AgriEngineering

MDPI

Article

MA-DeepLabV3+: A Lightweight Semantic Segmentation Model for Jixin Fruit Maturity Recognition

Leilei Deng *, Jiyu Xu, Di Fang and Qi Hou

College of Information and Technology, Jilin Agricultural University, Changchun 130118, China; xjy1253@mails.jlau.edu.cn (J.X.); 20231241@mails.jlau.edu.cn (D.F.); houqi@mails.jlau.edu.cn (Q.H.)
* Correspondence: dengleilei@jlau.edu.cn

Abstract

Jixin fruit (*Malus domestica* '*Jixin*') is a high-value specialty fruit of significant economic importance in northeastern and northwestern China. Automatic recognition of fruit maturity is a critical prerequisite for intelligent harvesting. However, challenges inherent to field environments—including heterogeneous ripeness levels among fruits on the same plant, gradual color transitions during maturation that result in ambiguous boundaries, and occlusion by branches and foliage—render traditional image recognition methods inadequate for simultaneously achieving high recognition accuracy and computational efficiency. Although existing deep learning models can improve recognition accuracy, their substantial computational demands and high hardware requirements preclude deployment on resource-constrained embedded devices such as harvesting robots. To achieve the rapid and accurate identification of *Jixin* fruit maturity, this study proposes Multi-Attention DeepLabV3+ (MA-DeepLabV3+), a streamlined semantic segmentation framework derived from an enhanced DeepLabV3+ model. First, a lightweight backbone network is adopted to replace the original complex structure, substantially reducing computational burden. Second, a Multi-Scale Self-Attention Module (MSAM) is proposed to replace the traditional Atrous Spatial Pyramid Pooling (ASPP) structure, reducing network computational cost while enhancing the model's perception capability for fruits of different scales. Finally, an Attention and Convolution Fusion Module (ACFM) is introduced in the decoding stage to significantly improve boundary segmentation accuracy and small target recognition ability. Experimental results on a self-constructed *Jixin* fruit dataset demonstrated that the proposed MA-DeepLabV3+ model achieves an mIoU of 86.13%, mPA of 91.29%, and F1 score of 90.05%, while reducing the number of parameters by 89.8% and computational cost by 55.3% compared to the original model. The inference speed increased from 41 frames per second (FPS) to 81 FPS, representing an approximately two-fold improvement. The model memory footprint is only 21 MB, demonstrating potential for deployment on embedded devices such as harvesting robots. Experimental results demonstrate that the proposed model achieves significant reductions in computational complexity while maintaining high segmentation accuracy, exhibiting robust performance particularly in complex scenarios involving color gradients, ambiguous boundaries, and occlusion. This study provides technical support for the development of intelligent *Jixin* fruit harvesting equipment and offers a valuable reference for the application of lightweight deep learning models in smart agriculture.

Keywords: semantic segmentation; lightweight deep learning; maturity recognition; *Jixin* fruit; precision agriculture

Check for updates

Academic Editor: Ray E. Sheriff

Received: 24 November 2025
Revised: 24 December 2025
Accepted: 20 January 2026
Published: 23 January 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Jixin fruit, also known as Jinxiu Begonia, Dian thorn olive, Manban tree, and Stenosepalous Pittosporum, with the Latin name *Xantolis stenosepala*, is a type of pocket apple and an updated variety of Hanfu apple. It is named after the conical shape of its fruit, resembling a chicken heart, combining both ornamental and edible characteristics. The *Jixin* fruit variety exhibits excellent yield characteristics, with yields reaching 2000–2500 kg per mu (approximately 30,000–37,500 kg/ha) during peak production periods. At wholesale market prices of approximately 10–15 CNY/kg, the per-mu economic benefit can exceed 10,000 CNY, making it an important income-generating industry for rural communities in northern China. It is one of the regional specialty fruit crops in Northeast and Northwest China. Currently, the harvesting of *Jixin* fruit mainly relies on traditional manual methods. This production mode, which is highly dependent on manpower, has become a common bottleneck restricting the development of the specialty fruit industry [1]. With the continuous growth of market demand, *Jixin* fruit production faces multiple challenges, such as uneven fruit maturity, inefficient picking, and rising labor costs, leading to increased post-harvest losses, escalating labor costs, and unstable product quality, which seriously restrict the economic benefits and sustainable development of the industry. In recent years, machine vision technology has been widely applied in the field of fruit and vegetable harvesting, and various fruit-picking robots have been introduced to the market [2]. However, existing picking robot systems generally suffer from high equipment costs, high system complexity, and poor adaptability to specific orchard environments, making it difficult to promote their application in small and medium-scale orchards. Therefore, this study aims to develop an efficient, accurate, and lightweight *Jixin* fruit maturity recognition model to achieve automatic grading of *Jixin* fruit maturity through deep learning methods, verify the recognition performance of the lightweight model in actual orchard scenarios, and provide theoretical support and technical reference for algorithm optimization and mobile application of intelligent picking systems in the future.

In recent years, domestic and foreign scholars have conducted extensive research on fruit segmentation and maturity assessment, including traditional methods and currently popular deep learning segmentation methods. In the field of fruit segmentation, traditional methods [3,4] primarily rely on image processing and feature extraction algorithms. Feng et al. [5] pointed out that although traditional machine learning-based technologies have seen improvements in speed, accuracy, and robustness, they remain sensitive to abnormal data inputs, require pre-set parameters, and the final classification performance is closely tied to parameter configurations. Moreover, current mainstream image segmentation and classifier solutions based on traditional machine learning are often tailored to specific scenarios, lacking universality and performing poorly in multi-class classification tasks. Payne et al. [6] conducted detailed research on the assessment of mango maturity and found that the color space threshold segmentation technology is extremely sensitive to ambient light changes. They specifically demonstrated that the method based on color space threshold would experience a sharp decline in performance under outdoor non-uniform lighting conditions, which confirmed the fragility of traditional methods when dealing with "color gradients" and "lighting variations." Tian et al. [7], in their review of apple recognition, concluded that although algorithms relying on contour analysis and template matching can identify individual apples, they fundamentally lack the ability to distinguish the complex boundaries of densely clustered apples. Traditional computer vision cannot correctly separate touching fruits, which is considered a major obstacle to deploying robust automation systems in dense orchards. These studies highlight the inherent fragility of traditional image processing methods, necessitating more adaptive and robust solutions for reliable fruit segmentation.

With the rapid development of deep learning technology, semantic segmentation methods based on Convolutional Neural Networks (CNNs) [8–10] have achieved remarkable progress in the field of image processing, and fruit maturity detection is no longer limited to traditional methods. Xie et al. [11] proposed ECD-DeepLabv3+, a lightweight semantic segmentation model for postharvest maturity detection of sugar apples. The model employs MobileNetV2 as the backbone network and integrates ECA and CA attention modules with a Dense ASPP structure, achieving an mIoU of 89.95%. However, this method was validated only on a self-constructed dataset with three maturity levels, and its generalization capability requires further verification. Nuanmeesri et al. [12] proposed a Hybrid Attention Convolutional Neural Network (HACNN) for avocado maturity classification, combining spatial, channel, and self-attention modules. The model achieved a test accuracy of 91.25% with a memory footprint of 59.81 MB and an inference time of 280.67 ms. However, this method was validated on a single fruit species only, exhibiting limited cross-variety generalization capability and presenting trade-offs between high accuracy and lightweight design. To improve the model's lightweight and real-time performance, Chen et al. [13] proposed a lightweight semantic segmentation model based on an improved DeepLabv3+ for young plum stem recognition and picking point localization. The model adopts MobileNetV2 as the backbone network and integrates CBAM attention modules with a DenseASPP structure, achieving an MIoU of 86.13% and a picking point localization success rate of 88.8%. However, this method was validated on a specific fruit only, and its performance under extreme occlusion conditions requires improvement. Hou et al. [14] proposed LM-DeepLabV3+ on this basis, which combines multi-scale feature interaction modules with lightweight attention mechanisms, achieving a significant reduction in model complexity while maintaining high accuracy. Liu et al. [15] proposed MFA-DeepLabV3+, which introduces SE modules in the decoder structure. This method improves the feature expression capability to some extent, but its spatial attention capability is relatively limited. In terms of attention mechanisms, Cao et al. [16] replaced the YOLOv5n backbone network with FasterNet and integrated MobileViT, CBAM attention mechanisms, and the SPPELAN module, achieving a detection precision of 98.94% and an mAP of 99.43%. However, the detection frame rate was only 16.61 FPS, indicating poor real-time performance, and the model size increased to 53.22 MB, limiting the lightweight effect. Wang et al. [17] proposed the ECA-Net module, which achieves channel feature weighting at minimal parameter cost, providing an efficient alternative for lightweight models. In terms of feature fusion, Chen et al. [18] proposed FAFNet, which adopts cross-layer feature fusion concepts and achieves efficient integration of multi-modal data, validating the advantages of feature hierarchical fusion in complex semantic segmentation.

In summary, existing research in fruit maturity segmentation still exhibits several limitations. Although current lightweight models reduce computational requirements, their segmentation accuracy degrades in complex scenarios, making it difficult to balance real-time performance with accuracy. Furthermore, traditional ASPP modules employ dilated convolutions with fixed dilation rates, which cannot adaptively capture semantic information across different fruit scales, particularly limiting performance in scenarios with significant fruit size variations. Additionally, existing decoder architectures suffer from information loss during high-level and low-level feature fusion, resulting in difficulties in precisely segmenting ambiguous boundaries between semi-ripe and ripe fruits caused by color gradients. To address these issues, the main objectives of this study are as follows: (1) to develop a lightweight fruit maturity segmentation model with fewer than 10 M parameters capable of real-time operation on resource-constrained embedded devices; (2) to design a multi-scale feature perception module to enhance the model's recognition capability for fruits of different scales and maturity stages; (3) to optimize high-level and

low-level feature fusion strategies to improve segmentation accuracy in color gradient regions and ambiguous boundary scenarios.

