Journal of Real-Time Image Processing (2024) 21:162
https://doi.org/10.1007/s11554-024-01540-7

RESEARCH

ESC-YOLO: optimizing apple fruit recognition with efficient spatial and channel features in YOLOX

Jun Sun[1] · Yifei Peng[1] · Chen Chen[1] · Bing Zhang[2] · Zhaoqi Wu[1] · Yilin Jia[1] · Lei Shi[1]

Received: 26 June 2024 / Accepted: 14 August 2024

Abstract

Accurate localization of apple fruits and recognition of occlusion types in complex orchard environments play an important role in precision agriculture. This work proposes an efficient fruit recognition model called Efficient Spatial and Channel Feature YOLOX (ESC-YOLO). ESC-YOLO is built upon YOLOX and fully leverages and emphasizes spatial channel information, ensuring coherence between global information and local features. The optimization strategies for the backbone network involve adopting EfficientViT as the foundational backbone, integrating Spatial and Channel Reconstruction Convolution (SCConv) into the input stem to reorganize spatial channel features and reduce redundancy, and constructing the Efficient-MBConv module, which is optimally combined with the EfficientViTBlock for feature extraction. The optimization strategies for the neck network involve utilizing the Centralized Feature Pyramid Net (CFPNet) as the neck network and employing a Simple, Parameter-Free Attention Module (SimAM) to enhance model performance. In this work, we adopted the lightweight model of the ESC-YOLO for performance evaluation, namely ESC-YOLO-S. It achieves a 4.26% improvement in Top-1 mean Average Precision (mAP) compared to YOLOX-S and significantly reduces the false and missed detections caused by various types of occlusions. Therefore, the improved model meets the requirements for high-precision identification in complex orchard environments.

Keywords Object detection · Fruit recognition · YOLOX · Spatial channel information · Multi-scale features

1 Introduction

The apple tree belongs to the deciduous Rosaceae family. Its spherical fruits are rich in nutrients and represent significant crops in the fields of research and economics. Apples stand out as one of the most advantageous agricultural products globally, with widespread cultivation and production. To realize the efficient and automatic picking of apples, to ensure the timely harvest of mature fruits, and to improve the competitiveness of the apple market, further study of the key technologies of the apple-picking robot is essential [1, 2]. The labor-saving mechanical automation planting and harvesting mode will reduce the adverse effects caused by increased labor costs, promoting the development of agricultural intelligence.

The vision system is one of the most important parts of apple picking robot [3]. Fruit recognition and localization within the visual system play a crucial role in automated apple harvesting. Recent advancements in fruit detection research have shown promising progress. Zhang et al. [4] enhanced the YOLOv4 model by embedding attention mechanisms, resulting in a 3.45% increase in mean Average Precision (mAP) and achieving lightweighting. Divyanth et al. [5] employed generative adversarial networks to capture depth information on apple tree canopies, achieving only a 3.5% relative error. Wu et al. [6] utilized leaf insertion data augmentation to simulate apple fruits with insufficient feature visibility due to random occlusion. These algorithms focus on detecting single fruit features like shape, texture, and color, or single harvesting factors such as depth, occlusion, and lighting. This allows for accurate harvesting of fruits in complex environments. However, indiscriminate harvesting may lead to damage to robotic arms or fruit falling due to

✉ Jun Sun
sun2000jun@sina.com

[1] School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212013, China

[2] School of Electrical and Information Engineering, Changzhou Institute of Technology, Changzhou 213032, Jiangsu, China

Published online: 29 August 2024

collisions, making it unsuitable for tasks involving complex occlusion.

With the development of technology, the application of convolutional neural networks in agriculture is becoming increasingly extensive. For example, Sun et al. [7] proposed an improved wheat head counting network that achieved precise detection and counting of small objects within complex backgrounds. Currently, there are two main types of target detection algorithms: one-stage and two-stage. The SSD [8] and YOLO [9–12] series belong to the one-stage object detection models, widely used for real-time object detection due to their efficient speed and continuous accuracy enhancements. The R-CNN series are two-stage object detection models, which extract candidate regions and perform classification and bounding box regression for them. Classic models include R-CNN [13], Fast R-CNN [14], Faster R-CNN [15], and Mask R-CNN [16].

