Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.064635

ARTICLE

SFC_DeepLabv3+: A Lightweight Grape Image Segmentation Method Based on Content-Guided Attention Fusion

Yuchao Xia and Jing Qiu*

College of Mathematics and Computer Science, Zhejiang A&F University, Hangzhou, 311300, China

*Corresponding Author: Jing Qiu. Email: qiujing@zafu.edu.cn

Received: 20 February 2025; Accepted: 30 April 2025; Published: 03 July 2025

ABSTRACT: In recent years, fungal diseases affecting grape crops have attracted significant attention. Currently, the assessment of black rot severity mainly depends on the ratio of lesion area to leaf surface area. However, effectively and accurately segmenting leaf lesions presents considerable challenges. Existing grape leaf lesion segmentation models have several limitations, such as a large number of parameters, long training durations, and limited precision in extracting small lesions and boundary details. To address these issues, we propose an enhanced DeepLabv3+ model incorporating **S**trip Pooling, Content-Guided **F**usion, and **C**onvolutional Block Attention Module (SFC_DeepLabv3+), an enhanced lesion segmentation method based on DeepLabv3+. This approach uses the lightweight MobileNetv2 backbone to replace the original Xception, incorporates a lightweight convolutional block attention module, and introduces a content-guided feature fusion module to improve the detection accuracy of small lesions and blurred boundaries. Experimental results show that the enhanced model achieves a mean Intersection over Union (mIoU) of 90.98%, a mean Pixel Accuracy (mPA) of 94.33%, and a precision of 95.84%. This represents relative gains of 2.22%, 1.78%, and 0.89% respectively compared to the original model. Additionally, its complexity is significantly reduced without sacrificing performance,the parameter count is reduced to 6.27 M, a decrease of 88.5% compared to the original model, floating point of operations (GFLOPs) drops from 83.62 to 29.00 G, a reduction of 65.1%. Additionally, Frames Per Second (FPS) increases from 63.7 to 74.3 FPS, marking an improvement of 16.7%. Compared to other models, the improved architecture shows faster convergence and superior segmentation accuracy, making it highly suitable for applications in resource-constrained environments.

KEYWORDS: Grape leaf; leaf segmentation; lightweight; feature fusion; DeepLabv3+

1 Introduction

Grapes are one of the most widely consumed fruits globally and serve as the primary raw material for wine production, making them an economically important crop cultivated extensively worldwide [1]. However, grape leaves are highly susceptible to various diseases caused by fungi, viruses, and bacteria, influenced by environmental and climatic factors [2]. Black rot, a major fungal disease, can lead to substantial damage to fruits and leaves if not promptly identified and controlled, thus impacting grape yield and quality and causing economic losses for growers. Currently, the severity of black rot is primarily assessed based on the proportion of lesion area on the leaf surface. Deep learning technology offers the potential for accurate lesion segmentation on leaves, enabling rapid disease assessment and playing a critical role in the effective health management of vineyards.

Traditional disease detection methods primarily rely on visual inspection by agricultural experts, which is inefficient on a large scale, requiring substantial labor and being heavily dependent on the subjective experience of the evaluator [3]. With advances in image processing and computer technology, image segmentation methods have evolved through three major stages: from classical segmentation to machine learning-based segmentation, and finally to deep learning-based segmentation. Classical segmentation techniques include thresholding, region-based, edge detection, and clustering methods. Babu et al. utilized the Otsu multi-threshold method combined with the chimp optimization algorithm for segmenting disease images of tomato leaves, demonstrating robust generalization [4]. Jaskaran et al. applied a region-based KNN classifier with texture feature analysis for plant disease detection [5]. Additionally, Sudha et al. proposed a machine learning model for detecting anthracnose in cashew leaves, combining random forest with the K-Means algorithm to achieve effective leaf segmentation [6].

Deep learning provides a more effective solution for segmentation tasks to address the high image quality requirements of classical segmentation methods and the complexity of machine learning approaches. In recent years, the rapid advancement of deep learning technology has significantly driven progress in image segmentation. Long et al. introduced the fully convolutional network (FCN), which enabled the application of deep learning in semantic segmentation by replacing fully connected layers [7]. Subsequent models like SegNet [8] and DeconvNet [9] refined the FCN structure through an encoder-decoder architecture, enhancing detail recovery. To improve multi-scale object recognition, PSPNet introduced global pyramid pooling for multi-scale information integration [10]. Deepak Kumar et al. advanced the field with a multi-stage model (PSGIC) based on PSPNet and fuzzy rules, enabling applications in multi-stage scenarios [11]. Mask R-CNN [12] added a parallel branch for generating segmentation masks on top of Faster R-CNN [13], achieving both object detection and pixel-level segmentation. U-Net captured contextual information by combining encoding and decoding paths, refining the FCN architecture, and improving segmentation accuracy with a skip connection mechanism [14]. Yi et al. [15] further enhanced U-Net by integrating VGG as the backbone with dual attention modules for channel and spatial dimensions, enabling segmentation of disease-affected areas on grape leaves. BiSeNet [16] balances speed and accuracy with a dual-path design but compromises precision for small objects. OCRNet [17] enhances segmentation via object-contextual relations, yet its heavy backbone reduces efficiency. Lightweight models like ENet [18] and Fast-SCNN [19] prioritize speed, struggling with fine details and edges due to shallow architectures. In contrast, these methods often fail to address the specific challenge of segmenting small, irregularly shaped targets like grape leaf lesions, where both efficiency and precision are paramount.

To enhance the ability to capture image details, the DeepLab series introduced Atrous Spatial Pyramid Pooling to expand the receptive field, progressively optimizing multi-scale feature extraction capabilities from DeepLab [20] to DeepLabv3+ [21]. DeepLabv3+ demonstrates high accuracy in semantic segmentation and has been widely applied to complex image segmentation tasks [22]. However, DeepLabv3+ and related models face significant limitations in lightweight design, detail recovery, and edge detection, particularly when addressing the challenges of segmenting small targets prevalent in current research. First, its backbone network Xception, exhibits a high parameter count and computational complexity, impeding its deployment in large-scale, real-time detection tasks, such as vineyard monitoring [23]. Second, Spatial reduction in feature extraction loses fine details vital for small lesion detection, and poor edge restoration affects boundary accuracy for irregular or blurred lesions. These deficiencies are not unique to DeepLabv3+; for instance, U-Net and FCN similarly struggle with spatial detail loss, while PSPNet sacrifices local precision for global context.

We proposes an improved segmentation model, SFC_DeepLabv3+[1], to address the limitations of DeepLabv3+ in terms of parameters, training time, and detail extraction. Our model effectively enhances segmentation accuracy for small lesions and edge features while reducing parameters and computational complexity, supporting efficient agricultural disease management and decision-making. The main contributions of our work are as follows.

1. We proposed a lightweight segmentation model SFC_DeepLabv3+. It utilizes MobileNetV2 as the backbone, integrates Strip Pooling, CBAM and CGAFusion. It effectively detects blurred edges and small lesions in leaf disease segmentation, achieving an mIoU of 90.98%, mPA of 94.33%, and precision of 95.84%, improvements of 2.22%, 1.78%, and 0.89% over baseline. The model reduces parameters to 6.27 M (88.5% decrease), GFLOPs to 29.00 G (65.1% reduction), and increases processing speed to 74.3 FPS (16.7% improvement), making it ideal for resource-constrained agricultural applications.
2. Through ablation experiments and heatmap analysis, we demonstrated the individual and combined contributions of these modules to segmentation performance. By integrating all enhancements, the model achieved improvements in mIoU, mPA, and Precision, highlighting the effectiveness of multi-scale feature fusion and dynamic attention mechanisms. Compared to segmentation models on public datasets, including DeepLabv3+, Unet and HRNet, SFC_DeepLabv3+ exhibited superior performance.
3. We have proposed the dataset used in this study along with its annotation files, trained model weights, and complete source code. This not only ensures reproducibility of the research but also provides valuable support for further research and practical applications in the field of agricultural disease segmentation.

