and fused. The number of channels was maintained. A channel shuffle operation was performed to enable information communication between the two branches, where the connected fusion of the channel shuffle and channel split of the next modular unit were synthesized into an element-level operation.

For spatial downsampling, the stride=2 block removed the channel-split operator and channel split, and each branch copied the input directly. Each branch had stride=2 downsampling for concatenated fusion [30]. The filament feature-map space size was halved, but the number of channels doubled. Setting the number of channels in each block, such as 0.5× and 1×, can adjust the complexity of the model. The specific structure is shown in Fig. 4. The architecture was designed with high efficiency to realize information communication between different channel groups and improve the accuracy of filament feature extraction.

### *2.4.2 DDSC-ASPP*

Compared with other detection objects [23-25], the shapes and distributions of safflower filaments, fruit balls, and branches are unstable, varied, and susceptible to occlusion. The DeepLabv3+ algorithm uses an ASPP module with a combination of dilation rates of 6, 12, and 18 to extract multiscale features [44, 45]. However, its large dilation rate leads to a large receptive field, insufficiently detailed feature extraction, and poor extraction of small targets. It is difficult to capture all types of information in a single-scale receptive field [46]. Therefore, combined with the edge complexity of the safflower features and the high leakage rate of small targets, this study added three dilated depth-separable convolution [47] branches based on the ASPP module. The filament, fruit ball, and branch feature information are extracted under the receptive field using depth-separable convolution with higher sampling rates. Reducing the number of algorithmic parameters can satisfy the real-time algorithmic detection.

(1) Depth-separable convolution

By retaining the original feature extracted from the receptive field, the depth-separable convolution actively reduces the parameters and algorithm training [29]. The convolution process is divided into two parts: channel-by-channel and point-by-point convolution. The convolution kernel of the channel-by-channel convolution corresponds one-to-one to the channel of the input image. After the channel-by-channel convolution, the number of channels for the feature map remained the same as that before the input. Because the same locations of different channels had connections, a 1×1 point-by-point convolution was used to adjust the number of channels and establish connections between them. A schematic of the depth-separable convolution is shown in Fig. 5.

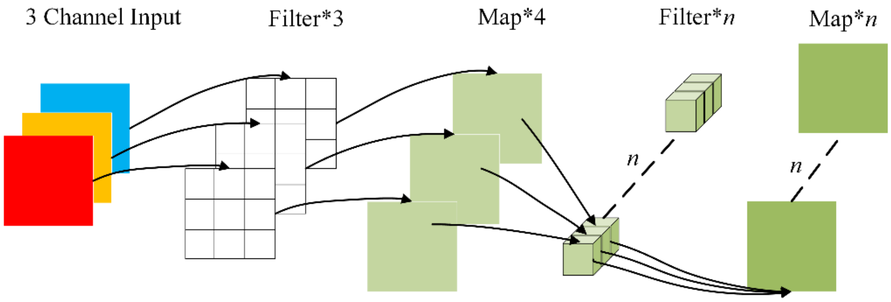

**Fig. 5.** Comparison of depth-separable convolution processes.

(2) Increasing separable convolution of the dilated depth at different sampling rates

The ASPP structure comprises a dilated convolution layer with different sampling rates, a 1×1 convolution layer, and an image-pooling layer [6]. However, because of the large sampling-rate interval of the dilated convolution, the feature extraction of the ASPP structure was more sparse, resulting in the loss of some local information from the safflower image [48, 49]. Therefore, three dilate depth-separable convolution branches were added, based on the original sampling rate. The combination of the dilate rate was modified to 3, 6, 9, 12, 15, and 18. The middle six convolutional branches were subjected to two 3×3 convolutional cascade operations. A dilated depth-separable convolution that realizes multiscale feature extraction can be added to the receptive field.

When the dilation rate of the dilated depth-separable convolution is $r$, and the size of the convolution kernel is $k$, the size of the receptive field is as follows:

$$R=(r-1)(k-1)+k \tag{1}$$

However, when the two-layer dilated depth-separable concentration series is cascaded, the receptive field size is as follows:

$$R=R_1+R_2-1 \tag{2}$$

where $R_1$ and $R_2$ denote the receptive fields provided by the two-layer dilated depth-separable convolution.

DDSC-ASPP first reduced the middle four convolutional branches to 64 channels using a 1 × 1 convolution. Two 3×3 convolutions were then performed. Subsequently, using 1×1 convolution dimensions for 256 channels, the bottleneck structure effectively reduced the parameters. The DDSC-ASPP decreased the loss rate of edge information and small-target safflower information in the safflower image. This can facilitate detailed feature extraction of small targets and safflower contour edges. Thus, the problem of missing the segmentation of the filament, fruit ball, and branch was effectively minimized.

### *2.4.3 CBAM*

To accurately extract edge features and solve the interference problem of background information for filaments, fruit balls, and branches, this study proposes injecting the CBAM into the feature-extraction module of the encoding and decoding layers [50, 51].

Unlike Squeeze-and-Excitation [52] and Efficient Channel Attention [53], which consider channels and ignore spatial information, CBAM is a lightweight module that combines channel and spatial attention mechanisms [54]. Its core aim is to provide segmentation attention to the desired region, as shown in Fig. 6. Given a safflower-feature map, the CBAM was able to sequentially generate attention feature-map information in both channel and spatial dimensions. The channel-attention mechanism was operated first, followed by the spatial-attention mechanism. Then, the information of the two feature maps was multiplied by the original input feature map for adaptive feature correction to produce the final feature map. The CBAM was embedded into the backbone network to improve performance and achieve an enhanced ROI in both the channel and spatial dimensions.

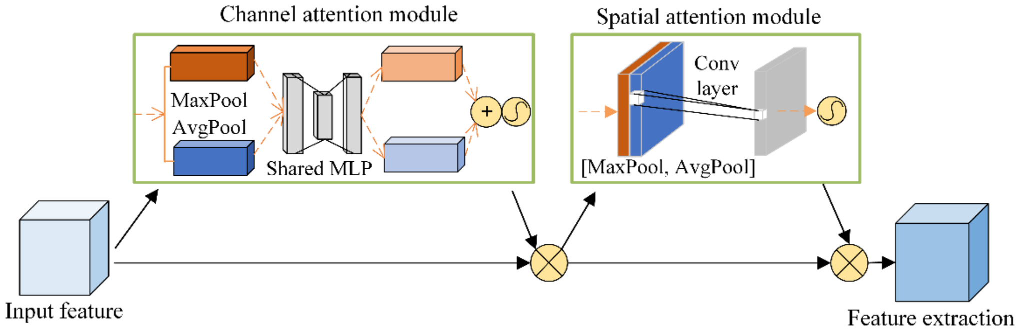

**Fig. 6.** CBAM structure.

The CBAM captured the correlation between features in different dimensions by adapting learning channels and spatial-attention weights. The enhanced features were used as inputs for the subsequent network layers. It suppressed safflower-image noise and irrelevant information while preserving key information. Therefore, the performance of safflower recognition and detection was enhanced, realizing the accurate segmentation of filaments, fruit balls, and branches.

### *2.5. Localization method for filament picking point*

To improve the localization accuracy and speed of damage-free safflower-filament picking and remove the influence of image noise on the localization of the picking point, combined with the characteristics of safflower-growth diversity, this study set up an ROI in the branch region [55]. The flowchart of the filament picking point localization method is shown in Fig. 7.