Extensive research efforts and numerous scientific attempts have been dedicated to fruit recognition through deep learning and have achieved some progress. However, various types of occlusions have an impact on fruit detection and automated production, leading to positional deviations due to occlusion and blurriness, as well as false and missed detections. While simple fruit recognition tasks have achieved high accuracy, the challenge of determining whether a fruit is harvestable and accurately locating fruits with insufficient features remains in automated harvesting. To identify the types of occlusions in complex environments, this work proposes an improved apple fruit recognition model based on YOLOX [17]. This work presents an improved model that fully leverages and emphasizes spatial and channel information. The model achieves precise classification learning of multi-type and multi-scale features through a more comprehensive fusion of high fluidity and coherent feature information. Experimental results show that the model significantly improves recognition accuracy and reduces false and missed detections. Therefore, the improved model meets the requirements for high-precision identification in complex orchard environments. It provides new insights into the application of real-time detection in fruit recognition and contributes to automated harvesting decision-making by providing effective technical support for apple-picking robots.

2 Materials and methods

2.1 Data sources

The dataset used in the experiments includes 800 images of mature Scifresh apples from a commercial orchard near Prosser, Washington. These pictures were captured by a Microsoft Kinect V2 sensor at a height of 1.7 m above the ground, each with a resolution of 1920 × 1080 pixels. This dataset contains diverse occlusions of apples with different scales, poses, lighting conditions against complex backgrounds. These diverse characteristics contribute to enriching the variety of training samples and enhancing the generalization capability of the model.

2.2 Dataset construction

In the orchard, various occlusion scenarios can affect fruit picking. These include leaves, branches, and fruit occlusion, as well as potential obstructions from wires, stakes, and pipes. In situations with various occlusions, inadequate harvesting strategies may result in the robotic arm squeezing or colliding with nearby fruits or becoming entangled with branches and other obstacles. This could lead to damage to both the fruits and the robotic arm.

This work categorizes common occlusion scenarios into four types. The first is leaf-shaped occlusion, mainly caused by foliage. The second is strip-shaped occlusion, covering branches, wires, supports, and pipes. The third is spherical occlusion, primarily caused by fruit occlusion. The fourth is no occlusion (Fig. 1). When the target under examination encounters multiple occlusions simultaneously, the safety of harvesting is taken into consideration. Strip-shaped occlusions are given priority over spherical occlusions, and spherical occlusions are given priority over leaf-shaped occlusions. Based on these four types of occlusions and labeling principles, LabelImg is used to annotate the apple dataset. The annotation of fruits depends on the minimum bounding rectangle of the apple fruits. The size of the bounding box corresponds to the region of the image and the area of the original environment, which can be categorized as small, medium, or large. Therefore, the size of the bounding box can also indicate the distance of the fruit target from the camera. Due to the limited reach of the robotic arm, small targets in the distance are considered as background and not labeled to reduce false detections and performance waste. The apple images were processed with random image augmentation after completing the annotation. This step was aimed at enhancing the richness and diversity of the training dataset to improve the model's generalization capability and robustness while reducing data bias to mitigate the risk of overfitting. Following image augmentation, the dataset was expanded to 4800 images and randomly divided into training set, validation set, and test set in a ratio of 7:2:1.

2.3 Model selection

Due to the diversity in dataset characteristics and detection model structures, different models perform differently on various datasets. Figure 2 presents the results obtained from training the apple dataset using YOLO series object