In the following sections, Section 2 provides a detailed description of the dataset and our SFC_DeepLabv3+. Section 3 discusses the experimental setup and the results. Section 4 analyses the results of the experiment. Finally, Section 5 presents the conclusions of this study.

2 Material and Methods

2.1 Dataset

This study utilized the publicly available PlantVillage dataset [24], which contains 61,486 RGB images of plant leaves and backgrounds across 39 different categories. To focus on the segmentation of black rot lesions in grape leaves, 400 grape leaf images exhibiting black rot symptoms were selected and verified by experts in grape disease. To enhance data quality and model generalization while mitigating over fitting, traditional data augmentation techniques were applied after dataset construction, including rotation, mirroring, brightness adjustment, Gaussian noise addition, and random masking. We employed an 8:1:1 split strategy, dividing the augmented set of images into three subsets. Cross-validation was not utilized in this study, this partitioning approach was designed to effectively leverage the augmented dataset. Image annotation was performed using the open-source tool Labelme, which recorded information such as the original image name, leaf contours, and black rot lesion coordinates. Fig. 1 shows examples of original images and their corresponding mask annotations in the dataset.

[1] https://github.com/xiayuchao/SFC_DeepLabv3Plus (accessed on 29 April 2025)

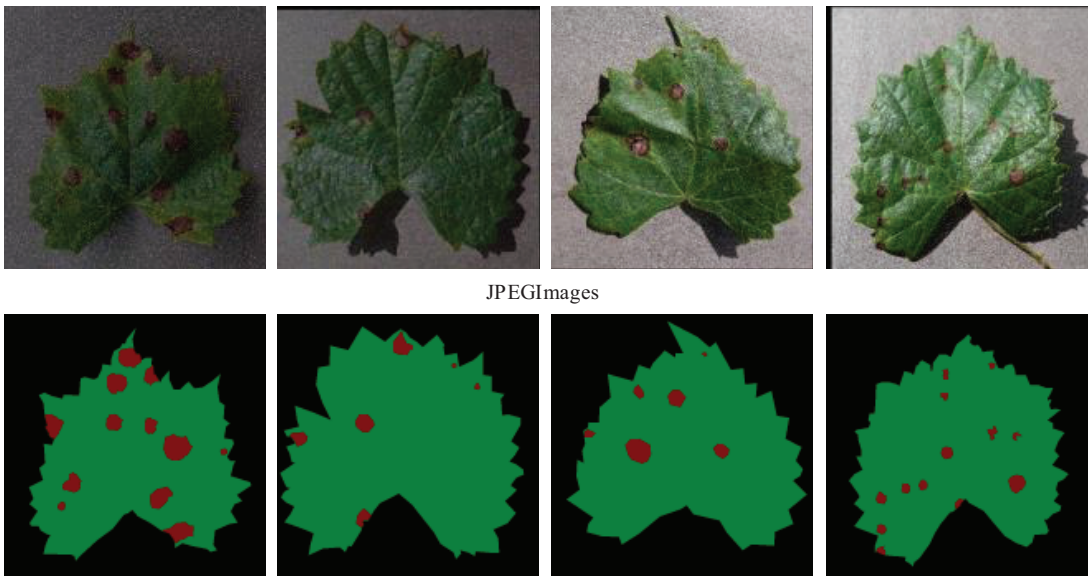

Figure 1: Examples from the segmentation-labeled dataset. JPEGImages represents the original images. SegmentationClass represents the corresponding mask annotations. Green indicates leaf contours and red denotes black rot lesions

2.2 SFC_DeepLabv3+

DeepLabv3+ employs an encoder-decoder architecture. In the encoder, feature extraction is achieved using a deep convolutional neural network (DCNN) to generate shallow and deep feature layers at different spatial resolutions. Atrous convolution and Atrous Spatial Pyramid Pooling (ASPP) are introduced to capture multi-scale features, with depthwise separable convolutions and residual connections in the Xception network enhancing feature extraction. The ASPP structure leverages atrous convolutions with varying rates to capture features at multiple scales, and a 1×1 convolution is used to adjust the channel count for the fused feature layer. In the decoder, bilinear upsampling is applied to align shallow and deep feature layers, followed by 1×1 and 3×3 convolutions for further feature extraction, producing the final prediction results.

We propose a lightweight black rot leaf segmentation model, SFC_DeepLabv3+, based on the DeepLabv3+ architecture to address issues in the original model, such as high parameter count and imprecise lesion and edge segmentation. MobileNetV2 [25] is adopted as the backbone network to reduce the parameter count and optimize computational complexity. Additionally, the global average pooling branch in ASPP is replaced with Strip Pooling [26], and a CBAM [27] attention mechanism is introduced at the input stage. We designed a content-guided attention feature fusion module (CGA Fusion [28]) to improve detection accuracy for small lesions and blurred edges. It incorporates dynamic upsampling technology (DySample_UP [29]) for feature fusion, balancing efficiency and performance. The input image is processed by MobileNetV2, producing a feature map enhanced by CBAM. This map enters ASPP, where Strip Pooling and atrous convolutions generate multi-scale features (f_{d8}, f_{d16}). CGA Fusion combines these with the shallow feature f_{low}, creating a refined feature map (low_level_features). The decoder upsamples and applies convolutions to produce the final segmentation, as shown in Fig. 2.