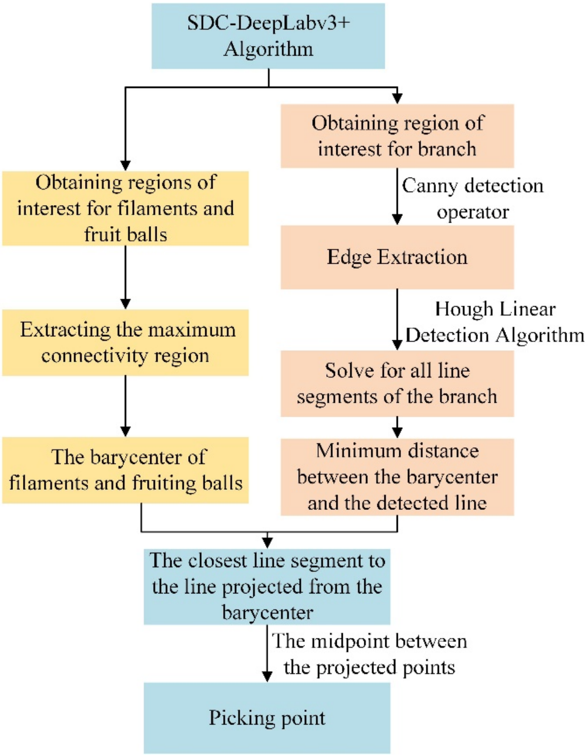
SDC-DeepLabv3+ Algorithm
Obtaining region of interest for branch
Canny detection operator
Edge Extraction
Hough Linear Detection Algorithm
Solve for all line segments of the branch
Minimum distance between the barycenter and the detected line
Obtaining regions of interest for filaments and fruit balls
Extracting the maximum connectivity region
The barycenter of filaments and fruiting balls
The closest line segment to the line projected from the barycenter
The midpoint between the projected points
Picking point

**Fig. 7.** The flowchart of the filament picking point localization method.

Firstly, the safflower image was preprocessed according to the SDC-DeepLabv3+ algorithm and segmented into filaments, fruit bulbs, branches and trunks. The interest regions of filaments and fruit bulbs were obtained. The maximum connectivity region of the two was extracted, and the center of mass coordinates of the filament and fruit ball were solved. Meanwhile, edge detection was performed on the ROI based on the branch ROI obtained by the SDC-DeepLabv3+ algorithm to obtain an edge binary map. All the line segments in the binary map were solved using the Hough line-detection algorithm [56]. The line segment that solved for the minimum distance between the barycenter coordinates of the filament and fruit ball and the line was considered as the line position of the picking point and was projected onto this line segment. Finally, the midpoint of the two barycenters for the projection points on the line section was used as the picking point. A schematic of the picking-point calculation model is shown in Fig. 8.

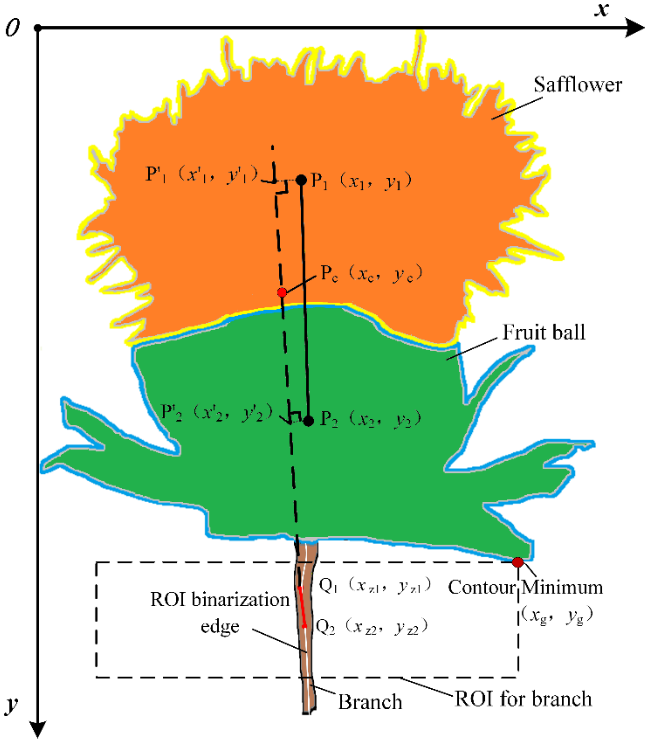

**Fig. 8.** Schematic of picking-point calculation model. The barycenter coordinates of the filament and fruit ball are $P_1$ $(x_1, y_1)$ and $P_2$ $(x_2, y_2)$, respectively. The barycenter projection of the filament and fruit ball are $P'_1$ $(x'_1, y'_1)$ and $P'_1$ $(x'_1, y'_1)$, respectively. The values of the two endpoint coordinates are $Q_1$ $(x_{z1}, y_{z1})$ and $Q_2$ $(x_{z2}, y_{z2})$, respectively. The value of the picking point coordinates is $P_c$ $(x_c, y_c)$.

### *2.5.1. ROI Extraction with Linear Detection*

Safflower grows naturally upward. In the absence of interference from other supports, safflower filaments open on the fruit ball, and the branch is typically located directly below the filament and fruit ball. To minimize environmental interference with the localization of the picking point, the ROI of the branch was determined based on the barycenter of the filament, barycenter of the fruit ball, lowest point of the contour, and distance between the left and right poles, as shown in Fig. 8.

In the ROI images, the edge response of the branches was stronger. To simultaneously suppress noise and enhance the boundary, a safflower image was subjected to a binarization operation [57]. The barycenter coordinates of the filament $P_1$ ($x_1$, $y_1$) and fruit ball $P_2$ ($x_2$, $y_2$) were derived. The edge gradient-detection algorithm was used for edge extraction of the branch ROI. The gradient-direction vector was calculated using a finite difference. The gradient of the pixel points in the ROI image was calculated as follows:

$$\begin{cases} \left\|\nabla f(x,y)\right\| = \left\{ f_w(x,y)^2 + f_h(x,y)^2 \right\}^{1/2} \\ f_w(x,y) = \frac{1}{2}\left( f(x+1,y) - f(x-1,y) \right) \\ f_h(x,y) = \frac{1}{2}\left( f(x,y+1) - f(x,y-1) \right) \end{cases} \tag{3}$$

where $f(x, y)$ denotes the pixel-point value in the ROI image, and $f_w(x, y)$ and $f_h(x, y)$ denote the gradients in the direction of the horizontal and vertical coordinate axes, respectively.

The local maximum points of the gradient are considered as the corresponding edge pixel points in the safflower image. Subsequently, the generated gradient map was thresholded for image segmentation to obtain a binary edge map of the ROI of the branch.

### *2.5.2. Picking-point localization based on barycenter projection*

Because the edges of the branches were presented as lines, the Hough line-detection algorithm was used to detect the lines in the binary edge map within the branch ROI. All the line segments in the region that satisfied these conditions were detected. The values of the two endpoint coordinates $Q_1$ ($x_{z1}$, $y_{z1}$) and $Q_2$ ($x_{z2}$, $y_{z2}$) were recorded for all detected line sections, as shown in Fig. 8. Based on the coordinates of the two endpoints, the equation for the line is as follows:

$$\frac{x - x_{z1}}{x_{z2} - x_{z1}} = \frac{y - y_{z1}}{y_{z1} - y_{z2}} \tag{4}$$

where ($x$, $y$) are the pixel coordinates, and ($x_{z1}$, $y_{z1}$) and ($x_{z2}$, $y_{z2}$) are the coordinates of the first and second endpoints of the branch line, respectively.

The distance from the barycenter of the filament and fruit ball to each line $L_b$ was calculated based on the coordinates of the endpoints for all satisfied lines. The minimum constraint function for the distance between the barycenter and line min($L_b$) was solved to determine the line with the minimum distance from the picking point. The line distance $L_b$ to the center of mass is calculated as follows:

$$L_b = \frac{\left| \frac{y_{z1} - y_{z2}}{x_{z1} - x_{z2}} x_b + \frac{x_{z1} y_{z2} - y_{z1} x_{z2}}{x_{z1} - x_{z2}} - y_b \right|}{\sqrt{1 + \left( \frac{y_{z1} - y_{z2}}{x_{z1} - x_{z2}} \right)^2}} \tag{5}$$

Because safflower naturally grows upward and opens flowers on the fruit ball, the branch is typically located directly below the barycenter. Theoretically, the extension line through the branch must pass through the barycenter. However, under the influence of image-segmentation errors, blurred and irregular contours of the filament and fruit ball, and other disturbing factors in practice, the calculated extension of the branch line tends to be offset from the barycenter to a certain extent. The barycenter projection of the filament and fruit ball, $P'_a$ ($x'_a$, $y'_a$) and $P'_b$ ($x'_b$, $y'_b$), is obtained by calculating the barycenter-projection point on the line, where the picking point is located at the minimum distance using Equation (5).

The midpoint between the two barycenter projection points on the line was considered as the picking point $P_c$ $(x_c, y_c)$.

## 3. Localization test of filament picking point

### *3.1. Test platform*

The training platform used in this study was a Windows 11 (64-bit) operating system (Lenovo R9000X, Razor 8-core R7-5800H, RTX3060). The computer was equipped with an Intel(R) Xeon (R)CPUE52630v4@2.20GHz operating memory, 64 GB running memory, 1 TB SSD, and 12 GB GTX1080Ti×2 GPU. Anaconda 3.5.0 (Anaconda Inc., USA), CUDA 11.0 (Nvidia, USA), and cuDNN 6.0 (Nvidia, USA) libraries were used. In addition, the open-source deep learning framework Pytorch was used as the development environment, and the programming language used was Python 3.9 (Python Software Foundation, USA).