Based on this, this study proposed the MA-DeepLabV3+ model, with specific improvements including replacing the backbone network of the DeepLabV3+ model with MobileNetV2 [19] to achieve lightweight, replacing Atrous Spatial Pyramid Pooling (ASPP) [20] with the Multiscale Self-Attention Module (MSAM) [21] to achieve cross-scale semantic interaction and target perception, while introducing the Attention and Convolution Fusion Module (ACFM) [22] in the decoding stage, enhancing the connection between high and low-level features through attention-guided cross-layer feature fusion, thus improving boundary details and small target recognition capabilities while maintaining lightweight. This research provides technical support for the intelligent harvesting of *Jixin* fruit. The lightweight design of the proposed MA-DeepLabV3+ model greatly reduces the hardware cost and energy consumption of intelligent harvesting systems, providing a feasible solution for promoting the digital transformation of niche specialty fruit industries. Through sufficient verification on the self-built *Jixin* fruit dataset, the model achieves an mIoU of 86.13% while reducing the number of parameters to 5.58 M and computational cost to 74.64 GFLOPs, achieving an optimal balance between accuracy and efficiency. This achievement lays an algorithmic foundation for the future development of *Jixin* fruit intelligent picking robot systems and can provide a reference for maturity recognition tasks of other niche fruits.

2. Materials and Methods

2.1. Dataset

The *Jixin* fruit dataset in this study was collected from a plantation in Caomiaozi Village, Beidahu Town, Yongji County, Jilin Province, China (126°23′ E, 43°25′ N, elevation approximately 320 m). The geographical location of the collection site is illustrated in Figure 1. Data collection was conducted from August to October 2024, covering the entire maturation cycle of *Jixin* fruit. Image acquisition was performed under three weather conditions—sunny, cloudy, and overcast—during the time periods of 8:00–11:00 and 14:00–17:00 daily to ensure diversity in illumination conditions. The average temperature during the collection period ranged from 15 °C to 25 °C, with relative humidity between 50% and 75%.

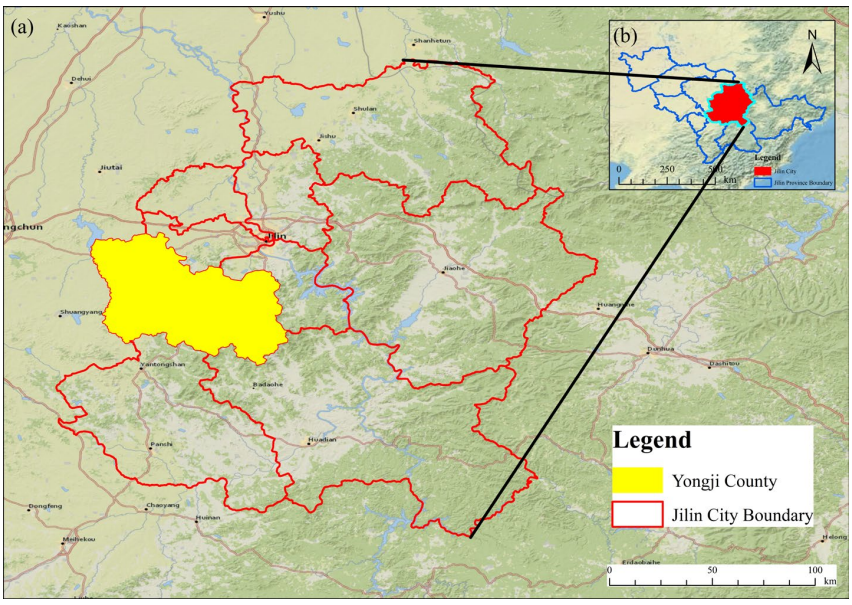


Figure 1. Geographic location of the data collection site. (**a**): Yongji County; (**b**): Jilin Province.

According to harvesting requirements, the maturity of *Jixin* fruit was defined as unripe, semi-ripe, and ripe stages, with the ripe stage being the optimal harvesting period. Samples were collected using a Huawei P50 Pro smartphone (Huawei, Shenzhen, China) with a camera resolution of 4032 × 2268 pixels. To ensure the diversity of *Jixin* fruit samples, a total of 844 images of *Jixin* fruit at different maturity stages under various natural environments were collected, including 277 images of the unripe stage, 342 images of the semi-ripe stage, and 225 images of the ripe stage. Specific examples of *Jixin* fruit images are shown in Figure 2. Data collection was conducted from July to September 2024, covering the main growth and maturity stages of *Jixin* fruit. Images were collected under natural lighting conditions, with shooting distances ranging from 0.3 to 1.5 m, and shooting angles including frontal, side, and oblique upward views to ensure the model's robustness to different viewpoints.



Figure 2. Examples from the *Jixin* fruit image dataset. (**a**) Unripe fruit, (**b**) semi-ripe fruit, (**c**) ripe fruit, (**d**) mixed ripe fruit, (**e**) fruit under low light conditions, (**f**) fruit under strong light conditions, (**g**) fruit under leaf obstruction conditions, and (**h**) fuzzy processing of fruit.

The collected *Jixin* fruit image dataset was divided into training, validation, and test sets in an 8:1:1 ratio, ensuring uniform distribution of different samples in the dataset. To ensure uniformity of model input, all images were resized to 512 × 512 pixels. X-AnyLabeling was selected as the annotation tool, and images were annotated according to the VOC dataset annotation format, with unripe fruits represented in yellow, semi-ripe fruits in green, and ripe fruits in red. To quantitatively evaluate inter-annotator consistency, this study employed Cohen's Kappa coefficient. The specific procedure was as follows: 100 images were randomly sampled from the dataset and independently annotated by three annotators; the Kappa coefficient was then calculated between each pair of annotators; finally, the average value was computed as the overall consistency metric. Experimental results indicate an average Kappa coefficient of 0.92, which falls within the "Almost Perfect Agreement" category ($\kappa > 0.81$) according to the classification criteria proposed by Landis and Koch. Specific annotation examples are shown in Figure 3, validating the reliability of the annotation results.

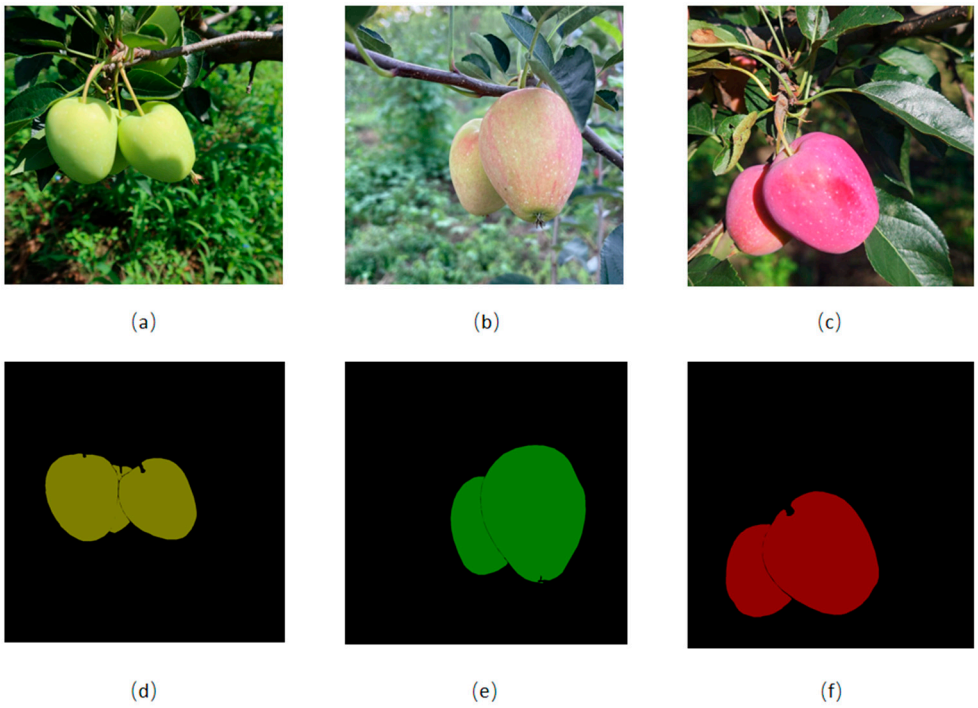


Figure 3. Annotation examples. (**a**) Original image of unripe fruit, (**b**) original image of semi-ripe fruit, (**c**) original image of ripe fruit, (**d**) annotation of unripe fruit, (**e**) annotation of semi-ripe fruit, and (**f**) annotation of ripe fruit.

2.2. Fruit Segmentation Model

2.2.1. MA-DeepLabV3+

DeepLabV3+ [23] is a classic semantic segmentation network that obtains more precise segmentation boundaries by adding an efficient decoder module on the basis of DeepLabV3. This model is based on spatial pyramid pooling technology and integrates the advantages of multiple models to construct the core network architecture. The core structure of DeepLabV3+ consists of two parts: the Atrous Spatial Pyramid Pooling (ASPP) module and the encoder-decoder structure. The former is used to capture rich semantic information through feature pooling at different resolutions; the latter is used to obtain more accurate boundaries. This structure has achieved excellent results in various semantic segmentation tasks. However, its backbone networks, such as Xception [24] or ResNet [25] have relatively complex structures, a large number of parameters, and high computational overhead, which are unfavorable for deployment on resource-constrained devices and difficult to meet the real-time requirements of scenarios such as fruit maturity grading. Furthermore, although the Atrous Spatial Pyramid Pooling can extract multi-scale contextual features, its ability to focus on inter-channel correlations and spatial salient regions is limited, which can easily lead to uneven weight distribution of target regions during the feature extraction process, especially when the color changes on the fruit surface are subtle, or the lighting is uneven, the segmentation accuracy will decrease.

To address the above issues, this paper proposed the MA-DeepLabV3+ model. First, the backbone network was replaced from Xception to the lightweight MobileNetV2. MobileNetV2 employed 17 inverted residual blocks distributed across 7 groups to extract multi-scale features. This study extracted low-level features containing detailed information from shallow Block 5–7 (resolution $28 \times 28 \times 32$) and high-level features containing semantic information from deep Block 17 (resolution $7 \times 7 \times 320$), which serve as Low-level and High-level features, respectively, for subsequent modules. This greatly reduces computational cost and number of parameters while maintaining high feature extraction

capability, making it particularly suitable for real-time applications on embedded and mobile devices. Next, ASPP was replaced with the MSAM. MSAM dynamically selected and weighted feature information at different scales by introducing attention mechanisms, thereby enhancing the model's capability to capture multi-scale features. After the MSAM module, the ACFM was introduced to further enhance feature fusion capability, especially in the fusion between low-level and high-level features. The ACFM module selectively fused features at different scales through an adaptive weighting mechanism, thereby improving segmentation accuracy and detail recovery capability. Compared to the simple concatenation operation in the original DeepLabV3+, ACFM can more intelligently select and fuse features. These improvements enable MA-DeepLabV3+ to significantly improve segmentation accuracy while ensuring computational efficiency, and further enhance feature fusion capability in detail recovery, especially suitable for segmentation application scenarios of *Jixin* fruit maturity. The overall architecture of the MA-DeepLabV3+ model is shown in Figure 4.