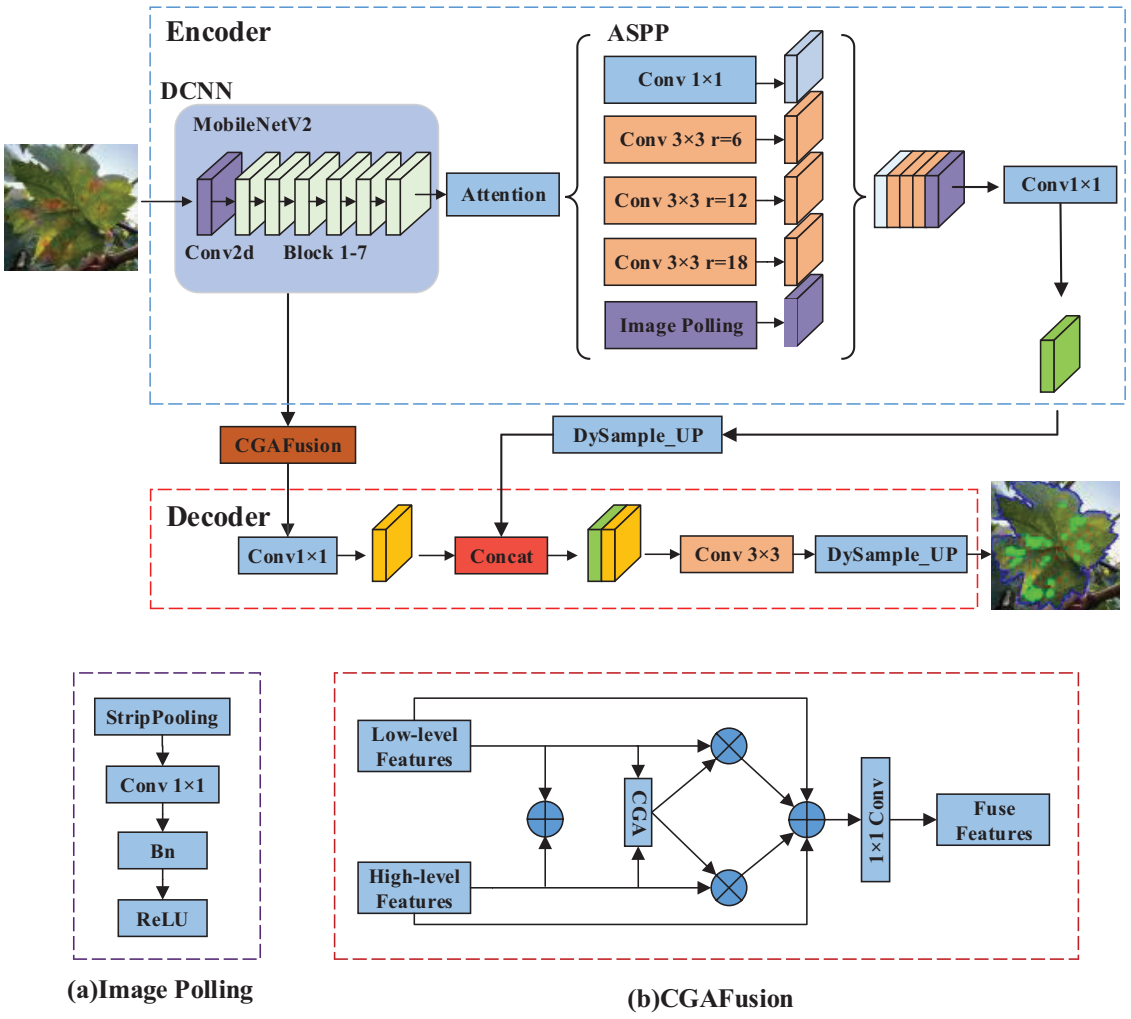

Figure 2: Architecture of SFC_DeepLabv3+

2.2.1 Backbone Network

MobileNetV2 is adopted as the backbone for *SFC_DeepLabv3+* to address the high parameter count and computational complexity of traditional segmentation models. Selected for its superior efficiency-feature extraction balance, MobileNetV2 outperforms EfficientNet [30] by avoiding computationally intensive network scaling strategies, and surpasses MobileViT [31] in handling fine-grained lesion boundaries through its convolution-based design. The network's efficiency stems from its inverted residual structure: 1×1 convolutions first expand feature dimensions, followed by lightweight 3×3 depthwise separable convolutions for spatial feature extraction, and finally project back to lower dimensions. Combined with ReLU activation for feature preservation, this architecture achieves effective preliminary feature fusion while maintaining low computational costs, making it ideal for resource-constrained leaf disease segmentation tasks.

2.2.2 CGA Fusion

In DeepLabv3+, feature maps from different layers vary significantly in scale and semantic content, making simple fusion strategies inadequate for capturing multi-scale information, especially for small lesion detection where high-level features may lose spatial details. To address this, we propose Content-Guided

Attention Fusion (CGA Fusion) which dynamically fuses features to enhance detection accuracy, adaptability to diverse scales, and segmentation performance. Using a content-guided attention mechanism, CGA Fusion optimizes channel contributions during fusion. The detailed process is outlined in Algorithm 1 and its structure is depicted in Fig. 3.

Algorithm 1: Content-guided attention fusion (CGA Fusion) algorithm

Input: Feature maps f_{low}, f_{d8}, f_{d16} from the DCNN backbone
Output: Fused feature map low_level_features
Initialization:
    $f_{\text{d16\_up}} \leftarrow \text{DySample}(f_{\text{d16}})$ ▷ Upsample f_{d16} to match f_{d8} resolution
Step 1: First Fusion
    $f_{\text{fuse1}} \leftarrow \text{CGAFusion}(f_{\text{d8}}, f_{\text{d16}_{\text{up}}})$ ▷ Fuse mid- and high-level features
Step 2: Upsampling
    $f_{\text{fuse1\_up}} \leftarrow \text{DySample}(f_{\text{fuse1}})$ ▷ Upsample f_{fuse1} to match f_{low} resolution
Step 3: Second Fusion
    low_level_features $\leftarrow \text{CGAFusion}(f_{\text{low}}, f_{\text{fuse1}_{\text{up}}})$ ▷ Fuse with low-level features
Return: low_level_features

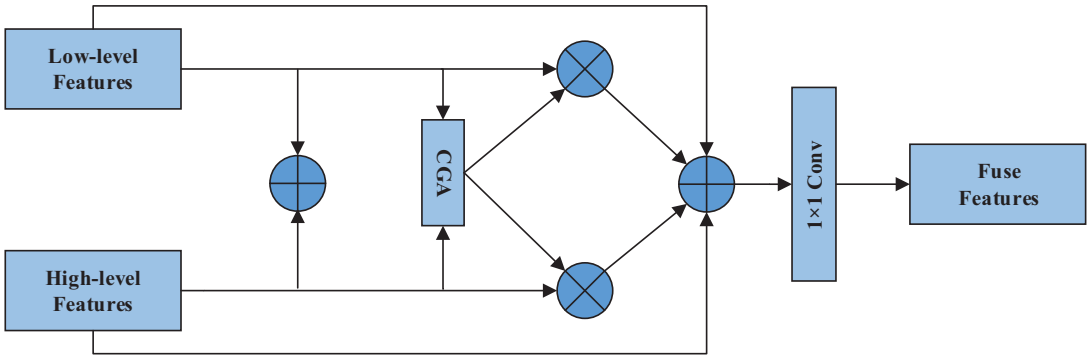

Figure 3: Structure of CGA fusion

The CGA module dynamically computes channel weights W as defined in Eq. (1), which transforms input features F into weighted features F' per Eq. (2). The FUSE module subsequently combines low-level F_{low} and high-level F_{high} features through the weight-adaptive fusion described in Eq. (3).

$$W = \sigma(\text{FC}(\text{GAP}(F))) \tag{1}$$

$$F' = W \cdot F \tag{2}$$

$$F_{\text{fuse}} = F_{\text{low}} \times W + F_{\text{high}} \times (1 - W) \tag{3}$$

2.2.3 Strip Pooling

Atrous Spatial Pyramid Pooling (ASPP) performs well in multi-scale information processing. However, it often fails to retain fine spatial details when using global average pooling for capturing contextual information, especially in edge refinement and small object detection. Strip Pooling provides an alternative approach, focusing on modeling long-range dependencies in isolated regions, allowing the network to effectively integrate global and local information. Its structure is shown in Fig. 4. Strip Pooling applies

elongated pooling kernels along either the horizontal or vertical dimension to capture global information over long-range dependencies. In the remaining spatial dimensions, it retains a narrow kernel shape to obtain detailed local context information. For a feature map of size $H \times W$, strip pooling is applied horizontally and vertically to form $H \times 1$ and $1 \times W$ shapes (Eq. (4)). The results are convolved and expanded, then fused using a 1×1 convolution and Sigmoid function (Eq. (5)).