### *3.2. Evaluation indicators*

In this study, mean pixel accuracy (mPA), mean intersection over union (mIoU), frames per second (FPS), and params were used as algorithm performance-evaluation metrics. mPA calculates the number of pixels that are correctly classified in each pixel category within a class and then determines the average of all classes, %. The mIoU is a standard metric for semantic segmentation. It calculates the ratio of the intersection and concatenation of the set of true and predicted values [58]. The evaluation metrics are calculated as follows:

$$\mathrm{mPA} = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k} p_{ij}} \tag{6}$$

$$\mathrm{mIOU} = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{i=0}^{k} p_{ij}+\sum_{i=0}^{k} p_{ji}-p_{ii}} \tag{7}$$

$$MP = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k} p_{ii}+\sum_{j=0}^{k} p_{ij}} \tag{8}$$

$$MR = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k} p_{ji}-p_{ii}} \tag{9}$$

where $p_{ii}$ denotes the number of true instances, while $p_{ij}$ and $p_{ji}$ are interpreted as false-negative and false-positive instances, respectively. False-negative instances are samples predicted to be non-filament, fruit ball, and branch regions and actual filament, fruit ball, and branch regions. False-positive instances are samples predicted to be filament, fruit ball, and branch regions and actual non-filament, fruit ball, and branch regions; $k$ denotes category.

### *3.3. Localization performance test*

A test set of images was used to validate the performance of the SDC-DeepLabv3+-based picking-point localization method. A target safflower was selected from each test image by manual labeling, and the coordinate position of the best picking point was recorded. In the test, SDC-DeepLabv3+ was used to process the safflower images of the test, and the two-dimensional coordinate values of the predicted target picking points were recorded. These errors were used as an important basis for judging the performance of the picking-point localization method [55]. The prediction accuracy of the picking-point localization method was evaluated using pixel error. The formula for the pixel error is as follows:

$$\begin{cases} e=\sqrt{\left(e_x{}^2+e_y{}^2\right)} \\ e_x=\min\left|X_{\mathrm{bt}}-x\right| \\ e_y=\min\left|Y_{\mathrm{bt}}-y\right| \end{cases} \tag{10}$$

where $X_{\mathrm{bt}}$ and $Y_{\mathrm{bt}}$ denote the horizontal and vertical coordinate values of the manually labeled best picking point in the image, respectively; $x$ and $y$ denote the horizontal and vertical coordinate values of the picking point predicted by the picking-point localization method in the image, respectively; $e_{\mathrm{x}}$ and $e_{\mathrm{y}}$ denote the pixel-point error value between the predicted and manually labeled picking points in the $x/y$ direction; and $e$ denotes the combined pixel-point error value of the predicted and manually labeled picking points.

The two-dimensional coordinate values of the filament picking point were mapped and matched with the three-dimensional depth map acquired by a camera (RealSense D435i, Intel, USA). Using the inherent internal parameters of the depth camera, the image coordinate system was converted to the camera's optical coordinate system to obtain the three-dimensional spatial coordinate values of any coordinate point in the camera coordinate system. Subsequently, the global coordinate system was set on a calibration plate. The first square was used as the reference point in the upper-left corner of the calibration plate and set as the origin of the global coordinate system. The normal direction of the calibration-plate plane was used as the $Z$-axis depth direction, and the horizontal direction was used as the $X$-axis direction. The three-dimensional coordinates of the filament picking point were obtained by interconverting the camera coordinate system with the global coordinate system of the harvesting robots in the global coordinate system. By comparing the results from the manual measurement and depth camera, the visual localization error $E_{\mathrm{Z}}$ in the $Z$ direction and localization error $E_{\mathrm{X}}$ in the $X$ direction were calculated as follows:

$$E_Z=\frac{\sum_{m=1}^{M}\left|Z\left(CP\right)_m-Z\left(GT\right)_m\right|}{M} \tag{11}$$

$$E_X=\frac{\sum_{m=1}^{M}\left|X\left(CP\right)_m-X\left(OP\right)_m-L_m\right|}{M} \tag{12}$$

where $M$ and $m$ denote the total number of image combinations and image sequence number, which are $M$=50, $m$ [1, 50], respectively; $Z(CP)_{\mathrm{m}}$ and $X(CP)_{\mathrm{m}}$ denote the values of the $Z$ and $X$ coordinates of the picking point measured by the depth camera, respectively; $Z(GT)_{\mathrm{m}}$ denotes the value of the $Z$ coordinates of the manually measured picking point; $X(OP)_{\mathrm{m}}$ denotes the value of the $X$ coordinates of the calibration-plate reference point (the origin of the global coordinate system); and $L_m$ denotes the manually measured distance between the calibration-plate reference point and filament picking point.

## 4. Results and Analyses

### *4.1. Algorithm training results*

To improve algorithm performance and decrease overfitting, the initial learning rate of the SDC-DeepLabv3+ algorithm was 0.01, the batch size was set to eight, and the number of training iterations was 1000. If the accuracy was not increased within 15 rounds of training, the learning rate would be reduced to 0.01 times the original one using the SGD optimizer [41]. The network-weight file was saved after every 10 training epochs. The parameters with the highest and lowest accuracies were saved. The algorithm was validated using a test set, and the detection results were output. The training results for the SDC-DeepLabv3+ algorithm are shown in Fig. 9.

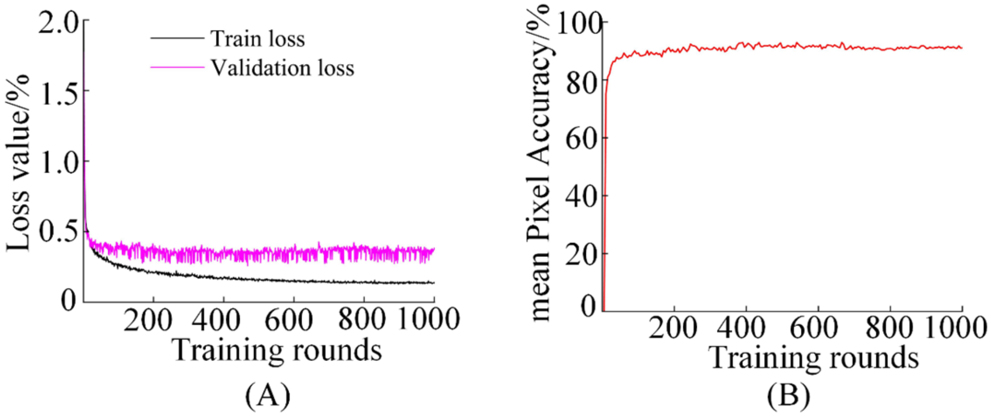

**Fig. 9.** Parameters of the training process. (A) Loss curve and (B) mean pixel accuracy.

The loss value decreased the fastest in the first 163 rounds of the training process. It was still decreasing after 163 rounds, and the rate gradually became slower. When the number of iterative training rounds reached 902, the loss curve tended to flatten out. It eventually stabilized at the minimum value and completed the training, as shown in Fig. 9A. Meanwhile, the mPA curve gradually rose with the increase of training rounds and finally stabilized. As shown in Fig. 9B, the mPA (IoU=0.5) is 92.61%, which indicates that the algorithm converges. The trend of each curve during the training process can reflect the effectiveness of training. It is favorable to improve the accuracy of safflower segmentation.

### *4.2. Evaluation of filament-segmentation algorithm*

#### *4.2.1 Ablation test*

Four sets of ablation tests were conducted to verify the effects of the ShuffletNetV2 backbone network, DDSC-ASPP, and CBAM on the recognition accuracy of the SDC-DeepLabv3+ semantic segmentation algorithm. The test results are listed in Table 1.