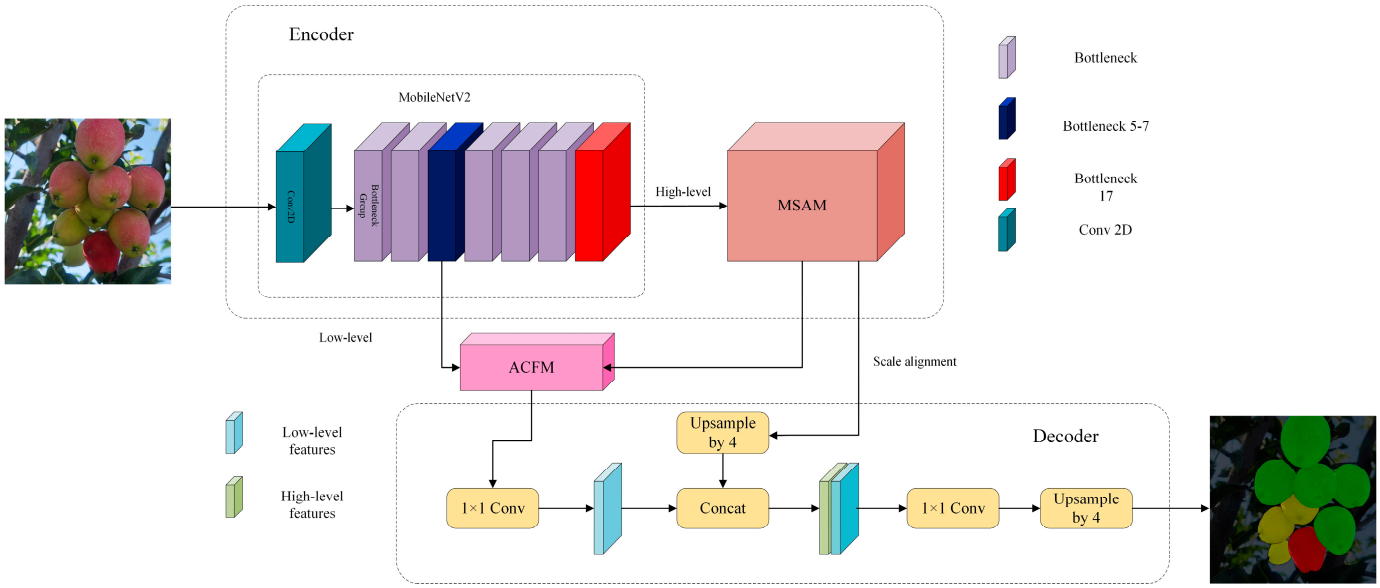


Figure 4. MA-DeepLabV3+ overall architecture.

2.2.2. MobileNetV2

For the *Jixin* fruit maturity recognition task, the selection of the backbone network must comprehensively consider the following three requirements: (1) Lightweight design: To facilitate deployment on mobile field equipment or embedded platforms, the model must possess low parameter count and computational complexity; (2) Feature representation capability: The network must effectively extract discriminative features such as color and texture at different maturity stages of *Jixin* fruit; (3) Multi-scale perception capability: The network must accommodate scale variations arising from different shooting distances and occlusion levels.

In the original DeepLabV3+ architecture, the Xception backbone is built upon depthwise separable convolution. This operation is executed in two stages: a depthwise convolution followed by a pointwise convolution. The depthwise convolution applies a single filter per input channel for efficient spatial feature extraction, which significantly reduces the model's computational burden. The subsequent pointwise convolution, which employs 1×1 convolution kernels, is then responsible for combining the outputs from the depthwise convolution across channels to create new feature representations and restore the model's expressive power. Although depth-wise separable convolutions enable Xcep-

tion to achieve superior performance at equivalent computational complexity, its deep network architecture and large parameter count fail to meet the lightweight deployment requirements of this study. To this end, the enhanced model integrates MobileNetV2 as its backbone network.

MobileNetV2 inherits the design concept of MobileNet, uses depthwise separable convolution, and innovates in inverted residual blocks and linear bottlenecks. Unlike the dimension expansion method of traditional residual structures, inverted residual blocks adopt the opposite dimensionality reduction approach: first increasing dimensions through a 1×1 convolution, then performing efficient feature learning through depthwise separable convolution, and finally reducing dimensions back through another 1×1 convolution. This design enables the model to learn richer feature representations at low computational cost, facilitating the capture of subtle color and texture variations during the *Jixin* fruit maturation process. The shortcut connection exists only when the stride equals 1, and the input and output feature map dimensions are identical, as illustrated in Figure 5. In this module, the bottleneck layer (i.e., the convolution layer with fewer channels) is designed with a linear activation function rather than using activation functions such as ReLU. This design is based on the following consideration: applying ReLU in low-dimensional spaces may cause information loss, whereas linear activation preserves the integrity of the information flow. For the *Jixin* fruit recognition task, the color gradient information related to maturity is relatively subtle, and the linear bottleneck layer helps prevent these critical features from being truncated during propagation.

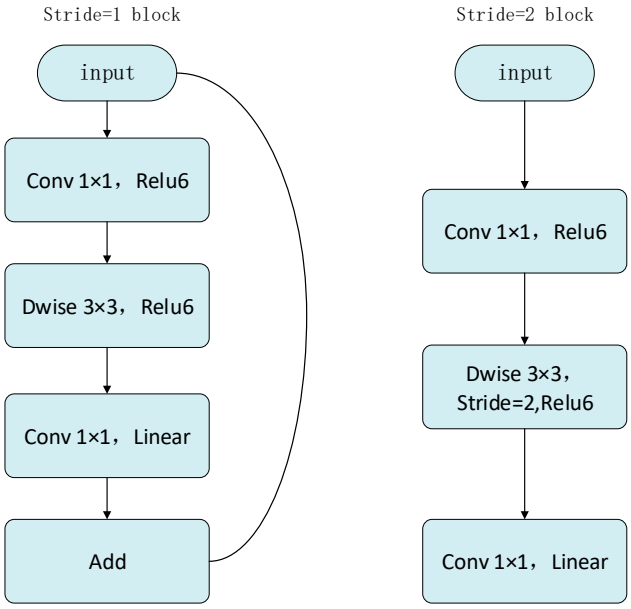


Figure 5. Structural diagram of the Inverted Residuals.

The specific architecture of MobileNetV2 begins with an initial 3×3 convolution layer for preliminary feature extraction, then performs multi-scale feature extraction through inverted residual modules composed of 17 inverted residual blocks, omitting the steps of pointwise convolution and global average pooling, effectively reducing computational cost and number of parameters. Its network structure configuration is shown in Table 1, where t represents the expansion rate of 1×1 convolution dimension increase in the Inverted Residuals structure, c is the depth of the output feature matrix, n represents the number of times the bottleneck (i.e., Inverted Residuals structure) is repeated, and s represents stride, but only represents the stride of DW convolution in the first bottleneck, and the strides of subsequent repeated bottlenecks are all equal to 1. MobileNetV2 is not just a fully

connected structure; it adopts a protective design similar to gradient explosion or gradient vanishing, maintaining the gradient flow and training stability of the network. Compared to Xception or other traditional Convolutional Neural Networks, MobileNetV2 has fewer parameters and computational cost, especially in inference tasks. It can run efficiently in low computational resource environments, providing faster speeds and lower power consumption, making it an ideal choice for deployment on mobile devices and embedded devices, meeting the lightweight requirements of this study.

Table 1. MobileNetV2 network structure configuration diagram.

Operator	t	C	n	s
Conv2d	-	32	1	2
bottleneck	1	16	1	1
bottleneck	6	24	2	2
bottleneck	6	32	3	2
bottleneck	6	64	4	2
bottleneck	6	96	3	1
bottleneck	6	160	3	2
bottleneck	6	320	1	1

2.2.3. Multi-Scale Self-Attention Module

In the original DeepLabV3+ model, ASPP is mainly used to extract multi-scale information through different atrous convolutions, enhancing the model's ability to capture features at different scales. However, when processing *Jixin* fruit maturity segmentation tasks, in natural environments, there are often complex backgrounds and small target fruits, which may cause ASPP to fail to capture local details or global context well when processing multi-scale information through different atrous convolution kernels. Especially in complex backgrounds, ASPP may lose some low-level features or high-dimensional information. Moreover, when segmenting small objects, since the ASPP module mainly relies on multi-scale atrous convolution for feature extraction, when fusing low-level and high-level features, it may not be able to effectively fuse spatial information at all levels. To solve these problems, we introduced the MSAM module to replace the ASPP of the original model. A schematic diagram of the proposed MSAM module is depicted in Figure 6. This module utilizes self-attention mechanisms [26] to capture long-range contextual semantic information and identify dependencies between objects under multi-scale conditions. By implementing mutual self-attention at different scales, MSAM can significantly improve the segmentation accuracy of small objects and details.

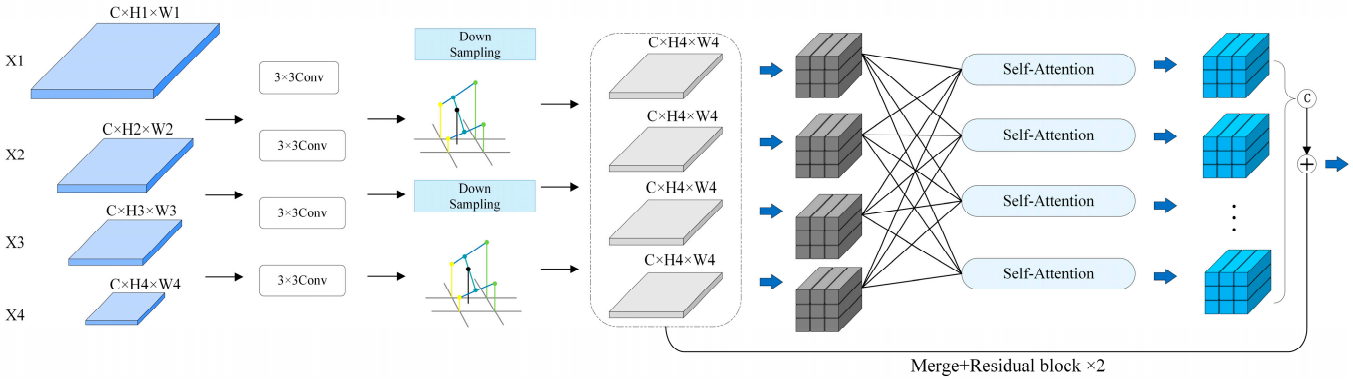


Figure 6. Schematic diagram of MSAM structure. Arrows indicate the direction of feature propagation.