$$X_{\text{horizontal}} = \text{AveragePool}(X, \text{size} = H \times 1), \quad X_{\text{vertical}} = \text{AveragePool}(X, \text{size} = 1 \times W) \tag{4}$$

$$X_{\text{exp}} = \text{Expand}(\text{Conv}(X_{\text{horizontal}} + X_{\text{vertical}})), \quad X_{\text{final}} = \text{Sigmoid}(\text{Conv}(X_{\text{exp}})) \cdot X \tag{5}$$

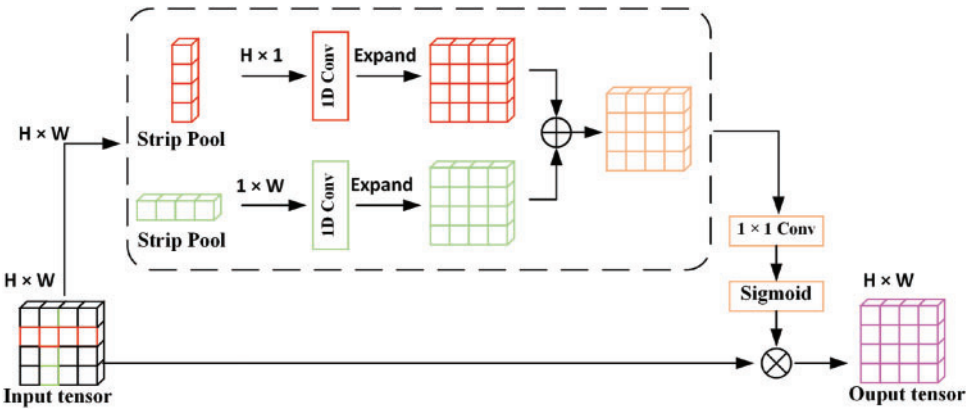

Figure 4: Structure of strip pooling

2.2.4 Upsampling Operation

The upsampling operation adjusts the size of the input feature map to enable the model to segment leaves at varying scales effectively. In the original DeepLabv3+ architecture, bilinear interpolation is commonly used for upsampling. However, this method may fail to fully recover high-level feature details and edge information, potentially leading to the loss of fine details essential for preserving the original characteristics of the image. DySample is a point-sampling-based upsampling approach that avoids the complexity of dynamic convolutions, enhancing the feature fusion capability of the network. The structure of DySample is shown in Fig. 5, where the input feature, upsampled feature, offset generation, original sampling grid, and Sigmoid function are represented by X, X', O, G and σ, respectively.

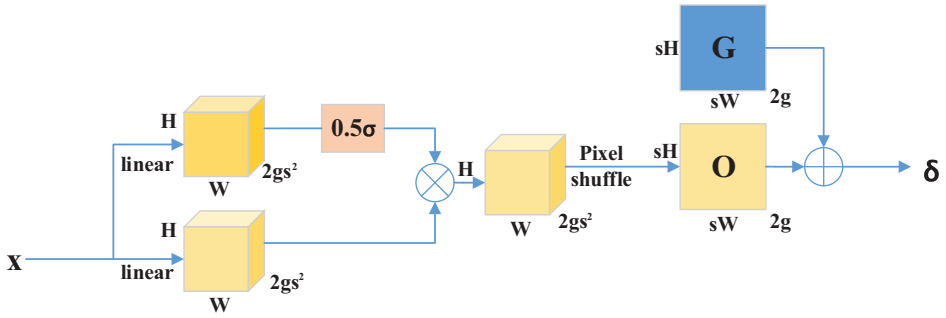

Figure 5: Structure of DySample

2.2.5 CBAM

CBAM (Convolutional Block Attention Module) is introduced into DeepLabv3+ to effectively address the uneven distribution of feature importance. By integrating CBAM into DeepLabv3+, we enhance the model's focus on task-critical features. CBAM sequentially applies channel and spatial attention mechanisms, amplifying relevant channels and regions while suppressing noise. This improvement boosts segmentation accuracy, particularly for blurred boundaries and small lesions in grape leaf analysis. The Channel Attention sub-module employs global average and max pooling to weigh channels through an MLP (Eq. (6)), while the Spatial Attention sub-module generates a spatial map using a 7×7 convolution (Eq. (7)), as illustrated in Fig. 6.

$$M_c(F) = \sigma(\mathrm{MLP}(\mathrm{AvgPool}(F)) + \mathrm{MLP}(\mathrm{MaxPool}(F))) \tag{6}$$

$$M_s(F) = \sigma(f^{7\times7}([\mathrm{AvgPool}(F); \mathrm{MaxPool}(F)])) \tag{7}$$

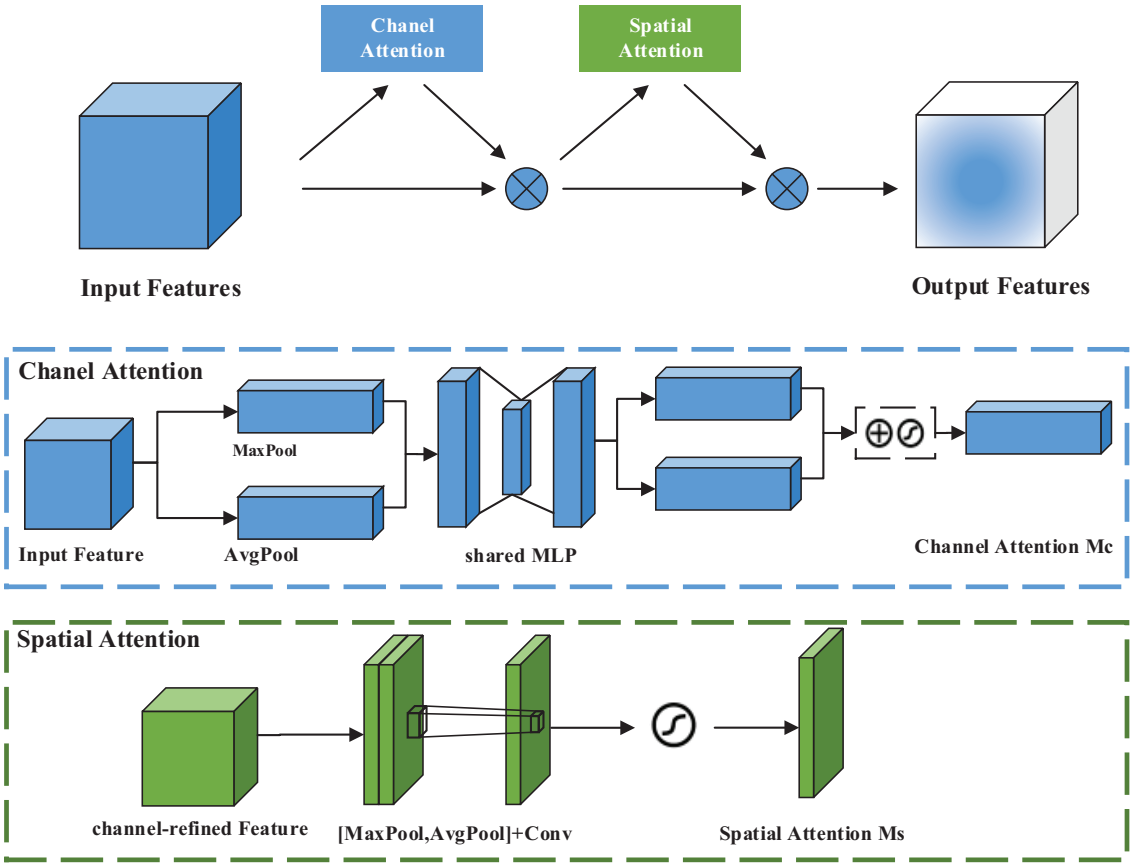

Figure 6: CBAM attention module structure

3 Results

3.1 Experimental Setup

All deep learning models in this study were trained and tested on a machine with a 32-core Intel(R) Xeon(R) Gold 6326 CPU at 2.9 GHz and an NVIDIA RTX A4000 GPU. All model training was conducted in PyTorch 1.13.1 using Python 3.8 and CUDA 11.7.