**Table 1.** Results of the ablation test.

| Algorithms | Backbone networks | Pyramid module | Add module | mIoU/ % | mPA/ % | FPS/(f·s $^{-1}$) | Params/M B |
|---|---|---|---|---|---|---|---|
| DeepLabv3 + | Xception | ASPP | none | 94.13 | 95.64 | 45.32 | 206.32 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| DeepLabv3+ | ShuffletNetV2 | ASPP | none | 95.01 | 96.33 | 56.19 | 22.75 |
| DeepLabv3+ | ShuffletNetV2 | DDSC-ASPP | none | 95.35 | 96.52 | 58.24 | 14.61 |
| DeepLabv3+ | ShuffletNetV2 | DDSC-ASPP | CBAM | 95.84 | 96.87 | 57.48 | 13.54 |

The backbone network of the original DeepLabv3+ was Xception. The mIoU of this algorithm for safflower detection was 94.13% and the mPA was 95.67%. After changing the backbone network of DeepLabv3+ from Xception to ShuffletNetV2, the mIoU and mPA improved by 0.88 and 0.69 percentage points, respectively. The parameters were significantly reduced to the original 0.26 percentage points. When ASPP was improved to DDSC-ASPP, mIoU and mPA were enhanced by 0.34 and 0.19 percentage points, respectively. The number of samples was reduced by 11.91 MB. This shows that DDSC-ASPP enriched the expression of effective features and enhanced the detection of filaments for small targets. To a certain extent, the algorithm-segmentation accuracy was improved, and the detection effect was enhanced. Finally, with the introduction of the CBAM, the SDC-DeepLabv3+ algorithm enhanced the mIoU and mPA to 95.84% and 96.87% for safflower detection, respectively. This significantly improved the linkage of filament features in the channel and space, suppressed background interference, and highlighted more safflower filaments. In addition, the FPS decreased marginally with the addition of DDSC-ASPP alone. These parameters were further reduced by 0.66 MB.

Comprehensive ablation tests showed that SDC-DeepLabv3+ significantly improved the mIoU, mPA, and FPS. The parameters of the improved network algorithms were significantly reduced. Therefore, the safflower features refined by the SDC-DeepLabv3+ algorithm received attention weights in the channel and spatial dimensions and more prominent target edge features. This provides support for the fast and accurate localization of the picking points.

### *4.2.2. Comparison of different algorithms with backbone networks*

To verify the effectiveness of the feature-extraction network and further analyze the segmentation performance of the improved DeepLabv3+, five segmentation algorithms, including PSPNet [23], SegNet [24], U-Net [25], DeepLabv3+ [32], and SDC-DeepLabv3+, were comparatively tested under the same guarantee of other parameters, as shown in Table 2.

**Table 2.** Comparison of different segmentation algorithms and backbone networks.

| Algorithms | Backbone networks | mIoU/% | mPA/% | FPS/($f\cdot s^{-1}$) | Params/MB |
|---|---|---|---|---|---|
| PSPNet | ResNet50 | 92.86 | 94.99 | 48.37 | 176.53 |
| U-Net | ResNet50 | 93.15 | 95.13 | 39.65 | 92.07 |
| SegNet | ResNet50 | 92.69 | 94.66 | 47.11 | 24.26 |
| DeepLabv3+ | Xception | 94.13 | 95.64 | 45.32 | 206.32 |
| SDC-DeepLabv3+ | ShuffletNetV2 | 95.84 | 96.87 | 57.48 | 13.54 |

(1) Performance comparison of different segmentation algorithms

As shown in Table 2, the PSPNet algorithm failed to completely detect the safflower filaments and branches (Fig. 10B). The mIoU and mPA values for the U-Net algorithm were 93.15% and 95.12%, respectively. The detection results were unclear, and the algorithm could not effectively address the irregular and blurred contours of the branches (Fig. 10C). By comparison, the mIoU and mPA values of the SegNet algorithm were 0.46% and 0.47% lower, respectively. The detection results were consistent; however, sporadic regions were incorrectly detected (Fig. 10D). The DeepLabv3+ algorithm used cavity convolution and a multi-scale strategy to significantly increase the receptive field; however, depressions and burrs remained (Fig. 10E). The accuracy of safflower segmentation was better than that of the previous three algorithms, but the parameters and prediction speed may be improved. After replacing the backbone network of DeepLabv3+ with ShuffletNetV2, SDC-DeepLabv3+ effectively decreased the number of parameters and improved the prediction speed. Relative to DeepLabv3+, the mIoU and mPA improved by 1.71% and 1.23%, respectively, and the FPS improved by 12.16 f/s. This was because the filament was characterized, and the fruit balls and branches differed from the background region. However, SDC-DeepLabv3+ was more capable of detecting small targets with the accurate extraction of edge features. The interference problem of the background information for the filament, fruit ball, and branch feature extraction was further solved, as shown in Fig. 10F. This proved that the improved algorithm achieved excellent results in terms of both segmentation accuracy and prediction speed, making it suitable for deployment in embedded robots. The algorithm was guaranteed to have high segmentation accuracy while being lightweight.

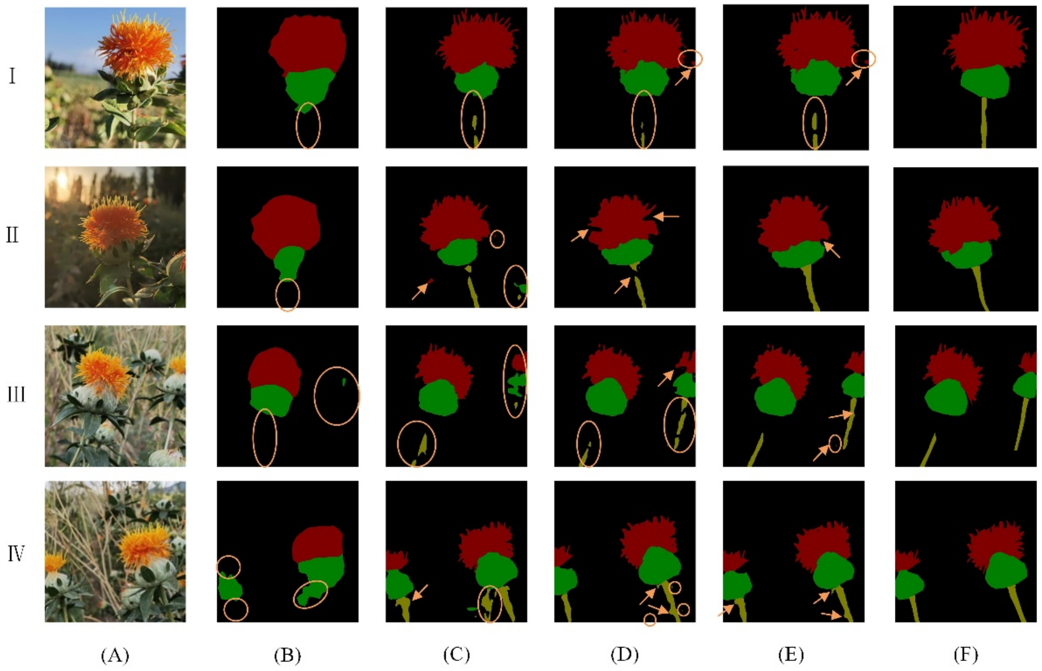

**Fig. 10.** Detection results of different segmentation algorithms. (I) Sunny day with light, ( II) sunny day with backlight, (III) overcast day with light, and (IV) cloudy day with light. (A) Original image, (B) resulting image of PSPNet, (C) resulting image of U-Net, (D) resulting image of SegNet, (E) resulting image of DeepLabv3+, and (F) resulting image of SDC-DeepLabv3+. The arrows indicate areas in which depressions or burrs were detected, and the circles indicate areas in which segmentation errors or omissions were detected.

(2) Performance comparison of different backbone networks

As shown in Table 2, the accuracy of SDC-DeepLabv3+ for safflower segmentation was superior to that of the previous four groups of algorithms. The mIoU and mPA improved, the number of model parameters decreased, and the prediction speed improved. In terms of image-segmentation performance, the mIoU of ShuffletNetV2 was able to achieve 95.84%, which was 2.69 and 1.71 percentage points higher than those of ResNet50 and Xception, respectively. The mPA was 96.87%, which was 1.75 and 1.23 percentage points higher than those of ResNet50 and Xception, respectively. In addition, different backbone networks could efficiently segment the safflower images. ShuffletNetV2 exhibited an improvement of at least 9.11 and 12.16 f/s over ResNet50 and Xception, respectively, and the parameters were reduced by at least 10.71 and 192.78 MB, respectively. The segmentation effectiveness and training efficiency showed that the lightweight ShuffletNetV2 feature-extraction network was effective and could be used as a backbone network for the safflower image-segmentation algorithm.