First, the module received four stage output feature maps from MobileNetV2: $\{X_1 \in \mathbb{R}^{C\times H_1\times W_1}, X_2 \in \mathbb{R}^{C\times H_2\times W_2}, X_3 \in \mathbb{R}^{C\times H_3\times W_3}, X_4 \in \mathbb{R}^{C\times H_4\times W_4}\}$, as inputs. These feature maps have different spatial dimensions and numbers of channels. A 3 × 3 convolution was used for feature extraction with channels set to 128, obtaining 4 preprocessed feature maps. At the same time, all feature maps were downsampled to the same spatial dimension as X_4 to achieve spatial alignment of multi-scale features. Subsequently, a combinatorial self-attention mechanism was adopted to perform attention calculation on all possible feature map pairs. Each attention calculation first uses a linear transformation to generate weight parameter matrices W_Q, W_K, and W_V for relevant paired feature maps. Then, query, key, and value are generated for each weight, which can be expressed as follows:

$$Q = J_i W_Q \tag{1}$$

$$K = J_j W_K \tag{2}$$

$$V = J_j W_V \tag{3}$$

where J_i and J_j represent feature maps at different scales. Then, the attention feature map C_{ij} is calculated:

$$C_{ij} = Softmax\left(QK^T \cdot \alpha\right)V \tag{4}$$

where QK^T transpose represents the dot product of the query matrix and the transpose of the key matrix, α is a scaling parameter used to adjust the numerical range. The Softmax function performs normalization processing to generate attention weights, which are finally multiplied by the value matrix V to obtain the cross-scale attention feature map C_{ij}. This process produces a total of 10 different attention feature maps. In the self-attention branch, all generated attention feature maps are concatenated along the channel dimension and integrated through 1 × 1 convolution. In the auxiliary branch, the four preprocessed feature maps are directly fused and perform deep feature extraction through two consecutive residual blocks to supplement channel information. Finally, the output feature maps from both branches were fused through element-wise addition, generating an enhanced feature map rich in multi-scale global information. After the above processing, the MSAM module outputs an enhanced feature map that integrates multi-scale contextual information, which can effectively capture the long-range dependency relationships between targets of different scales and provide rich semantic information for subsequent segmentation tasks.

2.2.4. Attention and Convolution Fusion Module

When performing semantic segmentation on *Jixin* fruit, maturity assessment relies on multi-scale information, while changes in target fruit maturity are usually non-uniform, especially during the semi-ripe stage. Due to lighting conditions, part of the surface of semi-ripe fruit shows color changes consistent with ripe fruit, but the backlit portion remains unripe or semi-ripe. These characteristics lead to certain difficulties in effectively separating fruit maturity in feature maps. The feature fusion stage in the original model directly concatenates high-level features (semantic information) and low-level features (edges, textures) with only a simple 1 × 1 convolution in between, without considering inter-feature correlations. Moreover, direct stacking of low-level and high-level features may generate a semantic gap and cannot adaptively focus on key regions. To address the above issues, this study introduced the ACFM in the feature fusion stage of DeepLabV3+, with its structure shown in Figure 7. The original ACFM aims to synergistically combine the global feature modeling capability of the attention mechanism and the local feature extraction advantage of the convolution operation in 2D image space. This study extended this architecture to 3D volumetric data to achieve effective feature learning across depth,

height, and width dimensions. This module features a dual-path design. Through parallel local and global branch structures, it can retain detailed information such as edges and textures and accurately capture the boundaries of color transitions and changes in surface luster. Establish semantic associations among different regions of fruits to understand the semantic distribution of maturity across the entire fruit. Through the self-attention mechanism, it can suppress irrelevant regions, focus on targets, and, to a certain extent, eliminate the feature interference among fruits of different maturities.

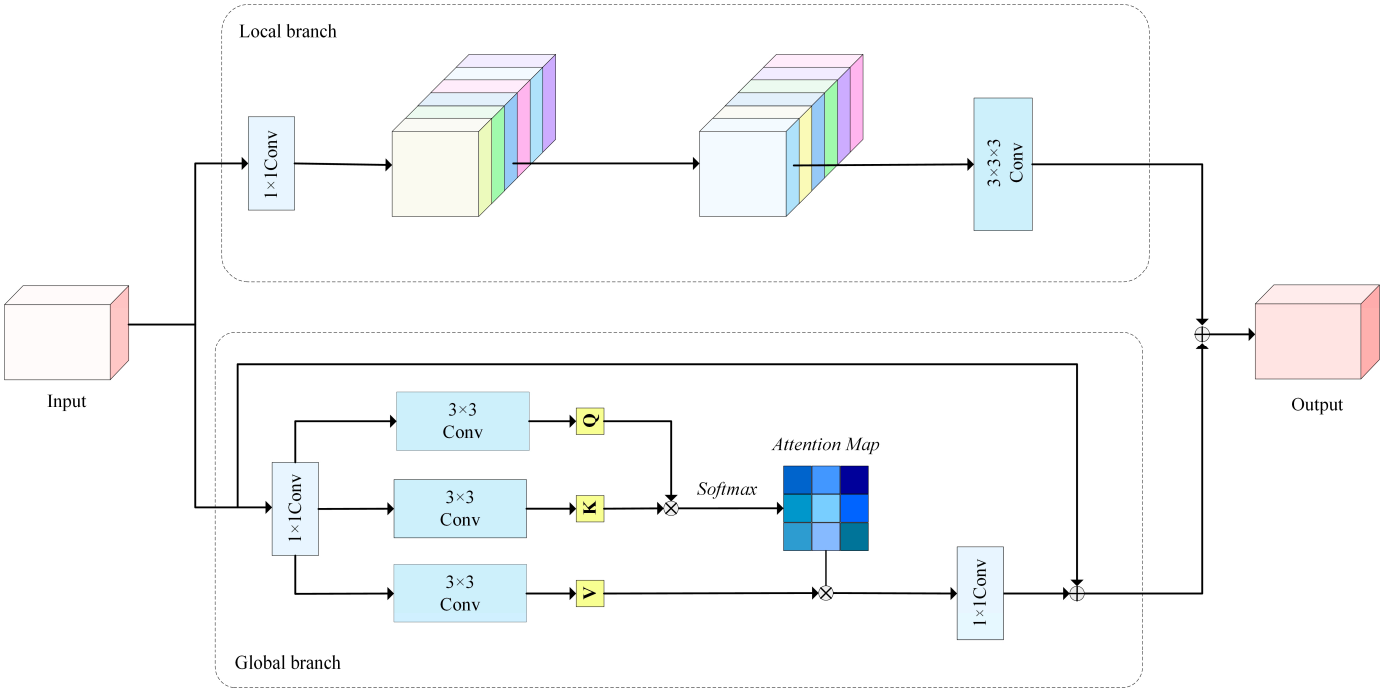


Figure 7. Schematic diagram of ACFM structure.

In the ACFM module, Attention and convolution mechanisms are fused to enhance the model's ability to model local and global features. Specifically, ACFM consists of a global branch and a local branch. The goal of the global branch is to enhance the model's ability to capture long-range dependencies through the self-attention mechanism. The steps are as follows: Firstly, given an input feature map $X \in \mathbb{R}\hat{}\{B \times C \times D \times H \times W\}$, where B, C, D, H, and W represent batch size, number of channels, depth, height, and width, respectively, and used 1×1 convolution and 3×3 depth convolution operations to generate query (Q), key (K), and value (V) tensors for feature transformation. Secondly, after reshaping Q and K, calculated the Attention Map through softmax. Its expression formula is as follows: Where alpha is a learnable temperature parameter that regulates attention distribution. The output formula of the global branch is as follows:

$$\text{Attention}(Q, K, V) = Softmax\left(\frac{QK}{\alpha}\right)V \tag{5}$$

where α is a learnable temperature parameter that regulates attention distribution. The output formula of the global branch is as follows:

$$f_{att} = W_{1\times1}\text{Attention}(Q, K, V) + X \tag{6}$$

where $W_{1\times1}$ represents 1×1 convolution, and X is the input feature. The global branch enables the network to establish feature correlations across the entire volumetric space while maintaining computational efficiency.

The local branch focuses on extracting local spatial features through enhanced 3D convolution operations combined with channel shuffle. Channel shuffle operation can improve information flow between channel groups, enhance the model's representation capability, while reducing overfitting risk. The formula for the local branch is as follows:

$$f_{conv} = W_{3\times3\times3}(CS(W_{1\times1}(Y))) \tag{7}$$

The final output of ACFM fuses global and local features through element-wise addition:

$$Output = f_{att} \oplus f_{conv} \tag{8}$$

This design enables the ACFM module to utilize both global attention and local convolution advantages, making feature fusion more intelligent and targeted, especially performing excellently in scenarios requiring fine-grained segmentation.

2.3. Evaluation Metrics

This study used the following evaluation metrics to comprehensively evaluate model performance: mean Intersection over Union (mIoU), mean Pixel Accuracy (mPA), F1-Score, Parameters, GFLOPs, Inference time (ms), frames per second (FPS), and memory footprint (MB). The calculation of the four segmentation accuracy metrics is shown in Equations (9)–(12):

$$IoU = \frac{TP_i}{TP_i + FP_i + FN_i} \tag{9}$$

$$mIoU = \frac{1}{n+1}\sum_{i=1}^{n}\frac{TP_i}{TP_i + FP_i + FN_i} \tag{10}$$

$$mPA = \frac{1}{n}\sum_{i=1}^{n}\frac{TP_i}{TP_i + FN_i} \tag{11}$$

$$F1_i = \frac{2 \times TP_i}{2 \times TP_i + FP_i + FN_i} \tag{12}$$

where TP_i represents the number of correctly predicted pixels of class i, FP_i represents the number of pixels incorrectly predicted as class i, FN_i represents the number of pixels belonging to class i but not predicted, and n is the total number of classes (including background). IoU can measure the degree of overlap between the predicted region and the ground truth labeled region, with a value range of [0, 1]; the larger the value, the more accurate the segmentation. In the *Jixin* fruit maturity segmentation task, IoU can directly reflect the model's positioning accuracy for individual fruit regions. mIoU is the average IoU of all maturity classes and is the most important comprehensive evaluation metric in semantic segmentation. In this study, mIoU can comprehensively evaluate the model's recognition ability for different maturity classes. mPA represents the average pixel accuracy of each class, reflecting the model's correct pixel classification ability for each class. Compared with mIoU, mPA focuses more on classification accuracy rather than regional completeness. F1-Score is the harmonic mean of Precision and Recall, comprehensively considering the model's precision and recall rates. On imbalanced datasets, F1-Score can better reflect the model's true performance than simple accuracy. For the *Jixin* fruit segmentation task, F1-Score can balance the two types of errors: "mistakenly judging ripe fruits as unripe" (false negatives) and "mistakenly judging unripe fruits as ripe" (false positives), ensuring that the model neither picks unripe fruits by mistake nor misses ripe fruits, thereby improving the picking efficiency and fruit quality.