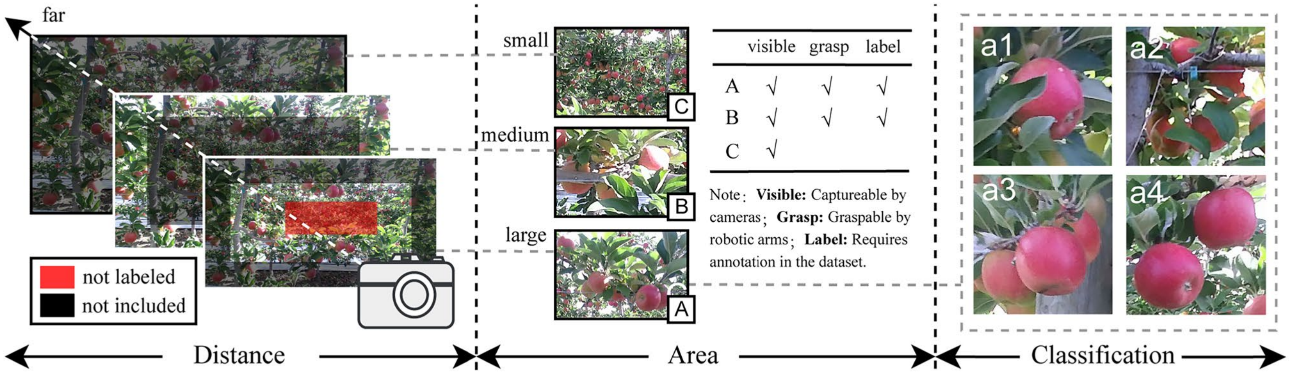
far
not labeled
not included
small
medium
large
C
B
A
visible grasp label
A √ √ √
B √ √ √
C √
Note: Visible: Captureable by cameras; Grasp: Graspable by robotic arms; Label: Requires annotation in the dataset.
a1
a2
a3
a4
Distance
Area
Classification

Fig. 1 Schematic diagram illustrating the correlation between the size of the bounding box and the distance of targets from the camera Classification of fruit occlusion: a1 for leaf-shaped, a2 for strip-shaped, a3 for spherical, and a4 for no occlusion

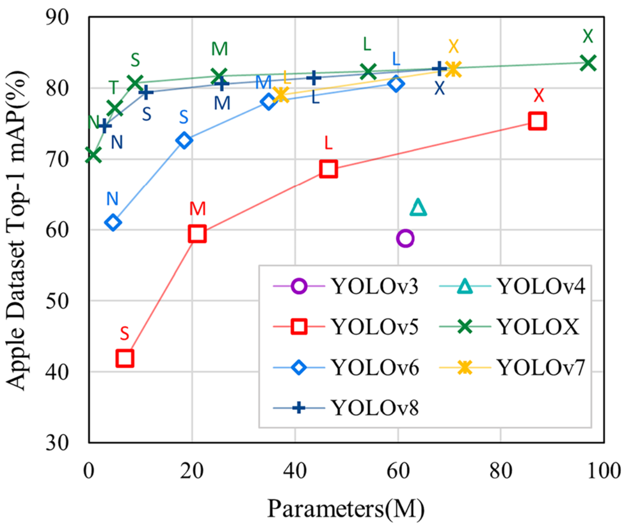

Fig. 2 Performance comparison of YOLO series detection models

detection models. It is crucial to select a suitable detection model based on the specific task and dataset features. Compared to other models in the YOLO series, YOLOX achieved the highest mAP when trained on the apple dataset. Additionally, it demonstrated faster convergence speed and lower latency. The scale of models from *N* to *X* increases sequentially. The *N* and *T* versions of YOLOX do not meet the experimental requirements in terms of recognition accuracy. The *M*, *L*, and *X* versions do not significantly improve accuracy despite having larger parameters (Fig. 2).

Therefore, this work adopts the YOLOX-S as the base model for apple fruit recognition tasks. This model has moderate parameters and strikes a balance between accuracy and detection speed.

2.4 Efficient spatial and channel Feature YOLOX

The backbone feature extraction network of YOLOX is CSPDarknet, while the enhanced feature extraction network adopts the Path Aggregation Network (PANet) [18] structure. In YOLOX, the decoupled detection head serves as the classifier and regressor, achieving separation of predicted classification, regression, and Intersection over Union (IoU) parameters. These structures enable YOLOX to demonstrate superior performance.

Due to the complexity of orchard environments and the multitude of detection targets, accurately identifying occlusion situations poses a challenge to existing YOLOX models. To provide a more suitable detection solution for identifying apple occlusions in orchard environments, this work proposes an improved detection model (Fig. 3). Efficient Spatial and Channel Feature (ESC-YOLO) is built upon YOLOX, with the primary objective of enhancing the utilization of spatial and channel information. The improvement aims to improve the flow and coherence of feature information within the model, fully learn and focus on global information and local features, as well as achieve recognition and detection of various feature information and small targets in complex environments.