(3) Performance comparison of four different weather and lighting environments

Comparative results of SDC-DeepLabv3+ algorithm and DeepLabv3+ algorithm for MP, MR, Miou, mPA under four different weather and lighting environments is shown in Table 3.

**Table 3.** Results of algorithm under four different weather and lighting environments.

| Algorithms | Evaluation indicators | Weather and lighting | | | |
|---|---|---|---|---|---|
| | | Sunny day with light | Sunny day with backlight | Overcast day with light | Cloudy day with light |
| DeepLabv3+ | *MP*/% | 93.12 | 92.58 | 90.63 | 91.12 |
| | *MR*/% | 85.47 | 84.84 | 82.91 | 84.05 |
| | mIoU/% | 93.61 | 91.70 | 90.28 | 91.13 |
| | mPA/% | 93.78 | 92.35 | 90.05 | 90.65 |
| SDC-DeepLabv3+ | *MP*/% | 95.16 | 94.08 | 90.95 | 93.84 |
| | *MR*/% | 87.27 | 86.60 | 85.19 | 86.02 |
| | mIoU/% | 95.07 | 94.57 | 91.26 | 93.57 |
| | mPA/% | 95.50 | 94.61 | 91.35 | 92.72 |

As shown in Table 3, the segmentation accuracy of SDC-DeepLabv3+ algorithm was better than the DeepLabv3+ algorithm for different weather and lighting environments. The SDC-DeepLabv3+ algorithm showed the best performance in the evaluation indicators MP, MR, Miou, mPA under sunny day with light, which all outperformed the sunny day with backlight and overcast day with light by at least 0.32%, and improved the performance relative to the DeepLabv3+ algorithm by 2.04%, 1.80%, 1.46%, and 1.72%, respectively. It indicated that the present algorithm was better in sunny conditions than in cloudy and cloudy conditions, which were more affected by light. Especially in the low light environment, the algorithm was difficult to distinguish safflower color and texture features, and mis-segmentation occurs, as shown in Fig. 10.

### *4.3. Localization-performance test*

The localization accuracy of SDC-DeepLabv3+ at the filament picking point was verified under four different weather conditions. The results of the filament picking-point localization are shown in Fig. 11. Four images were selected to test the pixel coordinates and localization errors of the picking points, as shown in Fig. 11 and Table 4.

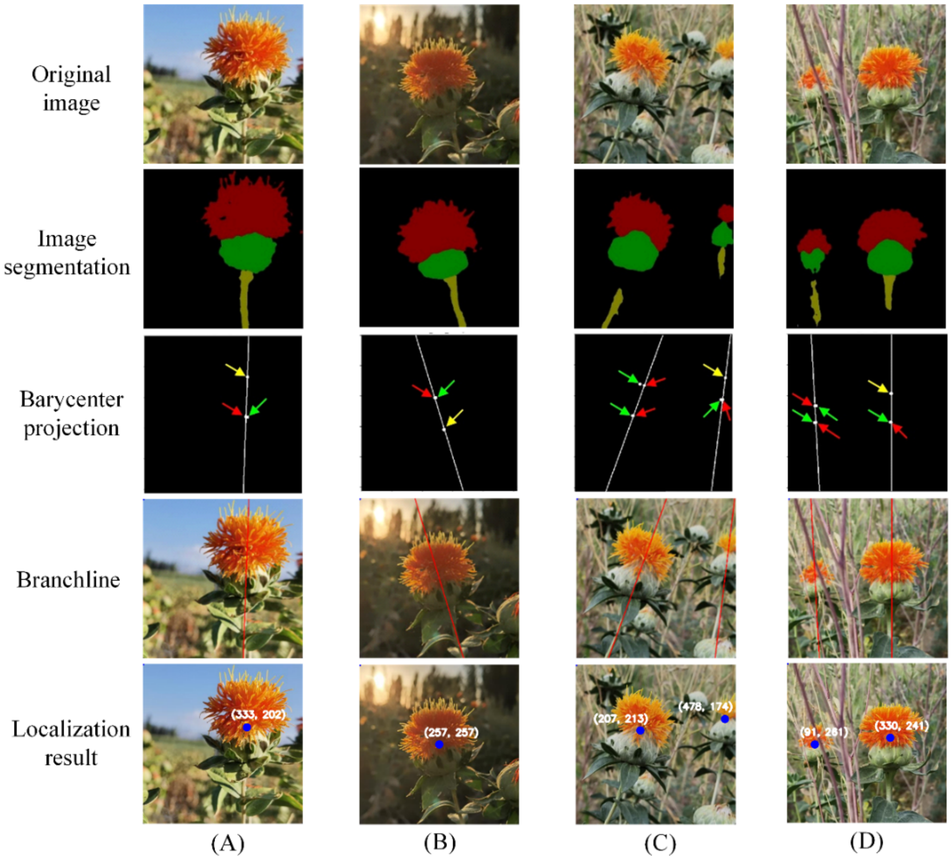

**Fig. 11.** Location process of picking points with four types of lighting. (A) Sunny day with light, (B) sunny day with backlight, (C) overcast day with light, and (D) cloudy day with light. Green arrows indicate the barycenter points of the safflower filaments or fruit balls; red arrows are projection points of the safflower filament or fruit ball barycenter; and yellow arrows are barycenter points coinciding with projection points.

**Table 4.** Picking point and pixel-localization errors.

| Types | The barycenter coordinates of filament/pixel | The barycenter coordinates of the fruit ball/pixel | Picking point/pixel | Pixel localization error/pixel | | |
|---|---|---|---|---|---|---|
| | $(X_1, Y_1)$ | $(X_2, Y_2)$ | $(x, y)$ | $e_x$ | $e_y$ | $e$ |
| Sunny day with light | (336, 135) | (335, 265) | (333, 199) | 0 | 0 | 0 |
| Sunny day with backlight | (244, 206) | (270, 311) | (256, 258) | 0 | 2 | 2 |
| Overcast day with light | (489, 143) | (475, 214) | (484, 178) | 0 | 1 | 1 |
| | (213, 162) | (189, 265) | (209, 216) | 0 | 5 | 5 |
| Cloudy day with light | (331, 194) | (330, 287) | (329, 240) | 0 | 2 | 2 |
| | (89, 234) | (84, 289) | (86, 261) | 1 | 0 | 1 |

The test results did not exhibit an error between the picking point and optimal picking point obtained by the proposed method on a sunny day with a backlight. The localization accuracy was marginally higher than that of the backlight. The optimal picking-point pixel range was manually set. The branch center axis was used as the location of the optimal picking point (generally a rectangular range of 30 × 10 pixels). Owing to the changes in lighting conditions, sunny days with light improved the light intensity and brightness on the safflower, improving the localization of the picking point. However, the error between the obtained and optimal picking points was two pixels greater in backlight than on a sunny day with light. the backlighting effect increased the intensity and brightness of the safflower; however, the filament blended with the surrounding complex environment.

However, the glare from the backlight caused the filament to overlap with the background color, resulting in the localization of safflowers with the same color as the surrounding background. The pixel error in the row direction was small, and the optimal picking point was between cloudy and sunny conditions with light. Moreover, 1–5 pixel errors occurred between the column direction and optimal picking point. In particular, the pixel error was largest for cloudy light in the direction of the column. The main reason for this was that the light intensity on cloudy days was weak compared with that on overcast days; thus, the color of the picking-point site was not prominent. Therefore, the localization of the picking point was prone to deviation.

Taking a safflower image from a sunny day with light as an example, a depth-measurement test was conducted using the same target filament picking point at different depth distances. The depth distance between the center of gravity of the camera and filament picking point ranged from 200 to 700 mm. Fifty sets of depth-measurement errors are shown in Fig. 12 in the *X*- and *Z*-directions.