2.4. Experimental Environment

The experimental host operating system is Windows 11, the central processing unit is 13th Gen Intel(R) Core(TM) i5-13600KF (3.50 GHz), the running memory is 32 GB, and the graphics card is NVIDIA GeForce RTX 4070 SUPER. The neural network was trained in the Anaconda3 virtual environment, using the PyTorch (version 1.12.1) deep learning framework, and the CUDA version used is 11.0. Specific experimental parameters are shown in Table 2.

Table 2. Model experiment parameters.

Hyperparameters	Value
Image size	512 × 512
Epoch	200
Batch Size	8
Optimizer	Sgd
Learning Rate	0.001
Weight Decay	0.0001

3. Results

3.1. Performance Analysis of the Improved Model

Figures 8 and 9 show the training and validation performance curves of the MA-DeepLabV3+ model. As can be seen from Figure 8, as the number of iterations increases, the loss values of both the training set and validation set gradually decrease and tend to stabilize, with the fastest decline rate in the range of 0–25 iterations. When the number of iterations is in the range of 150–200, the loss value changes little and basically tends to stabilize, reaching a convergent state.

At the same time, as shown in Figure 9, the mIoU on the validation set steadily increases. Although the segmentation accuracy in the early stage is relatively low, with continuous training, the model's feature extraction and target recognition capabilities have been improved, and the segmentation performance has also been optimized, indicating that the MA-DeepLabV3+ model has good generalization.

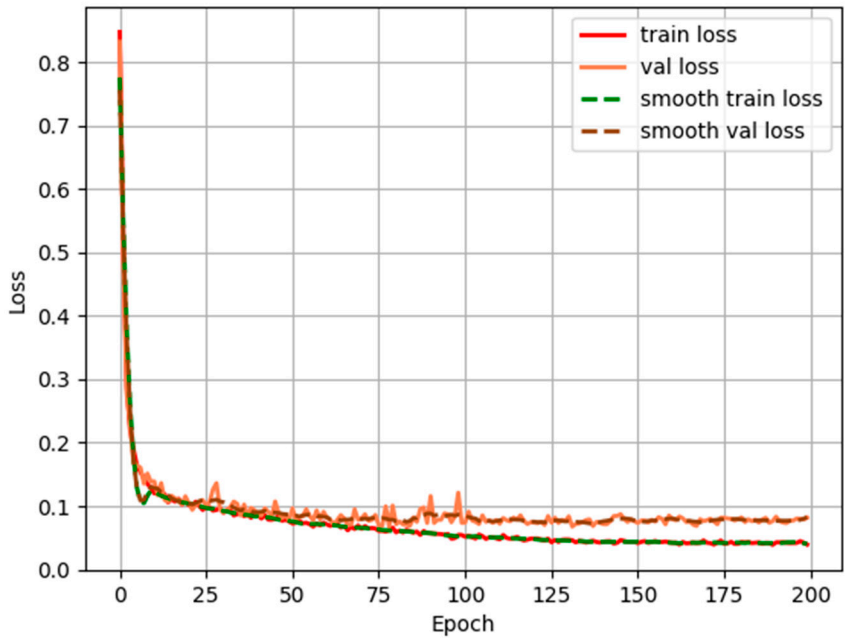


Figure 8. Loss curve of MA-DeepLabV3+ training.

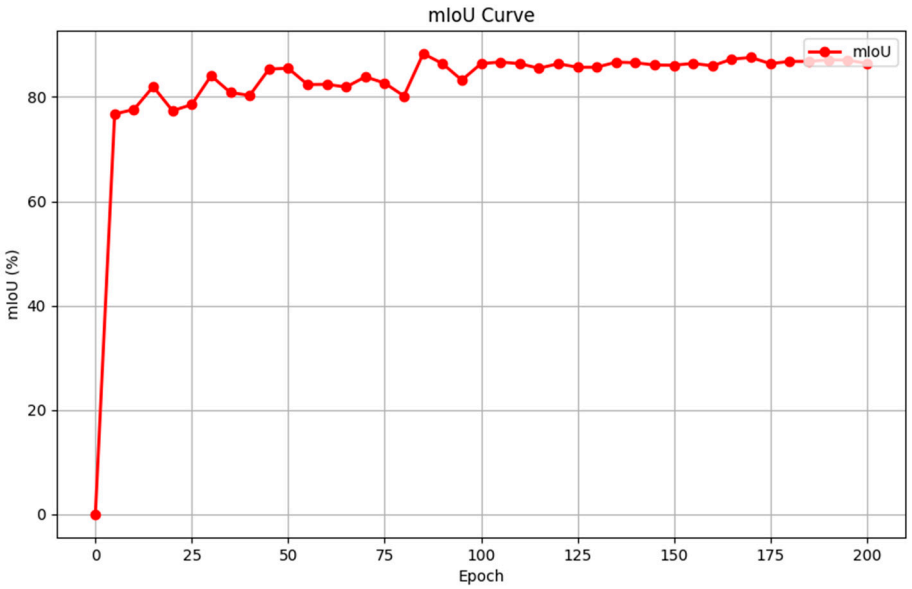


Figure 9. mIou value curve of MA-DeepLabV3+ training.

3.2. Comparison Experiments of Different Backbone Networks

When performing *Jixin* fruit maturity segmentation tasks, the backbone network's ability to extract multi-scale features such as texture, color, and morphology of fruits is key to model performance and efficiency. Since this study has high requirements for lightweight and real-time performance, robustness under complex lighting, occlusion, and background interference must also be taken into consideration. Therefore, it is necessary to conduct backbone network comparison experiments based on the DeepLabV3+ framework. To verify the effectiveness of the backbone network, this study compared the performance of different networks in terms of segmentation accuracy and lightweight metrics. The selected networks included ResNet50, VGG [27], Xception, ShuffleNetV2 [28], and MobileNetV2, and experimental control groups were constructed to compare their actual performance in *Jixin* fruit maturity segmentation. Comparison metrics include mIoU, mPA, GFLOPs, number of parameters, and F1-Score. The experimental results are shown in Table 3.

Table 3. Performance evaluation results of different backbone networks in the maturity segmentation task of *Jixin* fruit.

Backbone	mIoU (%)	mPA (%)	F1-Score (%)	Memory (MB)	Inference Time (ms)	FPS	Parameters (M)	GFLOPs (G)
ResNet50	85.87	90.84	89.76	151.19	18.52	54.0	39.64	85.32
VGG16	86.19	91.67	90.08	128.32	32.68	30.6	33.64	152.71
Xception	85.45	90.60	89.32	87.20	24.31	41.1	22.86	62.45
ShuffleNetV2	82.68	87.92	86.43	20.49	9.76	102.4	5.37	18.92
MobileNetV2	84.95	90.08	89.02	22.17	10.58	94.5	5.81	20.14

As can be seen from Table 3, although VGG16 achieved the highest mIoU of 86.19% and the highest mPA of 91.67%, its computational cost of 152.71 G far exceeds other networks, with memory usage as high as 128.32 MB, inference time of 32.68 ms, and FPS of only 30.6, which makes VGG16 face serious efficiency problems in actual *Jixin* fruit maturity segmentation and deployment, unsuitable for resource-constrained scenarios such as mobile devices and drones. ResNet50, as a deep residual network, had a parameter count of 39.64 M, memory usage of 151.19 MB, inference time of 18.52 ms, and FPS of 54.0, and achieved an mIoU of 85.87% with relatively excellent performance, but its large model size limited practical application. Xception, as the backbone network of the original

DeepLabV3+, used depthwise separable convolution and achieved an mIoU of 85.45% with 22.86 M parameters, memory usage of 87.20 MB, inference time of 24.31 ms, and FPS of 41.1, striking a good balance between accuracy and efficiency. ShuffleNetV2 was the lightest network with only 5.37 M parameters, memory usage of only 20.49 MB, inference time of 9.76 ms, and FPS as high as 102.4, demonstrating excellent real-time processing capability, but accuracy dropped to 82.68% with mPA of only 87.92%, which may not meet accuracy requirements in practical applications. In contrast, the model with MobileNetV2 as backbone maintains relatively high segmentation accuracy while achieving model lightweight. Its parameter count was 5.81 M. While maintaining lightweight, through inverted residual structure and linear bottleneck layer design, memory usage of 22.17 MB, inference time of 10.58 ms, FPS reached 94.5, and mIoU reached 85.32%, achieving a good balance between lightweight and accuracy, with inference speed improved by 56.5% compared to Xception, making it the best choice for lightweight backbone networks.

3.3. Comparison Experiments of Different Semantic Segmentation Models

To further verify the superior performance of MA-DeepLabV3+, this study compared it with the U-Net [29], PSPNet [30], SegFormer [31], and DeepLabV3+. Evaluation metrics included mIoU, mPA, F1-Score, number of parameters, and GFLOPs. Experimental results are shown in Table 4.

Table 4. Performance evaluation results of different semantic segmentation models in the maturity segmentation task of *Jixin* fruit.