The training process consisted of frozen training and unfrozen training. Initially, to facilitate rapid and stable convergence, the backbone parameters of a model were frozen, with training focused solely on the top layers. After a predetermined number of training cycles, all network layers were unfrozen for full parameter updating to further optimize the performance of a model.

All models were trained and tested under identical conditions using the same dataset to ensure fairness in network performance evaluation and comparability of results. All experiments utilized images at a resolution of 512×512 pixels, with a batch size of 4 and a total of 150 training epochs. The optimization used Stochastic Gradient Descent (SGD) with a momentum of 0.9 and weight decay of 1×10^{-4} to prevent overfitting. The learning rate started at 7×10^{-3}, with a minimum of 7×10^{-5}, following a cosine annealing decay. Batch size scaling adjusted the learning rate dynamically, with bounds of 1×10^{-1} and 5×10^{-4}. Random shuffling was enabled to improve robustness, and class weights were set to 1.0 for balanced loss.

Fig. 7 demonstrates the training dynamics of SFC_DeepLabv3+ with three key observations: (1) Training and validation losses co-converge to values below 0.1 within 150 epochs; (2) Validation loss fluctuations remain bounded within ±0.02 without showing training-validation divergence; (3) Stable convergence persists under aggressive data augmentation including random cropping and color jittering. These results confirm the model learns generalizable features without overfitting or data dependency.

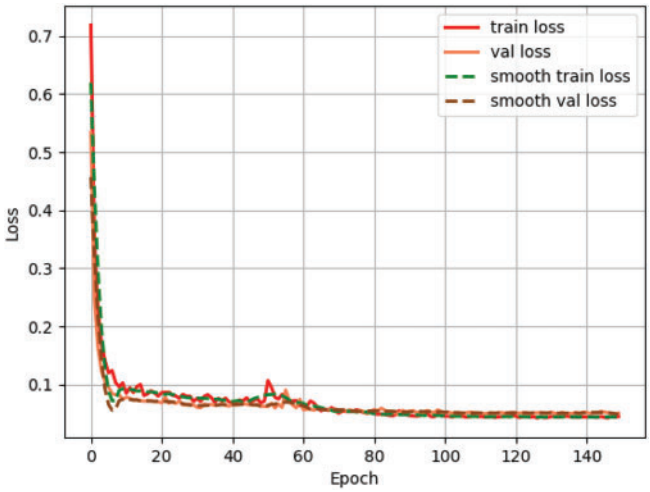

Figure 7: The evolution of both raw and smoothed loss values across training epoch

3.2 Evaluation Metrics

We employ mean intersection over union (mIoU), mean pixel accuracy (mPA) and precision to evaluate segmentation performance of models. mIoU represents the mean intersection over union across all classes, where TP_c, FP_c, and FN_c denote the true positives, false positives, and false negatives for class c, respectively. mPA is the average pixel accuracy across all classes. Precision assesses the proportion of correctly predicted positive samples among all predicted positives. They are computed as follows.

3.3 Ablation Study

An ablation study was conducted by sequentially replacing and adding modules using Xception as the backbone on the dataset used in this study to evaluate the contribution of each improvement module to segmentation performance. The experimental steps included: substituting MobileNetV2 for Xception as the backbone, replacing the traditional global average pooling branch in the ASPP module with Strip Pooling, adding CBAM before ASPP on shallow features, and finally introducing the CGA Fusion module to enhance feature fusion. The experimental configuration and parameters are detailed in Section 3.1, and the specific results of this ablation study are presented in Table 1.

Table 1: Ablation study results for SFC_DeepLabv3+ components

Model	MobileNetV2	Strip pooling	CGA Fusion	CBAM	mIoU (%)	mPA (%)	Prec. (%)
0 (Baseline)	×	×	×	×	88.76	92.55	94.95
1	✓	×	×	×	90.25	93.55	95.73
2	✓	✓	×	×	90.88	94.18	95.42
3	✓	×	✓	×	90.70	93.93	95.88
4	✓	×	×	✓	90.41	93.39	96.15
5	✓	✓	✓	×	90.74	94.03	95.83
6	✓	×	✓	✓	90.85	94.10	96.05
7	✓	✓	×	✓	90.79	94.00	95.90
8	✓	✓	✓	✓	90.98	94.33	95.68

Table 1 shows that the performance improves progressively with the step-by-step introduction of enhancement modules. **Model 0** (i.e., DeepLabv3+) uses Xception as the backbone. It serves as a benchmark for subsequent improvements.

Model 1 substitutes Xception with MobileNetV2. It improves the accuracy of mIoU, mPA by 1.49%, 1.00%, and 0.42%. This substitution maintains the lightness of the model without significantly impacting segmentation accuracy. Building on this,

Model 2 incorporates Strip Pooling to replace the global average pooling branch in the ASPP module, resulting in additional improvements of 0.63% in both mIoU and mPA, while maintaining a stable Precision. It illustrats the global semantic capturing capability of Strip Pooling significantly enhances the model's adaptability to complex scenarios.

Model 3 uses CGA Fusion for feature integration. While mIoU and mPA experience slight decreases compared to Model 2, Precision improves to 95.88%. This demonstrates the potential of CGA Fusion in enhancing segmentation accuracy, particularly for fine-grained details, though its standalone impact is limited when not combined with complementary modules.

With the inclusion of CBAM in **Model 4**, Precision achieves its highest value at 96.15%. However, mIoU and mPA show minor decreases compared to Models 2 and 3, indicating that CBAM is particularly effective in optimizing attention to key regions but contributes less to global semantic modeling.

Model 5 combines MobileNetV2, Strip Pooling, and CGA Fusion. This configuration achieves balanced performance, demonstrating the effectiveness of module integration in addressing the challenges of complex segmentation tasks.

Model 6 combines MobileNetV2, CGA Fusion, and CBAM, achieving 90.85% mIoU, 94.10% mPA, and 96.05% Precision. Compared to Model 3 (CGA Fusion alone), CBAM brings improvements of 0.15% mIoU, 0.17% mPA, and 0.17% Precision, enhancing attention refinement. Relative to Model 4 (CBAM alone), it gains 0.44% mIoU and 0.71% mPA, showing CGA Fusion's complementary global feature integration.

Model 7 integrates MobileNetV2, Strip Pooling, and CBAM, yielding 90.79% mIoU, 94.00% mPA, and 95.90% Precision. Compared to Model 2 (Strip Pooling alone), CBAM adds 0.66% Precision, boosting local attention. Relative to Model 4 (CBAM alone), Strip Pooling contributes 0.38% mIoU and 0.61% mPA, demonstrating their global-local synergy.

Model 8 integrating all enhancement modules yields the highest performance. Compared to the baseline, it delivers improvements of 2.22% in mIoU, 1.78% in mPA, and 0.89% in Precision. These results validate the complementary functionality of the modules, achieving comprehensive performance enhancements across all metrics.