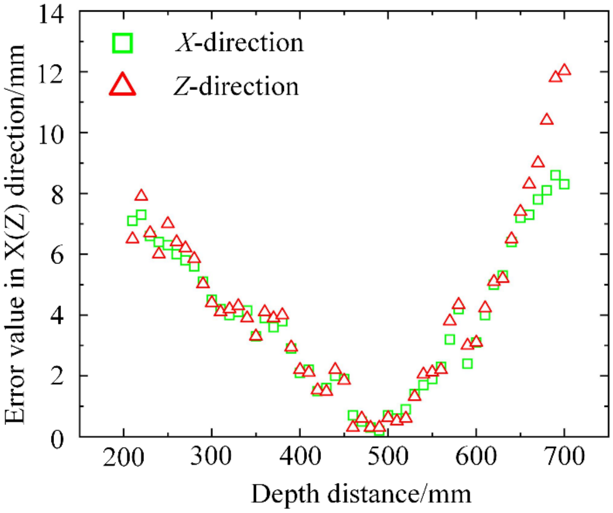

**Fig. 12.** Schematic representation of the error distribution of picking points in the global coordinate system.

As shown in Fig. 12, the depth-measurement error of the camera continuously fluctuated with an increase in the depth distance between the depth camera and target filament. In particular, the depth-measurement error was within 13 mm in the *Z* direction. When the depth distance was 450-510 mm from the depth camera to the target filament, the depth-measurement error was minimized in the *Z* direction to less than 1 mm. When the depth distance exceeded 640 mm, the depth-measurement error was greater than 7 mm and continued to increase with increasing depth distance. The depth-measurement error was within 9 mm in the *X* direction. When the depth distance of the camera from the target filament was 450–510 mm, the depth-measurement error was minimized in the *X* direction. With an increase in the depth distance, the measurement error increased. Therefore, the optimal depth-measurement distance was 450–510 mm between the depth camera and target filament. The target visual-localization error between these distances was minimized to meet the harvesting-accuracy requirements.

### *4.4. Picking-point real-time localization test*

The XYZ three-axis sliding table module was used as the motion platform, the end effector as the picking robot, and a depth camera as the vision core, as shown in Fig. 13. The SDC-DeepLabv3+ algorithm detected real-time safflower scenes in the field under different weather conditions. Taking the correct rate, error rate and leakage rate of real-time frame detection as evaluation indexes, the real-time safflower detection results are shown in Table 5.

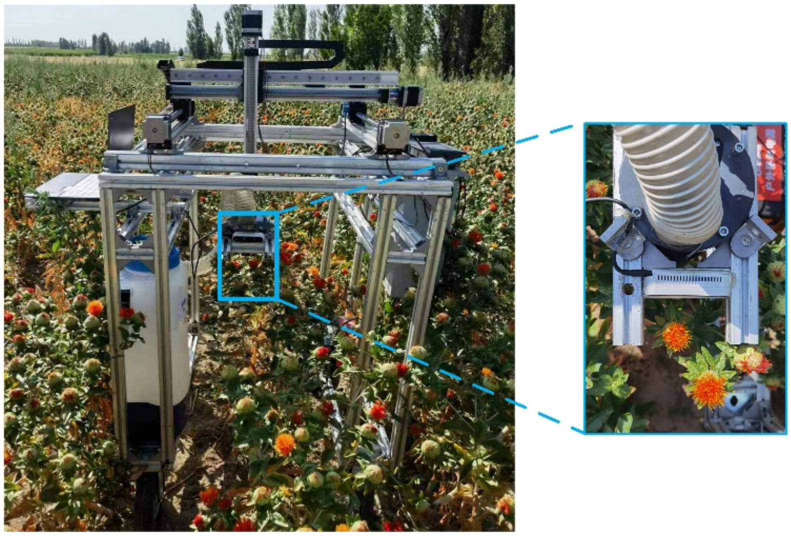

**Fig. 13.** Validation tests and effects of safflower-harvesting robots.

**Table 5.** Real-time detection results under different weather conditions.

| Types | No. | Frame detection correct rate/% | Frame detection error rate/% | Frame detection leakage rate/% |
|---|---|---|---|---|
| Sunny day with light | 119 | 98.32 | 0 | 4.20 |
| Sunny day with backlight | 107 | 94.39 | 1.87 | 3.74 |
| Overcast day with light | 128 | 85.16 | 7.81 | 7.03 |
| Cloudy day with light | 115 | 88.70 | 5.22 | 6.09 |

Sunny day with light was applied to the safflower to improve the light intensity and brightness, which made the safflower color and texture features more obvious with no error rate. The correct rate of real-time frame detection is higher than the other cases by at least 3.93%, and the leakage rate is lower by at least 0.46%. Although the sunny day with backlight effect improves the light intensity and brightness on safflower, the filament itself blends with the complex surrounding environment, which is not conducive to identification and localization. Further analysis revealed that cloudy day with light was weak, making safflower color features unremarkable and with less relevant information. However, overcast day with light caused the color and texture features of the filaments to be blurred. The correct rate of real-time frame detection was the lowest, and the error and leakage rate were also relatively high.

The improved DeepLabv3+ was verified to localize the safflower-filament picking points under different weather conditions [4], as listed in Table 6. The results showed that the filament-localization and -picking success rates were the highest on sunny days with light, at 98.33% and 96.67%, respectively; the localization success rate was 92.50%. The filament leakage picking rate was also the lowest, at 3.33%. Furthermore, the color and texture features of the filament were blurred because of the influence of light and other shading conditions on cloudy days. The filament-localization and -picking success rates were lower than those under other weather conditions by at least 1.67% and 2.5%, respectively. The filament damage rate exceeded 0.74%.

**Table 6.** Field-harvesting test data for filaments.

| Types | No. | Localization success rate/% | Picking success rate/% | Filament leakage rate/% | Filament damage rate/% |
|---|---|---|---|---|---|
| Sunny day with light | 120 | 98.33 | 96.67 | 3.33 | 7.07 |
| Sunny day with backlight | 120 | 95.00 | 92.50 | 7.50 | 6.01 |
| Overcast day with light | 120 | 87.50 | 85.83 | 14.17 | 7.81 |
| Cloudy day with light | 120 | 89.17 | 88.33 | 11.67 | 6.14 |

## 5. Discussion

Accurate and rapid detection of safflower filaments in complex environments is of utmost importance for harvesting on schedule and preventing wilting [1, 3]. Previous studies have demonstrated the identification and detection using deep learning methods combining background color and contour with top localization methods, which are prone to blindness and uncertainty in locating the filament picking point [59]. Previous studies have demonstrated improvements in algorithms that can alleviate the impact of complex backgrounds on the detection task [4, 8, 13]. However, these methods usually increase the model size and computational complexity, resulting in slower detection and causing severe filament damage. To solve the difficulty of accurately locating safflower-filament picking points with near-color backgrounds and irregular contours in complex environments, this study proposes a localization method for safflower filament picking-point detection using an improved DeepLabv3+ algorithm.

(i) SDC-DeepLabv3+ uses a lightweight network, ShuffletNetV2, as the backbone network for detection. The background areas, contour edges, and interference with filament, fruit ball, and branch segmentation were reduced.

(ii) The dilated depth-separable convolution added three convolutional branches at different sampling rates. A CBAM was introduced to extract information on safflower features under different receptive fields.

(iii) The line segment that minimizes the distance between the barycenter coordinates and straight line was solved. The line was considered as the line position of the picking point and projected onto this line segment with the midpoint as the picking point.

The test results show that SDC-DeepLabv3+ enhanced the mIoU and mPA to 95.84% and 96.87%, respectively, compared with the original algorithm. This significantly improved the linkage of safflower filament features in the channel and space, suppressed background interference, and highlighted the filament more prominently. Furthermore, the FPS increased by 12.16 percentage points and the parameters decreased by 192.78 MB.

Compared with the other five segmentation algorithms, the SDC-DeepLabv3+ algorithm increased the mIoU and mPA by at least 1.71% and 1.23%, respectively [23, 24, 25, 32]. The FPS increased by at least 9.11 f/s. In addition, the parameters with only 13.54 MB were significantly smaller than those of the other models. A safflower filament picking-point test was performed under various weather conditions. The test results showed that the best distance was 450–510 mm, average localization success rate was 92.50%, and average picking success rate was 90.83%. This indicates that the performance index of the filament picking-point localization method improved significantly based on the proposed DeepLabv3+ algorithm, which exhibited a high performance stability and good adaptability.

In addition, the performance of the model on the test dataset is better than in training. The main reason for this is too much model regularization [60-61]. The difference between the model at training, which cuts out the random set of classifiers due to too much Dropout. The model at validation is large. However, Dropout will be automatically turned off during validation and all weak classifiers will be used, resulting in a relatively high test accuracy. Another reason is the lag of small batch statistics and the data preprocessing of the training set, such as geometric transformations (translation, rotation) with color transformations (contrast, brightness) and other operations [62-63]. Excessive preprocessing leads to changes in the distribution of the training set, making the test accuracy relatively high.