Models	mIoU (%)	mPA (%)	F1-Score (%)	Memory (MB)	Inference Time (ms)	FPS	Parameters (M)	GFLOPs (G)
U-Net	88.09	93.55	88.10	94.97	38.24	26.1	24.89	451.74
PSPNet	84.10	90.03	85.32	178.24	31.47	31.7	46.71	118.43
SegFormer	75.97	84.57	77.06	14.15	7.82	127.8	3.71	13.55
DeepLabV3+	85.45	90.60	89.32	208.76	24.31	41.1	54.71	166.86
Ours	86.13	91.29	90.05	21.29	12.36	80.9	5.58	74.64

From Table 4, different semantic segmentation models show obvious differences in performance on the *Jixin* fruit maturity segmentation task. U-Net, as a classic encoder-decoder structure, had an mIoU of 88.09%, mPA of 93.55%, and F1-Score of 88.10%, demonstrating strong performance. However, its parameter count and GFLOPs reached 24.89 M and 451.74 G, respectively, with memory usage of 94.97 MB, inference time as high as 38.24 ms, and FPS of only 26.1. The huge computational overhead makes it difficult to deploy on resource-constrained devices. Although the classic semantic segmentation model PSPNet is representative, it had an excessively large parameter count of 46.71 M, memory usage of 178.24 MB, inference time of 31.47 ms, and FPS of 31.7, and the accuracy of 84.1% mIoU was also at a relatively low level. Overall performance and lightweight design cannot meet the requirements of this study. The SegFormer model adopted a lightweight design with only 3.71 M parameters and 13.55 G GFLOPs, memory usage of only 14.15 MB, inference time of 7.82 ms, and FPS as high as 127.8, possessing excellent real-time processing capability, but its segmentation accuracy was seriously insufficient, with mIoU reaching only 75.97%, making it difficult to meet actual requirements. The original DeepLabV3+ model achieved an mIoU of 85.45% and mPA of 90.60%, demonstrating good segmentation performance, but its parameter count of 54.71 M and computational cost of 166.86 G, as well as memory usage of 208.76 MB, inference time of 24.31 ms, and FPS of 41.1, limited its deployment on mobile devices.

In contrast, the MA-DeepLabV3+ model proposed in this study achieved an mIoU of 86.13%, second only to the U-Net's 88.09%. At the same time, the mPA and F1-Score

of MA-DeepLabV3+ were 91.29% and 90.05%, respectively, performing best among all comparison models. While maintaining high accuracy, its parameter count was reduced to 5.58 M, and GFLOPs were 74.64 G, memory usage of only 21.29 MB, inference time of 12.36 ms, and FPS reached 80.9, with inference speed improved by 49.2% compared to the original DeepLabV3+, meeting lightweight requirements. The above data demonstrate the effectiveness of MA-DeepLabV3+, which can achieve high-precision segmentation of *Jixin* fruit maturity while maintaining lightweight, possessing the capability for real-time deployment on mobile devices and embedded platforms, achieving a balance between accuracy, speed, and model size, and has good engineering application value.

3.4. Ablation Study

To verify the feasibility of MA-DeepLabV3+, this study conducted ablation experiments to evaluate the effect of each individual component. Taking DeepLabV3+ as the baseline model, improvements were introduced step by step. The experimental settings are as follows: the first group used the original model DeepLabV3+; the second group replaced the original model's backbone with MobileNetV2, with the rest unchanged; the third group replaced the ASPP part with the MASM module, and the backbone remained MobileNetV2, and the following groups all used this backbone; the fourth group only introduced the ACFM module in the feature fusion stage; the fifth group introduced both modules. Evaluation metrics were mIoU, mPA, F1-Score, number of parameters, and GFLOPs. The experimental results are summarized in Table 5.

Table 5. Comparison of ablation study results.

Model Configuration	mIoU (%)	mPA (%)	F1-Score (%)	Parameters (M)	GFLOPs (G)
(Baseline)	85.45	90.60	89.32	54.71	166.86
Baseline + MobileNetV2	84.95	90.40	89.02	5.81	52.88
MSAM-Only	82.95	88.81	87.24	5.11	59.03
ACFM-Only	85.70	90.94	89.66	6.28	68.49
Proposed	86.13	91.29	90.05	5.58	74.64

From Table 5, after replacing the backbone with MobileNetV2 in the second group of experiments, the parameter count of the DeepLabV3+ model decreased from 54.709 M to 5.814 M, and GFLOPs decreased from 166.858 G to 52.884 G. However, the performance loss of the model is almost negligible, with mIoU, mPA, and F1-Score of 84.95%, 90.40%, and 89.02%, respectively, proving that MobileNetV2 is a reasonable backbone network choice.

Compared with the DeepLabV3+ model after replacing the backbone with MobileNetV2, the third group of experiments with MSAM-Only further reduced the parameter count to 5.112 M, but GFLOPs slightly increased (59.025 G). It is worth noting that mIoU dropped to 82.95%, a decrease of 2.0 percentage points compared to the basic MobileNetV2 model. This phenomenon indicates that the MSAM module, through multi-branch and multi-scale feature extraction and a self-attention mechanism, although it can enhance the multi-scale expression capability of features, may lead to slight performance degradation when used alone due to a lack of an effective feature fusion mechanism. Moreover, the lightweight design of MSAM proves its advantage in maintaining model efficiency, but it needs to be combined with appropriate feature fusion strategies to fully play its role. The fourth group of experiments evaluated the independent contribution of the ACFM module. This configuration added the ACFM module for feature fusion enhancement on the basis of MobileNetV2, with a parameter count of 6.28 M and GFLOPs of 68.485 G. Importantly, mIoU increased to 85.70%, an improvement of 0.75 percentage points compared to the

second group of experiments, even exceeding the original baseline model's 85.45%. At the same time, mPA reached 90.94%, and F1-Score reached 89.66%. These results prove that the ACFM module can effectively enhance feature fusion between the encoder and decoder through an adaptive channel attention mechanism, improve the model's ability to distinguish *Jixin* fruits of different maturities, and significantly improve segmentation accuracy with limited parameter increase, demonstrating good cost-effectiveness.

The fifth group of experiments was the complete MA-DeepLabV3+ model proposed in this paper, which integrated all three improved components: MobileNetV2, MSAM, and ACFM. MA-DeepLabV3+ achieved 5.581 M parameters and 74.635 GFLOPs, representing reductions of 89.8% and 55.3%, respectively, compared to the baseline. The mIoU reached 86.13%, the highest value among all configurations, with mPA and F1-Score of 91.29% and 90.05% respectively, indicating that the model achieves a good balance between precision and recall. The above results show that there is an obvious synergistic effect between the MSAM and ACFM modules. MSAM enhances the feature expression capability of the encoder through multi-scale feature extraction, while ACFM optimizes the feature fusion process of the decoder through adaptive channel attention. When used together, they not only compensate for the performance degradation when MSAM is used alone but also achieve better results than when ACFM is used alone, while maintaining low model complexity.

The ablation experiments systematically verified the effectiveness of each improvement measure proposed in this paper. The replacement of the backbone network significantly reduces model parameters and computational cost, meeting the lightweight requirements. At the same time, the multi-scale feature extraction of the MSAM module and the adaptive feature fusion mechanism of the ACFM module effectively compensate for the accuracy loss caused by lightweight, indicating that the improved architecture design in this paper can effectively integrate the advantages of multiple modules. The final MA-DeepLabV3+ shows significant advantages in the *Jixin* fruit maturity segmentation task.

3.5. Segmentation Performance Analysis of Different Maturity Categories

To deeply evaluate the recognition ability of each model on fruits of different maturities, this section separately analyzed the IoU metric for three categories: unripe, semi-ripe, and ripe, and revealed the confusion patterns between categories through confusion matrices.

3.5.1. IoU Comparison Analysis of Each Category

To comprehensively evaluate the segmentation performance of each model at different fruit maturity stages, this study conducted an IoU [32] metric analysis for immature, semi-mature, and mature categories separately. Table 6 shows the segmentation performance of five models across different maturity categories.

Table 6. Per-class IoU results of semantic segmentation models for *Jixin* fruit maturity.

Models	Immature IoU (%)	Semi-Mature IoU (%)	Mature IoU (%)
U-Net	91	79	77
PSPNet	88	76	75
SegFormer	75	63	71
DeepLabV3+	92	79	77
Ours	93	81	79

Unripe fruits usually have characteristics similar to background color, but their edge contours are relatively clear, with relatively regular shape features. In the segmentation task of unripe fruits, all models showed high recognition accuracy. Experimental results show

that the traditional U-Net model reached 91%, demonstrating good baseline performance. It is worth noting that PSPNet obtained 88% IoU, while the Transformer-based SegFormer only reached 75%, performing relatively weak in this category. MA-DeepLabV3+ achieved the best 93% IoU in this category, an improvement of 1 percentage point compared to the original model. The excellent performance in this category benefits from its enhanced feature extraction capability and multi-scale information fusion mechanism, which can better capture the subtle texture features and edge information of unripe fruits.

Fruits in the semi-ripe stage show color gradation characteristics, with blurred boundaries and reduced contrast with the background, which brings challenges to the segmentation task. The performance of all models decreased to varying degrees in this category. The IoU of PSPNet was 76%, with a relatively large drop. SegFormer performed worst in this category, obtaining only 63% IoU, a decrease of 12 percentage points compared to the unripe category. MA-DeepLabV3+ maintains a leading advantage. Through introducing the attention mechanism and optimizing the feature fusion strategy, it obtained 81% IoU, a decrease of 12 percentage points compared to the unripe category. DeepLabV3+ and U-Net reached 79%, with stable performance. It can be seen that MA-DeepLabV3+ can more effectively handle such complex visual features, maintaining high segmentation accuracy in blurred boundary regions.

Although ripe fruits have bright colors and high contrast with the background, they are often accompanied by problems such as irregular shapes, surface reflections, and occlusion overlaps. These factors also lead to performance degradation of all models in this category. In this type of task, DeepLabV3+ and U-Net both obtained 77% IoU, with a gap of 2 percentage points from MA-DeepLabV3+. The performance of PSPNet further declined to 75%, performing worst among the three categories. SegFormer showed some recovery in the ripe category, reached 71%, but it is still significantly lower than other models. MA-DeepLabV3+ consistently maintained optimal performance with an IoU of 79%, indicating that MA-DeepLabV3+ can better handle segmentation tasks in complex scenarios through improving decoder structure and multi-level feature fusion.

3.5.2. Confusion Matrix Analysis

Figure 10 presents the normalized confusion matrices of the original DeepLabV3+ and our MA-DeepLabV3+ on the test set, providing a detailed view of the inter-class confusions. The diagonal and off-diagonal elements respectively denote the per-class pixel accuracy and the misclassification rates between categories.