3.4 Heatmap Analysis of Model

The heatmap color variation reflects the response intensity of the model in specific areas. Colors closer to red indicate a stronger response, and colors closer to blue indicate a weaker response. Hook functions are placed on each module during the forward pass to capture output feature maps without altering the overall architecture. After capturing the feature maps, channel-wise responses are averaged to generate a single-channel heatmap, which is then normalized to highlight regional response differences. The processed heatmap is overlaid onto the original image, allowing analysis of the role and contribution of models to the segmentation task.

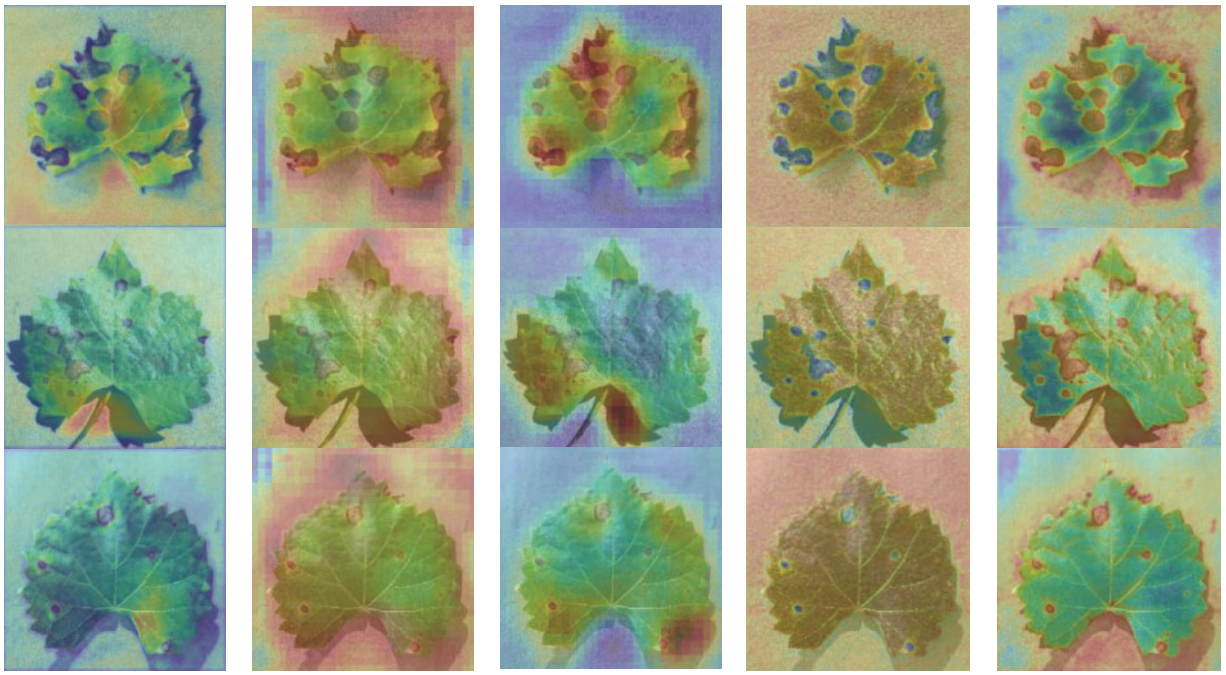

Figure 8: Performance of different enhancement models: (a) baseline model (i.e., DeepLabv3+), (b) baseline model with Strip Pooling, (c) baseline model with CBAM, (d) baseline model with CGA Fusion, and (e) SFC_DeepLabv3+

As shown in Fig. 8, the baseline model displays responses primarily concentrated in small regions, with green and blue as the dominant colors. The global receptive field of the model significantly expands after adding Strip Pooling (Fig. 8b). It results in increased red and yellow areas, reflecting broader detection and coverage. With the addition of CBAM, the response of the model becomes more focused and intense within lesion areas, especially along critical lesion boundaries, where high-intensity responses are observed. The further incorporation of the CGA Fusion module yields a wider and more uniform distribution of red and yellow regions, indicating that multi-scale feature fusion improves the stable detection of lesion areas. Finally, with all enhancement modules integrated, the red and yellow regions in the lesion areas become more prominent and well-defined, achieving optimal performance in feature extraction and fusion. This comprehensive approach enables precise and consistent recognition of lesion areas on grape leaves while effectively suppressing background noise.

3.5 Model Performance

To validate the effectiveness of the proposed enhancements, comparative experiments were conducted using the dataset in this study with models such as DeepLabv3 [32], DeepLabv3+, Unet, HRNet [33], FCN, PSPNet, and LR-ASPP [34]. All models were trained on the same dataset, with experimental settings and parameters detailed in paper's without any modification except epoch. To minimize potential experimental errors, multiple trials were conducted, and the median value was taken as the final experimental result. The segmentation results under different networks are presented in Table 2.

Table 2: Performance of models on grape leaf black rot disease segmentation

Model	Backbone Network	mIoU (%)	mPA (%)	Prec. (%)
DeepLabv3	ResNet50	86.30	90.23	90.23
DeepLabv3+	Xception	88.76	92.55	94.95
U-Net	ResNet50	88.93	92.18	95.43
HRNet	HRNetV2_w18	88.00	90.91	95.98
FCN	ResNet50	87.90	92.39	93.93
PSPNet	MobileNetV2	80.14	84.56	91.56
LR-ASPP	MobileNetV3	81.40	85.73	92.49
SFC_DeepLabv3+	MobileNetV2	90.98	94.33	95.84

Table 2 demonstrates the performance of different models on the grape leaf black rot segmentation task. SFC_DeepLabv3+ outperforms the original DeepLabv3+ in IoU, mPA, and precision with improvements of 2.22%, 1.78%, and 0.89%, respectively. It indicates significant optimization in model architecture and feature extraction mechanisms. By integrating MobileNetV2 as a lightweight backbone, combining strip pooling and CBAM, and employing CGAFusion for feature merging, SFC_DeepLabv3+ exhibits enhanced performance in fine-grained segmentation and local feature extraction. While HRNet shows a slight advantage in precision, its mIoU and mPA do not reach the levels achieved by SFC_DeepLabv3+.

4 Discussion

4.1 Computational Cost

Table 3 presents the parameters, GFLOPs, FPS, and model size for each model. Compared to DeepLabv3+, SFC_DeepLabv3+ demonstrates a substantial lightweight advantage, reducing model parameters by 49.44M, decreasing GFLOPs by 54.1, and increasing FPS by 12.46. This makes it more suitable for

real-time monitoring and detection requirements. Although other lightweight models such as PSPNet and LR-ASPP perform well in terms of Params and GFLOPs, their segmentation accuracy is considerably lower than that of SFC_DeepLabv3+.

Table 3: Parameters, GFLOPs, and FPS of models

Model	Backbone	Params/M	GFLOPs/G	Model Size/MB	FPS
DeepLabv3	ResNet50	41.99	179.30	320.79	49.65
DeepLabv3+	Xception	54.71	83.10	209.71	61.84
PSPNet	MobileNetV2	2.38	2.58	228.14	228.14
HRNet	HRNetV2_w18	9.64	16.31	31.69	31.69
FCN	ResNet50	35.31	148.27	47.53	47.53
LR-ASPP	MobileNetV3	3.22	2.03	198.51	198.51
U-Net	ResNet50	43.93	92.00	167.91	56.07
SFC_DeepLabv3+	MobileNetV2	6.27	29.00	24.23	74.30

In terms of parameter evaluation, SFC_DeepLabv3+ achieves a parameter count of 6.27M, second only to PSPNet and LR-ASPP. While these two models perform well in parameter reduction, as shown in Table 2, PSPNet achieves a parameter reduction of 3.89M compared to SFC_DeepLabv3+, but this comes at the expense of decreases in mIoU, mPA, and precision by 10.64%, 9.77%, and 4.12%, respectively. Similarly, LR-ASPP reduces parameters by 3.05M but sacrifices 9.58%, 8.6%, and 3.19% in these metrics. Additionally, SFC_DeepLabv3+ demonstrates a significant advantage in model size compared to PSPNet and LR-ASPP, making it an ideal lightweight model for applications in embedded devices or mobile platforms while maintaining high performance.