## 6. Conclusions and future work

In this study, an SDC-DeepLabv3+ filament picking-point detection and localization method was developed by improving the algorithm in multiple ways to address the problem of near-color backgrounds and irregular contours of the picking point. The test verified that the algorithm had a good segmentation performance and obvious performance improvements over similar existing networks. However, extensive work can be carried out for in-depth research in the future:

(1) This study only addressed safflower varieties grown in Xinjiang, China. Owing to the differences in the physical features related to the appearance of different safflower varieties, the robustness of the segmentation capability of the proposed algorithm must be verified. However, extending the application to localize more safflower varieties or even similar crops is necessary to obtain more comprehensive results.

(2) The CBAM was introduced to improve the feature extraction of the improved algorithm. In future studies, the effects produced when CBAM is inserted at different locations in the backbone network should be considered. In addition, the channel or spatial attention mechanism should be considered at different positions in the algorithm. The extent of influence of the attention mechanism should be further derived from the segmentation performance of the improved algorithm.

## Acknowledgments

**General**: The authors acknowledge the Key Laboratory of Xinjiang Intelligent Agricultural Equipment, China, for their assistance in conducting field experiments and Associate Professor Yunchao Tang of the Zhongkai University of Agriculture and Engineering, China for writing guidance.

**Author contributions: Zhenyu Xing**: Conceptualization, Methodology, Formal analysis, Writing - original draft, Visualization, Supervision. **Zhenguo Zhang**: Funding acquisition, Writing - review & editing, Data curation. **Yunze Wang**: Software, Formal analysis. **Peng**

**Xu:** Software, Formal analysis. **Quanfeng Guo**: Conceptualization, Methodology. **Ruimeng Shi**: Software, Writing - review & editing. **Chao Zeng:** Resources, Visualization.

**Funding:** This work was supported in part by the National Natural Science Foundation of China [grant numbers 52265041 and 31901417]; the Open Subjects of Zhejiang Provincial Key Laboratory for Agricultural Intelligent Equipment and Robotics, China [grant number 2022ZJZD2202], and the Graduate School-level Research and Innovation Program of Xinjiang Agricultural University, China [grant number XJAUGRI2023021].

**Competing interests:** The authors declare that they have no competing interests.

**Data Availability:** The image dataset used to support the findings of this study is available from the corresponding authors upon request.

## References

1. Zhang ZG, Xing ZY, Yang SP, Feng N, Liang RQ, Zhao MY. Design and experiments of the circular arc progressive type harvester for the safflower filaments. *Transactions of the Chinese Society of Agricultural Engineering*. 2022； 38(17): 10-21.

2. Gongora B, de Souza SNM, Bassegio D, Santos RF, Siqueira JAC, Bariccatti RA, Gurgacz F, Secco D, Tokura LK, Sequinel R. Comparison of emissions and engine performance of safflower and commercial biodiesels. *Industrial Crops and Products*. 2022; 179, 114680.

3. Zhang ZG, Zhao MY, Xing ZY, Liu XF. Design and test of double-acting opposite direction cutting end effector for safflower harvester. *Transactions of the Chinese Society for Agricultural Machinery.* 2022; 53(12), 160-170.

4. Zhang ZG, Shi RM, Xing ZY, Guo QF, Zeng C. Improved Faster Region-Based Convolutional Neural Networks (R-CNN) model based on split attention for the detection of safflower filaments in natural environments. *Agronomy*. 2023; 13(10):2596.

5. Raeisi N, Beheshti B, Sharifnasab H. Design construction and evaluation of a picking safflower harvesting machine. *Agricultural Mechanization and Systems Research*. 2020; 21(74), 237-250.

6. Magistri F, Weyler J, Gogoll D, Lottes P, Behley J, Petrinic N, Stachniss C. From one field to another—Unsupervised domain adaptation for semantic segmentation in agricultural robotics. *Computers and Electronics in Agriculture*. 2023; 212, 108114.

7. Abbood WT, Abdullah OI, Khalid EA. A real-time automated sorting of robotic vision system based on the interactive design approach. *International Journal on Interactive Design and Manufacturing (IJIDeM)*. 2020; 14, 201-209.

8. Zhang ZG, Xing ZY, Zhao MY, Yang SP, Guo QF, Shi RM, Zeng C. Detecting safflower filaments using an improved YOLOv3 under complex environments. *Transactions of the Chinese Society of Agricultural Engineering.* 2023; 39 (3), 162–170.

9. Sun ML, Xu LC, Chen XD, Ji Z, Zheng YJ, Jia WK. BFP Net: Balanced Feature Pyramid Network for Small Apple Detection in Complex Orchard Environment. *Plant Phenomics*. 2022; 2022.

10. Sanaeifar A, Guindo ML, Bakhshipour A, Fazayeli H, Li X, Yang C. Advancing precision agriculture: The potential of deep learning for cereal plant head detection. *Computers and Electronics in Agriculture*. 2023; 209, 107875.

11. Jia WK, Zhang Y, Lian J, Zheng YJ, Zhao DA, Li CJ. Apple harvesting robot under information technology: A review. *International Journal of Advanced Robotic Systems*. 2020; 17(3), 255688461.

12. Rahim UF, Utsumi T, Mineno H. Deep learning-based accurate grapevine inflorescence and flower quantification in unstructured vineyard images acquired using a mobile sensing platform. *Computers and Electronics in Agriculture*. 2022; 198, 107088.

13. Gao GM, Guo H, Zhou W, Luo D, Zhang J. Design of a control system for a safflower picking robot and research on multisensory fusion positioning. *Engenharia Agrícola*. 2023; 43, e20210238.

14. Dias PA, Tabb A, Medeiros H. Multispecies fruit flower detection using a refined semantic segmentation network. *IEEE robotics and automation letters*. 2018; 3(4), 3003-3010.

15. Tian M, Chen H, Wanc O. Detection and recognition of flower image based on SSD network in video stream. *Journal of Physics Conference Series*. 2019; 1237, 32-45.

16. Williams H, Nejati M, Hussein S, Penhall N, Lim JY, Jones MH, Bell J, Ahn HS, Bradley S, Schaare P, et al. Autonomous pollination of individual kiwifruit flowers: Toward a robotic kiwifruit pollinator. *Journal of Field Robotics*. 2020; 37(2), 246-262.

17. Palacios F, Bueno G, Salido J, Diago MP, Hernández I, Tardaguila J. Automated grapevine flower detection and quantification method based on computer vision and deep learning from on-the-go imaging using a mobile sensing platform under field conditions. *Computers and Electronics in Agriculture*. 2020; 178, 105796.

18. Zhou YC, Xu TY, Zheng W, Deng HB. Classification and recognition approaches of tomato main organs based on DCNN. Transactions of the Chinese Society of Agricultural Engineering. 2017; 33(15), 219-226.

19. Xiong JT, Liu BL, Zhong Z, Chen SM, Zheng ZH. Litchi flower and leaf segmentation and recognition based on deep semantic segmentation. *Transactions of the Chinese Society for Agricultural Machinery*. 2021; 52(6), 252-258.

20. Zhao CJ, Wen CW, Lin S, Guo WZ, Long JH. Tomato florescence recognition and detection method based on cascaded neural network. *Transactions of the Chinese Society of Agricultural Engineering*. 2020; 36(24), 143-152.

21. Zabawa L, Kicherer A, Klingbeil L, Töpfer R, Roscher R, Kuhlmann H. Image-based analysis of yield parameters in viticulture. *Biosystems Engineering*. 2022; 218, 94-109.

22. Toğaçar M, Ergen B. Classification of cloud images by using super resolution, semantic segmentation approaches and binary sailfish optimization method with deep learning model. *Computers and Electronics in Agriculture*. 2022; 193, 106724.

23. Kamal S, Shende VG, Swaroopa K, Bindhu Madhavi P, Akram PS, Pant K, Patil SD, Sahile K. FCN Network-Based Weed and Crop Segmentation for IoT-Aided Agriculture Applications. *Wireless Communications and Mobile Computing*. 2022.

24. Sun J, Yang S, Gao X, Ou D, Tian Z, Wu J, Wang M. Masa-segnet: A semantic segmentation network for polsar images. *Remote Sensing*. 2023; 15(14), 3662.

25. Fan X, Yan C, Fan J, Wang N. Improved U-net remote sensing classification algorithm fusing attention and multiscale features. *Remote Sensing*. 2022; 14(15), 3591.

26. Wang Y, Gu LJ, Jiang T, Gao F. MDE-UNet: A Multitask Deformable UNet Combined Enhancement Network for Farmland Boundary Segmentation. *IEEE Geoscience and Remote Sensing Letters*. 2023; 20, 1-5.