It can be seen that the original model, DeepLabV3+, shows good classification ability overall, with high diagonal values, indicating that most samples are correctly classified. However, there is a high confusion rate between semi-ripe and unripe fruits, which may be due to the similarity in color and texture features between the two. In addition, there is still a small amount of misclassification between the background and each category, indicating that there is still room for optimization in feature extraction and boundary judgment of the model. In contrast, MA-DeepLabV3+ achieves significant performance improvement in the most challenging semi-ripe category by introducing multi-scale attention mechanism and adaptive context fusion module, increasing the accuracy of the semi-ripe category from 93.6% to 96.5%, an improvement of 2.9 percentage points, and reducing the confusion rate between semi-ripe and unripe categories from 4.3% to 1.7%, a decrease of 2.6 percentage points. Although there is a slight decline in unripe and ripe categories, the overall model performance is more balanced, especially showing excellent performance in solving inter-category confusion. The confusion matrix of MA-DeepLabV3+ shows a clearer diagonal structure, proving the effectiveness of the new modules.

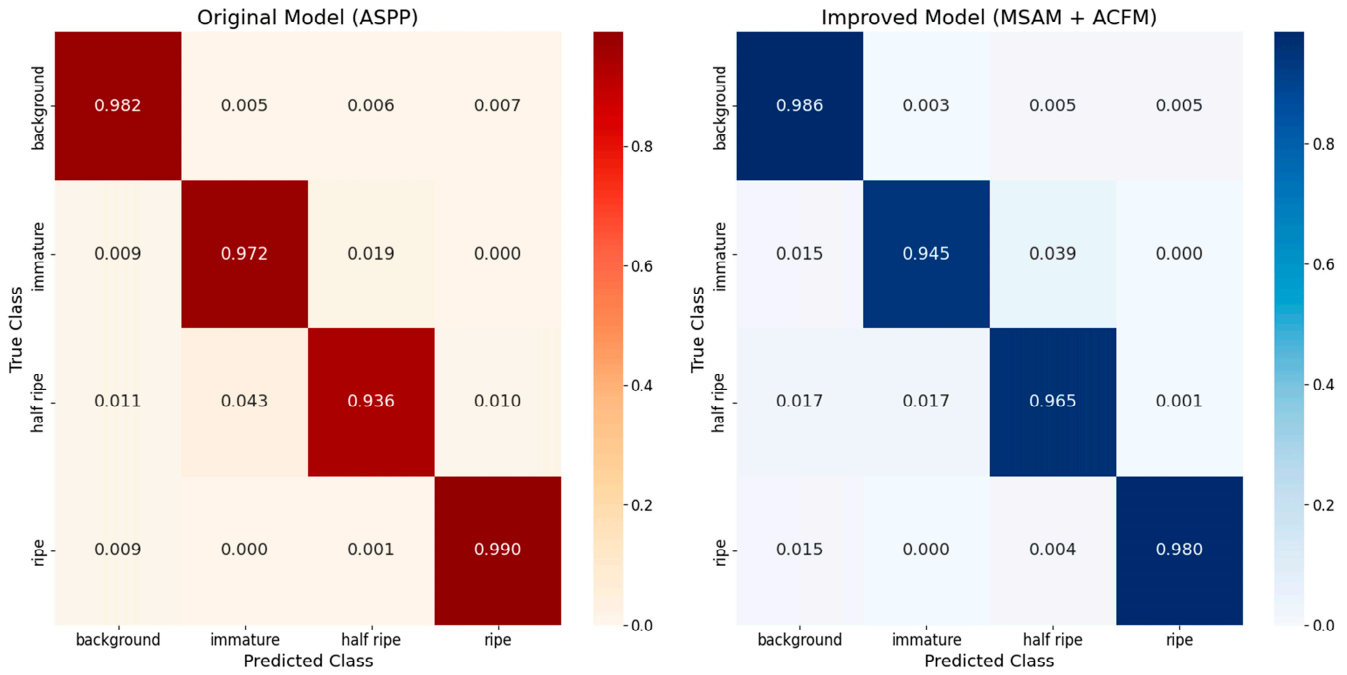


Figure 10. Normalized confusion matrix of DeepLabV3+ and MA-DeepLabV3+ on the test set.

3.5.3. Segmentation Result Visualization Analysis

To more intuitively evaluate the actual performance of different models on the *Jixin* fruit maturity segmentation task, this section conducts a visual comparison analysis of the segmentation results of typical samples. Representative images containing *Jixin* fruits of different maturity categories were selected to show the segmentation effects of each model. Through visual comparison, the differences between different models in boundary accuracy, small target detection, and complex scene processing can be clearly observed. Figure 9 shows the comparison of the segmentation results of each model.

Figure 11 visualizes the segmentation results, highlighting the differences between models. DeepLabV3+ maintains high segmentation accuracy in mixed scenarios and can accurately distinguish different maturity categories, but confusion still occurs in distinguishing semi-ripe and ripe fruits in Image 1, and blurred edge segmentation also appears in Image 3. U-Net can handle multi-category mixed scenarios well, but there are deficiencies in fine boundary segmentation, and like DeepLabV3+, there is confusion between different maturity categories, and the segmentation continuity of densely distributed fruits is not ideal. PSPNet's performance declines in mixed scenarios, especially in Images 3 and 4, where misclassification occurs due to leaf occlusion or uneven maturity distribution. SegFormer performs poorly in complex mixed scenarios, with serious category confusion and obvious fragmentation of segmentation results, performing poorly in the segmentation results of all images. In contrast, MA-DeepLabV3+ demonstrates excellent comprehensive performance in complex mixed scenarios. It can accurately identify and segment fruits of three maturity levels with clear category distinction and precise boundaries. Even when fruits are densely packed and mutually occluded, they can maintain high segmentation quality. Although some fragmentation phenomenon of some segmentation results occurs in Image 1, its lightweight design and high inference speed can balance some deficiencies in performance, enabling it to meet real-time processing requirements, fully verifying the superiority of this method in practical applications.

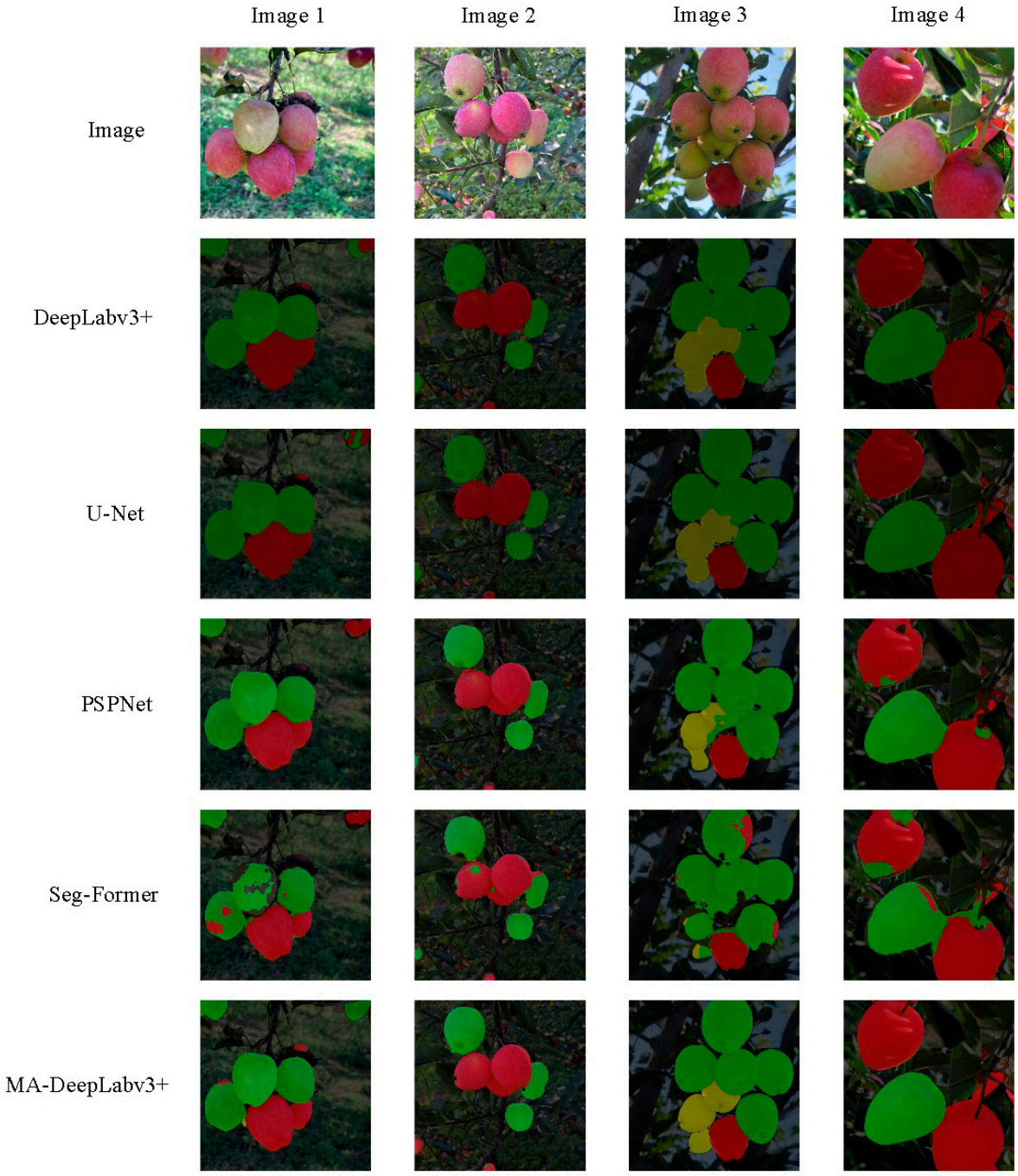


Figure 11. Visual comparison of segmentation results from different semantic segmentation models.

4. Discussion

4.1. Comparative Analysis with Existing Methods

The proposed MA-DeepLabV3+ demonstrates significant advantages over existing fruit segmentation methods. Compared with the Cherry-Net proposed by Cui et al. [33], MA-DeepLabV3+ enhances multi-scale feature perception capability through the MSAM module, exhibiting stronger robustness in color gradient and occlusion scenarios. Compared with Mask R-CNN employed by Zu et al. [34], the proposed model has a parameter count of only 5.58 M, substantially lower than the 44 M+ of Mask R-CNN, making it more suitable for edge device deployment. Furthermore, this study adopts a self-attention mechanism to replace the fixed dilation rate dilated convolutions in traditional ASPP, enabling

dynamic modeling of correlations between features at different scales and demonstrating better adaptability in scenarios with significant scale variations.