For GFLOPs, SFC_DeepLabv3+ achieves 29 G, which is relatively low compared to PSPNet, LR-ASPP, and HRNet. However, SFC_DeepLabv3+ outperforms these models in segmentation accuracy and pixel classification accuracy. Although HRNet demonstrates a lower GFLOPs than SFC_DeepLabv3+, it lags in mIoU and mPA. Furthermore, SFC_DeepLabv3+ enhances inference speed with an increase of 42.61 frames per second over HRNet, striking a balance between computational efficiency and segmentation precision.

4.2 Segmentation Prediction

We applied the models in Table 3 to the disease spot segmentation task and compared the predicted results with ground truth to provide a more visual comparison of model performance in disease spot segmentation. Based on prior discussions, models with lower scores on mIoU, mPA, and precision, such as PSPNet and LR-ASPP, are excluded from the visual predictions.

Fig. 9 reveals that SFC_DeepLabv3+ demonstrates advantages in lesion separation, edge misdetection, and omission. In the first row of Fig. 9, for tasks involving multiple lesion regions, other baseline models tend to incorrectly merge unconnected lesion areas. However, SFC_DeepLabv3+, equipped with CGA Fusion, effectively captures both global and local information during feature fusion, accurately distinguishing multiple lesion regions and reducing the issue of erroneous lesion connectivity.

The second row of Fig. 9 addresses complex edge regions. Models like DeepLabv3 and Unet often exhibit blurred or misdetected edges, particularly around leaf shadows, mistakenly identifying these shadows as lesions. SFC_DeepLabv3+ achieves more precise edge segmentation by incorporating CBAM, enhancing edge detection accuracy, allowing for smoother transitions along lesion boundaries, and effectively differentiating shadows from actual lesion edges.

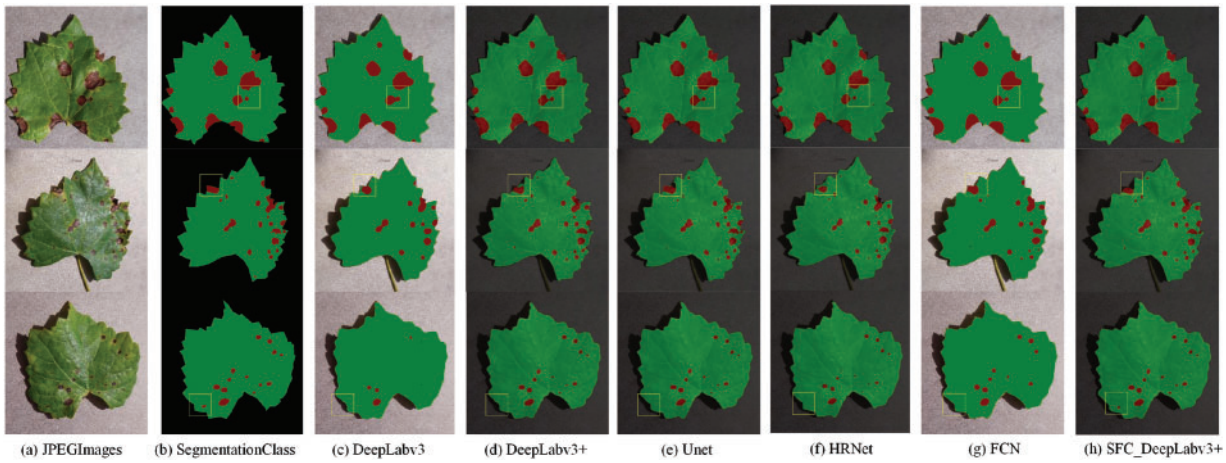

Figure 9: Results of segmentation prediction

The third row of Fig. 9 shows that SFC_DeepLabv3+ effectively detects smaller or more ambiguous lesion areas. Leveraging the lightweight feature extraction network MobileNetV2, it maintains model efficiency while boosting segmentation accuracy, successfully detecting lesions that other models fail to capture.

4.3 Limitation

While SFC_DeepLabv3+ demonstrates significant strengths, it has several limitations worth noting. First, its 6.27M parameter count, though relatively low, may still require additional memory optimization for deployment on ultra-low-capacity devices such as basic agricultural microcontrollers, where simpler models like LR-ASPP maintain a slight advantage. The computational requirement of 29G GFLOPs, while efficient, could lead to marginal performance variations on older hardware with limited floating-point capabilities, potentially necessitating configuration adjustments.

In terms of segmentation performance, the model shows minor limitations when processing extremely dense leaf patterns. The feature fusion process may occasionally prioritize prominent lesions at the expense of very faint ones, which could affect detection accuracy in complex foliage scenarios. In addition, current implementations have not been experimented under different environmental conditions (such as changes in lighting, occlusion, or weather effects) or the morphological diversity of grape leaves (such as changes in leaf shape or growth stages of specific varieties). These limitations suggest opportunities for refinement in both architectural efficiency and segmentation sensitivity for challenging agricultural environments, particularly for applications requiring robustness across diverse cultivation conditions and plant phenotypes.

5 Conclusion

In this paper, we introduce a lightweight segmentation model, SFC_DeepLabv3+, leveraging MobileNetV2 as the backbone and integrating Strip Pooling, CBAM, and CGA Fusion modules. The model addresses challenges in leaf disease segmentation, particularly for detecting blurred edges and small lesions, while significantly reducing computational complexity. Through comprehensive ablation studies, we demonstrate the individual and combined contributions of these modules to segmentation performance. By incorporating all enhancements, the model achieves relative improvements of 2.22%, 1.78%, and 0.89% in mIoU, mPA, and precision, respectively, highlighting the effectiveness of multi-scale feature fusion and dynamic attention mechanisms. Comparisons with segmentation models, including DeepLabv3+, Unet, and

HRNet, on our datasets reveal that SFC_DeepLabv3+ surpasses these models with mIoU, mPA, and precision of 90.98%, 94.33%, and 95.84%, respectively. The model excels in fine-grained segmentation and local feature extraction. Furthermore, SFC_DeepLabv3+ demonstrates significant advantages in parameter efficiency, model size, and inference speed, establishing it as a balanced solution for performance and efficiency in leaf disease segmentation.

Future work includes pre-training on additional plant leaf lesion datasets to enhance segmentation and detection in real-world applications. Additionally, optimizing the model for deployment on embedded devices through techniques such as quantization and pruning could further improve its suitability for real-time agricultural monitoring. Exploring multi-modal data fusion, integrating image data with spectral or environmental information, may also enhance segmentation accuracy and robustness, opening new avenues for early disease detection.

Acknowledgement: The authors gratefully acknowledge the institutional support provided by Zhejiang A&F University for this research. Special thanks are extended to colleagues who provided valuable technical assistance and constructive feedback.