27. Zhou T, Fu HZ, Gong C, Shao L, Porikli F, Ling HB, Shen JB. Consistency and diversity induced human motion segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2022; 45(1), 197-210.

28. Tian Z, Zhang BW, Chen H, Shen CH. Instance and panoptic segmentation using conditional convolutions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2022; 45(1), 669-680.

29. Palacios F, Melo-Pinto P, Diago MP, Tardaguila J. Deep learning and computer vision for assessing the number of actual berries in commercial vineyards. *Biosystems Engineering*. 2022; 218, 175-188.

30. Li JY, Zeng HY, Huang CX, Wu LB, Ma J, Zhou BB, Ye DP, Weng HY. Noninvasive Detection of Salt Stress in Cotton Seedlings by Combining Multicolor Fluorescence–Multispectral Reflectance Imaging with EfficientNet-OB2. *Plant Phenomics*. 2023; 5: 0125.

31. Zhu SS, Ma WL, Lu JW, Ren B, Wang CY, Wang JL. A novel approach for apple leaf disease image segmentation in complex scenes based on two-stage DeepLabv3+ with adaptive loss. *Computers and Electronics in Agriculture*. 2023; 204, 107539.

32. Peng HX, Zhong JR, Liu HN, Li J, Yao MW, Zhang X. ResDense-focal-DeepLabV3+ enabled litchi branch semantic segmentation for robotic harvesting. *Computers and Electronics in Agriculture*. 2023; 206, 107691.

33. Campos Y, Sossa H, Pajares G. Spatio-temporal analysis for obstacle detection inagricultural videos. *Applied Soft Computing*. 2016; 45, 86–97.

34. Liu Y, Li Y, Yi X, Hu Z, Zhang H, Liu Y. Micro-expression recognition model based on TV-L1 optical flow method and improved ShuffleNet. *Scientific Reports*. 2022; 12(1), 17522.

35. Wang C, Du P, Wu H, Li J, Zhao C, Zhu H. A cucumber leaf disease severity classification method based on the fusion of DeepLabV3+ and U-Net. *Computers and Electronics in Agriculture*. 2021; 189, 106373.

36. Hu Z, Zhao J, Luo Y, Ou J. Semantic SLAM based on improved DeepLabv3+ in dynamic scenarios. *IEEE Access*. 2022; 10, 21160-21168.

37. Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*. 2017; 39 (12), 2481–2495.

38. Xi D, Qin Y, Wang Z. Attention Deeplabv3 model and its application into gear pitting measurement. *Journal of Intelligent & Fuzzy Systems*. 2022; 42(4), 3107-3120.

39. Kim YH, Park KR. MTS-CNN: Multi-task semantic segmentation-convolutional neural network for detecting crops and weeds. *Computers and Electronics in Agriculture*. 2022; 199, 107146.

40. Balasooriya N, Dowden B, Chen J, De Silva O, Huang W. In-situ sea ice detection using DeepLabv3 semantic segmentation[C]//OCEANS 2021: San Diego–Porto. IEEE, 2021: 1-7.

41. Sun J, Zhou J, He Y, Jia H, Liang Z. RL-DeepLabv3+: A lightweight rice lodging semantic segmentation model for unmanned rice harvester. *Computers and Electronics in Agriculture*. 2023; 209, 107823.

42. Yang H, Liu J, Mei G, Yang D, Deng X, Duan C. Research on real-time detection method of rail corrugation based on improved ShuffleNet V2. *Engineering Applications of Artificial Intelligence*. 2023; 126, 106825.

43. Han J, Yang Y. L-Net: lightweight and fast object detector-based ShuffleNetV2. *Journal of Real-time Image Processing*. 2021; 18(6): 2527-2538.

44. Zhang PL, Sun XB, Zhang DH, Yang YC, Wang ZH. Lightweight Deep Learning Models for High-Precision Rice Seedling Segmentation from UAV-Based Multispectral Images. *Plant Phenomics*. 2023; 5: 0123.

45. Tassis LM, de Souza JET, Krohling RA. A deep learning approach combining instance and semantic segmentation to identify diseases and pests of coffee leaves from in-field images. *Computers and Electronics in Agriculture*. 2021; 186, 106191.

46. Zhao RN, Guan YJ, Lu YQ, Ji Z, Yin X, Jia WK. FCOS-LSC: A Novel Model for Green Fruit Detection in a Complex Orchard Environment. *Plant Phenomics*. 2023; 5: 0069.

47. Kumari L, Singh S, Rathore VVS, Sharma A. A Lexicon and Depth-Wise Separable Convolution Based Handwritten Text Recognition System. I*n International Conference on Image and Vision Computing New Zealand*. Cham: Springer Nature Switzerland. 2022; 442-456.

48. Jadon S. A survey of loss functions for semantic segmentation. *In 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*. IEEE. 2020; 1–7.

49. Meng F, Li J, Zhang Y, Qi S, Tang Y. Transforming unmanned pineapple picking with spatio-temporal convolutional neural networks. *Computers and Electronics in Agriculture*. 2023; 214, 108298.

50. Zhang D, Ding Y, Chen P, Zhang X, Pan Z, Liang D. Automatic extraction of wheat lodging area based on transfer learning method and deeplabv3+ network. *Computers and Electronics in Agriculture*. 2020; 179, 105845.

51. Wu F, Yang Z, Mo X, Wu Z, Tang W, Duan J, Zou X. Detection and counting of banana bunches by integrating deep learning and classic image-processing algorithms. *Computers and Electronics in Agriculture*. 2023; 209, 107827.

52. Jia Z, Cai X, Jiao Z. Multi-modal physiological signals based squeeze-and-excitation network with domain adversarial learning for sleep staging. *IEEE Sensors Journal*. 2022; 22(4), 3464-3471.

53. Wu Z, Yang R, Gao F, Wang W, Fu L, Li R. Segmentation of abnormal leaves of hydroponic lettuce based on DeepLabV3+ for robotic sorting. *Computers and Electronics in Agriculture*. 2021; 190, 106443.

54. Xue XY, Luo Q, Bu MF, Li Z, Lyu SL, Song SR. Citrus tree canopy segmentation of orchard spraying robot based on RGB-D image and the improved DeepLabv3+. *Agronomy*, 2023; 13(8), 2059.

55. Lin G, Tang Y, Zou X, Wang C. Three-dimensional reconstruction of guava fruits and branches using instance segmentation and geometry analysis. *Computers and Electronics in Agriculture*. 2021; 184, 106107.

56. Luo LF, Zou XJ, Xiong JT, Zhang Y, Peng HX, Lin GC. Automatic positioning for picking point of grape picking robot in natural environment. *Transactions of the Chinese Society of Agricultural Engineering*. 2015; 31(2), 14-21.

57. Yu Y, Zhang K, Yang L, Zhang D. Fruit detection for strawberry harvesting robot in non-structural environment based on Mask-RCNN. *Computers and Electronics in Agriculture*. 2019; 163, 104846.

58. Du WS, Liu P. Instance Segmentation and Berry Counting of Table Grape before Thinning Based on AS-SwinT. *Plant Phenomics*. 2023; 5: 0085.

59. Ye XQ, Pan J, Liu GS, Shao F. Exploring the Close-Range Detection of UAV-Based Images on Pine Wilt Disease by an Improved Deep Learning Method. *Plant Phenomics*, 2023; 5: 0129.

60. Wen Y, Xue JL, Sun H, Song Y, Lv PF, Liu SH, Chu YY, Zhang, TY. High-precision target ranging in complex orchard scenes by utilizing semantic segmentation results and binocular vision. *Computers and Electronics in Agriculture*, 2023; 215, 108440.

61. Xu CH, Gao MF, Yan JW, Jin YX, Yang GJ, Wu WB. MP-Net: An efficient and precise multi-layer pyramid crop classification network for remote sensing images. *Computers and Electronics in Agriculture*, 2023; 212, 108065.

62. Shen C, Liu LT, Zhu LX, Kang J, Wang N, Shao LM. High-throughput in situ root image segmentation based on the improved DeepLabv3+ method. *Frontiers in Plant Science*, 2020; 11, 576791.

63. Yu HL, Che MH, Yu H, Zhang J. Development of weed detection method in soybean fields utilizing improved deeplabv3+ platform. *Agronomy*, 2022; 12(11), 2889.