Firstly, the improved EfficientViT [19] serves as the backbone network of YOLOX. Within the ResBlock [20] of EfficientViT, Spatial and Channel Reconstruction Convolution (SCConv) [21] are added to further reduce feature redundancy, decrease model computational complexity, and improve model generalization capability. Secondly, the Efficient Mobile Inverted Residual Bottleneck Convolution (E-MBConv) module embedded with Efficient Local Attention (ELA) [22] is employed to replace certain MBConv [23, 24] modules in EfficientViT. This substitution aims to improve the efficiency of utilizing spatial information and paying attention to local spatial information. Thirdly, the Centralized Feature Pyramid Net (CFPNet) [25] is employed to replace the Path Aggregation

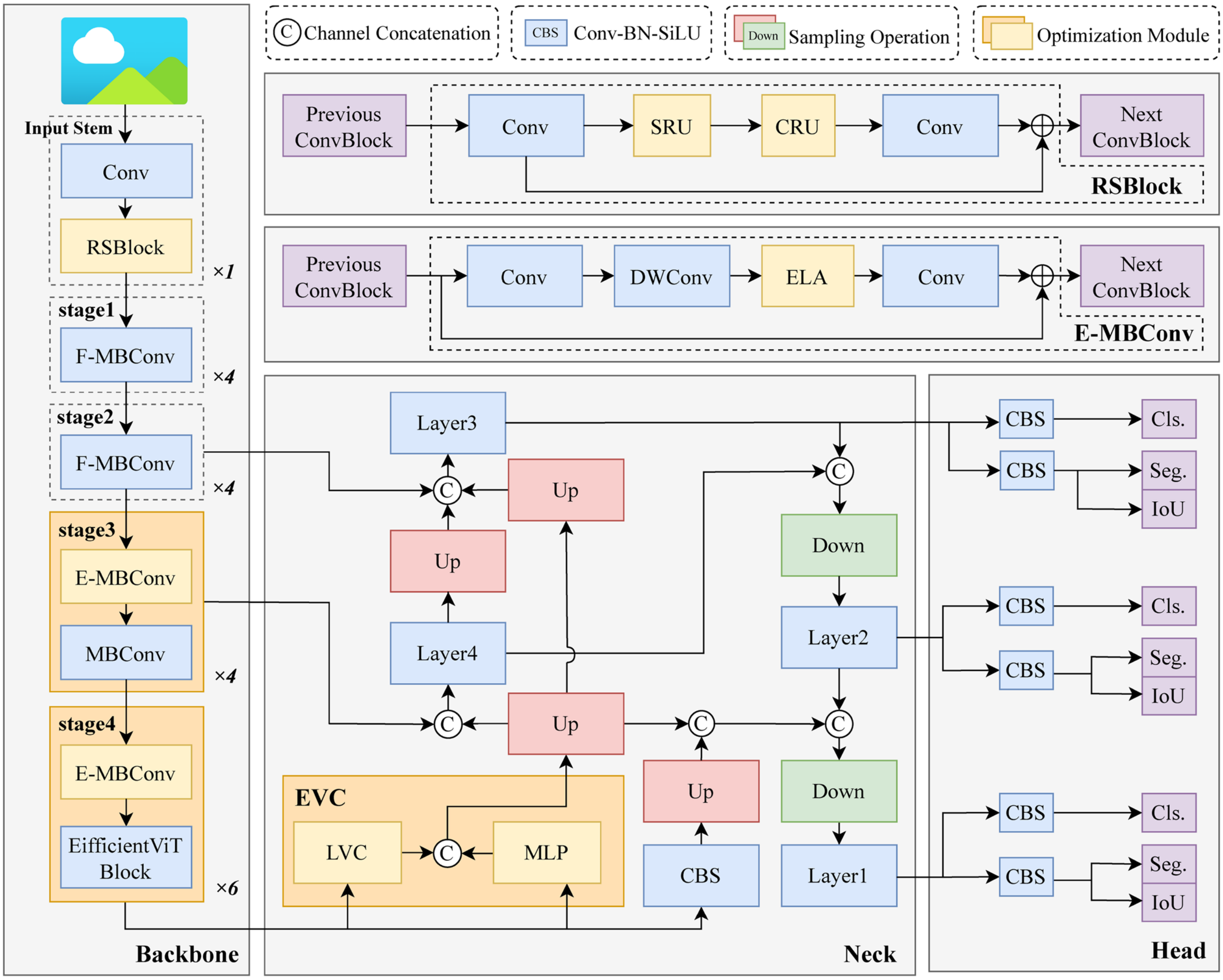

Fig. 3 The schematic diagram of the overall structure of ESC-YOLO. Here, **RSBlock** refers to embedding SCConv into the ResBlock module; **E-MBConv** refers to the Efficient-MBConv module, which represents replacing the SE module with the ELA in the MBConv. **Cls.** and **Seg.** respectively represent predictions for classification and regression