Funding Statement: This work was supported by the following grants: Zhejiang A&F University Research Development Fund (Talent Initiation Project No. 2021LFR048) and 2023 University-Enterprise Joint Research Program (Grant No. LHYFZ2302) from the Modern Agricultural and Forestry Artificial Intelligence Industry Academy.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Yuchao Xia, Jing Qiu; model development and implementation: Yuchao Xia; data collection and annotation: Yuchao Xia; analysis and interpretation of results: Yuchao Xia, Jing Qiu; draft manuscript preparation: Yuchao Xia; critical revision and editing: Jing Qiu. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The dataset used in this study along with its annotation files, trained model weights, and complete source code can be accessed at https://github.com/xiayuchao/SFC_DeepLabv3Plus (accessed on 29 April 2025).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Fraiwan M, Faouri E, Khasawneh N. Multiclass classification of grape diseases using deep artificial intelligence. Agriculture. 2022;12(10):1542. doi:10.3390/agriculture12101542.
2. Kunduracioglu I, Paçal I. Advancements in deep learning for accurate classification of grape leaves and diagnosis of grape diseases. J Plant Dis Prot. 2024;131(3):1061–80. doi:10.1007/s41348-024-00896-z.
3. Alajas OJY, Concepcion RS, Dadios ED, Sybingco E, Mendigoria CHR, Aquino HL. Prediction of grape leaf black rot damaged surface percentage using hybrid linear discriminant analysis and decision tree. In: 2021 International Conference on Intelligent Technologies (CONIT); 2021; Hubli, India. p. 1–6.
4. Babu PR, Srikrishna A, Gera VR. Diagnosis of tomato leaf disease using OTSU multi-threshold image segmentation-based chimp optimization algorithm and LeNet-5 classifier. J Plant Dis Prot. 2024;131(6):2221–36. doi:10.1007/s41348-024-00953-7.
5. Singh J, Kaur H. Plant disease detection based on region-based segmentation and KNN classifier. In: Pandian D, Fernando X, Baig Z, Shi F, editors. Proceedings of the International Conference on ISMAC in Computational Vision and Bio-Engineering 2018 (ISMAC-CVB). Cham, Switzerland: Springer International Publishing; 2019. p. 1667–75. doi:10.1007/978-3-030-00665-5_154.

6. Sudha PN, Kumaran P. Early detection and control of anthracnose disease in cashew leaves to improve crop yield using image processing and machine learning techniques. Signal Image Video Process. 2023;17(7):3323–30. doi:10.1007/s11760-023-02552-9.
7. Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2015 Jun 7–12; Boston, MA, USA. p. 3431–40.
8. Badrinarayanan V, Kendall A, Cipolla R. Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Transact Pattern Anal Mach Intell. 2017;39(12):2481–95. doi:10.1109/tpami.2016.2644615.
9. Noh H, Hong S, Han B. Learning deconvolution network for semantic segmentation. In: 2015 IEEE International Conference on Computer Vision (ICCV); 2015 Dec 7–13; Santiago, Chile; 2015. p. 1520–8.
10. 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; 2017 Jul 21–26; Honolulu, HI, USA. p. 2881–90.
11. Kumar D, Kukreja V. Application of PSPNET and fuzzy logic for wheat leaf rust disease and its severity. In: 2022 International Conference on Data Analytics for Business and Industry (ICDABI); 2022. p. 547–51. doi:10.1109/ICDABI56818.2022.10041575.
12. He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: 2017 IEEE International Conference on Computer Vision (ICCV); 2017 Oct 22–29; Venice, Italy. p. 2980–8.
13. Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Transact Pattern Anal Mach Intell. 2017;39(6):1137–49. doi:10.1109/tpami.2016.2577031.
14. Ronneberger O, Fischer P, Brox T, Navab N, Hornegger J, Wells WM, et al. U-Net: convolutional networks for biomedical image segmentation. Cham, Switzerland: Springer International Publishing; 2015.
15. Yi X, Zhou Y, Wu P, Wang G, Mo L, Chola M, et al. U-Net with coordinate attention and VGGNet: a grape image segmentation algorithm based on fusion pyramid pooling and the dual-attention mechanism. Agronomy. 2024;14(5):925. doi:10.3390/agronomy14050925.
16. Yu C, Wang J, Peng C, Gao C, Yu G, Sang N et al., BiSeNet: bilateral segmentation network for real-time semantic segmentation. Cham, Switzerland: Springer International Publishing; 2018.
17. Yuan Y, Chen X, Wang J, Bischof H, Brox T, Frahm JM. Object-contextual representations for semantic segmentation. Cham, Switzerland: Springer International Publishing; 2020.
18. Paszke A, Chaurasia A, Kim S, Culurciello E. ENet: a deep neural network architecture for real-time semantic segmentation. arXiv:1606.02147. 2016.
19. Poudel RPK, Liwicki S, Cipolla R. Fast-SCNN: fast semantic segmentation network. arXiv:1902.04502. 2019.
20. Chen L, Papandreou G, Kokkinos I, Murphy K, Yuille AL. DeepLab: semantic image segmentation with deep convolutional nets, Atrous convolution, and fully connected CRFs. arXiv:1606.00915. 2016.
21. Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y, editors. Computer Vision–ECCV 2018. Cham, Switzerland: Springer International Publishing; 2018. p. 833–51.
22. Zhang Y, Wang H, Liu J, Zhao X, Lu Y, Qu T, et al. A lightweight winter wheat planting area extraction model based on improved DeepLabv3+ and CBAM. Remote Sens. 2023;15(17):4156. doi:10.3390/rs15174156.
23. Mo L, Fan Y, Wang G, Yi X, Wu X, Wu P. DeepMDSCBA: an improved semantic segmentation model based on DeepLabV3+ for apple images. Foods. 2022;11(24):3999. doi:10.3390/foods11243999.
24. Hughes DP, Salathé M. An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing. arXiv:1511.08060. 2015.
25. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. MobileNetV2: inverted residuals and linear bottlenecks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA. p. 4510–20.
26. Hou Q, Zhang L, Cheng MM, Feng J. Strip pooling: rethinking spatial pooling for scene parsing. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun 13–19; Seattle, WA, USA. p. 4002–11.
27. Woo S, Park J, Lee JY, Kweon IS. CBAM: convolutional block attention module. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y, editors. Computer vision–ECCV 2018. Cham, Switzerland: Springer International Publishing; 2018. p. 3–19.

28. Chen Z, He Z, Lu Z. DEA-Net: single image dehazing based on detail-enhanced convolution and content-guided attention. arXiv:2301.04805. 2023.
29. Liu W, Lu H, Fu H, Cao Z. Learning to upsample by learning to sample. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV); 2023 Oct 1–6; Paris, France. p. 6004–14.
30. Tan M, Le QV. EfficientNet: rethinking model scaling for convolutional neural networks. arXiv:1905.11946. 2019.
31. Mehta S, Rastegari M. MobileViT: light-weight, general-purpose, and mobile-friendly vision transformer. arXiv:2110.02178. 2021.
32. Chen L, Papandreou G, Schroff F, Adam H. Rethinking Atrous convolution for semantic image segmentation. arXiv:1706.05587. 2017.
33. Wang J, Sun K, Cheng T, Jiang B, Deng C, Zhao Y, et al. Deep high-resolution representation learning for visual recognition. IEEE Transact Pattern Analy Mach Intell. 2021;43(10):3349–64. doi:10.1109/tpami.2020.2983686.
34. Howard A, Sandler M, Chen B, Wang W, Chen LC, Tan M, et al. Searching for MobileNetV3. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV); 2019 Oct 27–Nov 2; Seoul, Republic of Korea. p. 1314–24.