4.2. Analysis of Model Performance Differences

Based on the experimental results presented in Tables 3 and 4, the performance differences among models can be explained from the perspective of network architectural characteristics. VGG16 achieved the highest mIoU (86.19%) due to its strong feature representation capability from deep architecture, but its substantial computational requirements limit real-time applications. Although ShuffleNetV2 achieved the fastest inference speed (102.46 FPS), its channel shuffle operation sacrifices feature spatial continuity, resulting in lower mIoU (83.24%). MobileNetV2 preserves rich features in high-dimensional spaces through inverted residual structures, while the linear bottleneck layer prevents the loss of subtle color information, achieving an optimal accuracy-efficiency trade-off. SegFormer achieved an mIoU of only 73.52%, indicating that pure Transformer architectures lack the inductive bias advantages of CNNs on small-scale datasets. MA-DeepLabV3+ achieved the best mPA (91.29%) and F1-Score (90.05%), attributed to the adaptive fusion of global and local features by the ACFM module.

4.3. Practical Application and Deployment Analysis

The lightweight design of MA-DeepLabV3+ enables deployment across various edge computing platforms. Real-time inference can be achieved using the NVIDIA Jetson Nano developer kit (NVIDIA Corporation, Santa Clara, CA, USA; approximately 1000 CNY), reducing hardware costs by approximately 75% compared with the Jetson Xavier NX (NVIDIA Corporation, Santa Clara, CA, USA; approximately 4000 CNY) required for U-Net deployment. Regarding application scenarios, this model is suitable for natural lighting environments (8:00–17:00), with a recommended camera-to-fruit distance of 30–100 cm. Under 30 FPS conditions, single-tree fruit detection can be completed within 2–3 s. For strong backlighting scenarios, supplementary lighting equipment is recommended.

4.4. Research Limitations

This study has the following limitations: The dataset was collected from a single production area in Yongji County, Jilin Province, and the model's generalization capability in other cultivation regions requires further validation. Under strong backlighting or deep shadow scenarios, the confusion rate between semi-ripe and ripe fruits increases. The study focused solely on *Jixin* fruit, and the model's transferability to other fruit species has not been explored. All experiments were conducted offline, and the model's performance in actual field environments when coordinated with robotic arms requires verification.

4.5. Future Research Directions

To address the aforementioned limitations, future research will proceed in the following directions: expanding the dataset to cover more production regions and climatic conditions, and introducing domain adaptation techniques to enhance cross-domain generalization capability; investigating illumination-adaptive strategies to improve robustness under extreme lighting conditions; exploring transfer learning applications of the model to other fruits such as apples and tomatoes; and conducting integration and deployment studies of the model on actual harvesting robot platforms to verify performance in real operational environments.

5. Conclusions

This paper introduces MA-DeepLabV3+, a lightweight semantic segmentation model for assessing *Jixin* fruit maturity, developed to overcome the limitations of low recogni-

tion efficiency and high labor costs in manual picking. The proposed model employs MobileNetV2 as a lightweight backbone network, designs a Multi-Scale Attention Module (MSAM) to replace the traditional ASPP structure, and introduces an Attention and Convolution Fusion Module (ACFM) to enhance boundary segmentation capability. Experimental results demonstrate that MA-DeepLabV3+ achieves an mIoU of 86.13%, an mPA of 91.29%, and an F1-Score of 90.05% on the self-constructed *Jixin* fruit dataset. Compared with the original DeepLabV3+ model, the parameter count is reduced by 89.8%, computational complexity is decreased by 55.3%, inference speed reaches 80.91 FPS, and model size is only 21 MB, satisfying the real-time requirements of intelligent harvesting systems. Compared with classic semantic segmentation models, U-Net, PSPNet, SegFormer, and the original DeepLabV3+, MA-DeepLabV3+ still maintains high segmentation performance while significantly reducing parameter count and computational cost. Through comparative analysis of images of *Jixin* fruits at different maturities, the advantages of the model in handling complex features such as maturity color gradation and boundary blurring are further confirmed. This study validates that under lightweight constraints, precision loss can be effectively compensated through multi-scale attention mechanisms and adaptive feature fusion, providing a reference paradigm for semantic segmentation model design in resource-constrained scenarios. The lightweight characteristics of the model enable deployment on mobile devices and embedded platforms, significantly reducing the hardware costs of intelligent harvesting systems and providing a feasible technical solution for the intelligent upgrading of specialty fruit industries.

Author Contributions: Conceptualization, L.D. and J.X.; methodology, L.D., D.F. and J.X.; software, Q.H. and J.X.; validation, D.F. and Q.H.; formal analysis, Q.H.; investigation, Q.H., D.F. and J.X.; data curation, D.F.; writing—original draft preparation, J.X.; writing—review and editing, L.D. and J.X.; visualization, Q.H. and D.F.; supervision, L.D.; project administration, L.D.; funding acquisition, L.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation Project of Jilin Provincial Department of Science and Technology (20260102264JC).

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 conflicts of interest.

References

1. Lowenberg-DeBoer, J.; Huang, I.Y.; Grigoriadis, V.; Blackmore, S. Economics of robots and automation in field crop production. *Precis. Agric.* **2020**, *21*, 278–299. [CrossRef]
2. Vrochidou, E.; Tsakalidou, V.N.; Kalathas, I.; Gkrimpizis, T.; Pachidis, T.; Kaburlasos, V.G. An Overview of End Effectors in Agricultural Robotic Harvesting Systems. *Agriculture* **2022**, *12*, 1240. [CrossRef]
3. Zhang, Y.; Wu, L. Classification of Fruits Using Computer Vision and a Multiclass Support Vector Machine. *Sensors* **2012**, *12*, 12489–12505. [CrossRef]
4. Wu, G.; Li, B.; Zhu, Q.; Huang, M.; Guo, Y. Using color and 3D geometry features to segment fruit point cloud and improve fruit recognition accuracy. *Comput. Electron. Agric.* **2020**, *174*, 105475. [CrossRef]
5. Xiao, F.; Wang, H.; Li, Y.; Cao, Y.; Lv, X.; Xu, G. Object Detection and Recognition Techniques Based on Digital Image Processing and Traditional Machine Learning for Fruit and Vegetable Harvesting Robots: An Overview and Review. *Agronomy* **2023**, *13*, 639. [CrossRef]
6. Payne, A.; Walsh, K.; Subedi, P.; Jarvis, D. Estimating mango crop yield using image analysis using fruit at 'stone hardening' stage and night time imaging. *Comput. Electron. Agric.* **2014**, *100*, 160–167. [CrossRef]

7. Tian, Y.N.; Yang, G.D.; Wang, Z.; Wang, H.; Li, E.; Liang, Z.Z. Apple detection during different growth stages in orchards using the improved YOLO-V3 model. *Comput. Electron. Agric.* **2019**, *157*, 417–426. [CrossRef]
8. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
9. Kamilaris, A.; Prenafeta-Boldú, F.X. Deep learning in agriculture: A survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [CrossRef]
10. Jiang, H.; Zhang, C.; Qiao, Y.; Zhang, Z.; Zhang, W.; Song, C. CNN feature based graph convolutional network for weed and crop recognition in smart farming. *Comput. Electron. Agric.* **2020**, *174*, 105450. [CrossRef]
11. Xie, Z.; Ke, Z.; Chen, K.; Wang, Y.; Tang, Y.; Wang, W. A lightweight deep learning semantic segmentation model for optical-image-based post-harvest fruit ripeness analysis of sugar apples (*Annona squamosa*). *Agriculture* **2024**, *14*, 591.
12. Nuanmeesri, S. Enhanced hybrid attention deep learning for avocado ripeness classification on resource constrained devices. *Sci. Rep.* **2025**, *15*, 3719. [CrossRef]
13. Chen, X.; Dong, G.; Fan, X.; Xu, Y.; Liu, T.; Zhou, J.; Jiang, H. Fruit Stalk Recognition and Picking Point Localization of New Plums Based on Improved DeepLabv3+. *Agriculture* **2024**, *14*, 2120. [CrossRef]
14. Hou, X.; Chen, P.; Gu, H. LM-DeeplabV3+: A Lightweight Image Segmentation Algorithm Based on Multi-Scale Feature Interaction. *Appl. Sci.* **2024**, *14*, 1558. [CrossRef]
15. Liu, H.; Chen, Y.; Wang, R.; Li, M.; Li, Z. MFA-Deeplabv3+: An improved lightweight semantic segmentation algorithm based on Deeplabv3+. *Complex Intell. Syst.* **2025**, *11*, 424. [CrossRef]
16. Cao, X.; Zhong, P.; Huang, Y.; Huang, M.; Huang, Z.; Zou, T.; Xing, H. Research on Lightweight Algorithm Model for Precise Recognition and Detection of Outdoor Strawberries Based on Improved YOLOv5n. *Agriculture* **2025**, *15*, 90. [CrossRef]
17. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11534–11542.
18. Chen, J.; Zhan, Y.; Xu, Y.; Pan, X. FAFNet: Fully aligned fusion network for RGBD semantic segmentation based on hierarchical semantic flows. *IET Image Process.* **2023**, *17*, 32–41.
19. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
20. He, K.M.; Zhang, X.Y.; Ren, S.Q.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
21. Li, L.; Yi, J.; Fan, H.; Lin, H. A Lightweight Semantic Segmentation Network Based on Self-Attention Mechanism and State Space Model for Efficient Urban Scene Segmentation. *IEEE Trans. Geosci. Remote Sens.* **2025**, *63*, 4703215. [CrossRef]
22. Chen, Z.; Lu, S. Caf-yolo: A robust framework for multi-scale lesion detection in biomedical imagery. In Proceedings of the ICASSP 2025—2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Hyderabad, India, 6–11 April 2025; pp. 1–5.
23. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef]
24. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1800–1807.
25. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
26. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
27. Swapna, M.; Sharma, Y.K.; Prasadh, B. Cnn architectures: Alex net, le net, vgg, google net, res net. *Int. J. Recent Technol. Eng.* **2020**, *8*, 953–960. [CrossRef]
28. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 116–131.
29. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015, Munich, Germany, 5–9 October 2015; pp. 234–241.
30. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
31. Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J.M.; Luo, P. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 12077–12090.

32. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
33. Cui, J.; Zhang, L.; Gao, L.; Bai, C.; Yang, L. Cherry-Net: Real-time segmentation algorithm of cherry maturity based on improved PIDNet. *Front. Plant Sci.* **2025**, *16*, 1607205. [CrossRef] [PubMed]
34. Zu, L.L.; Zhao, Y.P.; Liu, J.Q.; Su, F.; Zhang, Y.; Liu, P.Z. Detection and Segmentation of Mature Green Tomatoes Based on Mask R-CNN with Automatic Image Acquisition Approach. *Sensors* **2021**, *21*, 7842. [CrossRef] [PubMed]

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.