Feature Pyramid Network (PAFPN) neck network of YOLOX. This change is made with the intention of capturing long-range dependencies and focusing on feature information at multiple scales. By implementing a globally unbiased learning pattern, the model's detection capability for dense targets has been strengthened. Additionally, embedding the SimAM [26] into the neck network endows the model with both spatial and channel attention. Finally, the Generalized Intersection over Union (GIoU) [27] is employed as the optimized loss function to accurately assess the performance of the fruit recognition model.

3 Model construction and analysis

3.1 Experimental setup

3.1.1 Environment and platform

All experiments were conducted on a 64-bit Ubuntu 18.04.6 LTS system using PyTorch 1.11.0 for deep learning and CUDA 11.4 for parallel computing. The hardware specifications include an Intel Core i7-10700K CPU@3.80 GHz × 16, 32 GB of RAM, and an NVIDIA GeForce RTX 3080 graphics card with 10 GB of VRAM.

Table 1 The performance comparison shows results from replacing the backbone network with various feature extraction networks while utilizing YOLOX-S as the base model

Backbone network	Top-1 mAP (%)	Latency (ms)	Params (M)
EfficientViT	84.11	9.60	24.532
YOLOX-S(base)	80.67	9.54	8.939
ConvNeXt [30]	78.01	13.30	26.854
EfficientNet [24]	83.04	13.28	18.504
EfficientNetV2 [31]	83.52	17.91	75.288
MobileOne [32]	75.14	12.10	5.684
RepViT [33]	80.96	13.90	12.501
ResNet [20]	76.18	9.51	6.211
ResNeXt [34]	79.15	9.33	6.181

3.1.2 Hyperparameter settings

The model is configured to utilize multi-threaded processing with 4 threads and unfreeze the backbone for training all network parameters. The Adaptive Moment Estimation (Adam) optimizer is used to optimize the network parameters, with the maximum learning rate set at 0.001, momentum set at 0.937, and another hyperparameter set at 0.999 [28]. The batch size is set at 4, and the number of epochs is optimized during the training process to ensure that the model is trained until convergence.

3.2 Improved backbone feature extraction network

3.2.1 EfficientViT

The Vision Transformer (ViT) [29] is a model that utilizes self-attention mechanisms to process image data and has been successful due to its efficient modeling capabilities. However, significant performance advantages come with high computational costs. EfficientViT adopts cascaded and lightweight multi-scale linear attention modules to reduce computational redundancy and enhance attention diversity for global receptive fields and multi-scale learning.

According to Table 1, EfficientViT demonstrates a significant improvement in recognition accuracy compared to other backbone networks. In comparison to lightweight models such as MobileOne, ResNet, and ResNeXt, achieves excellent frames per second. Compared to RepViT, which also belongs to the ViT, EfficientViT overcomes the high complexity and computational demands of ViT. By combining the efficient design of EfficientNet and EfficientNetV2 with the self-attention mechanism of ViT, it achieves a good balance between speed and accuracy while maintaining moderate parameters. Therefore, it surpasses the performance of other tested feature extraction networks.

3.2.2 Spatial and channel reconstruction convolution

SCConv consists of two parts: the Spatial Reconstruction Unit (SRU) and the Channel Reconstruction Unit (CRU). The SRU utilizes separation-reconstruction method to suppress spatial redundancy, while the CRU employs a segmentation-transformation-fusion strategy to reduce channel redundancy and lower computational costs and complexity. The utilization of SCConv strengthens the backbone network's attention to spatial channel information. As shown in Eq. (1), the input feature maps are normalized through Group Normalization (GN) [35].

$$X_{out} = GN(X) = \frac{(X - \mu)}{\sqrt{\sigma^2 + \varepsilon}} \tag{1}$$

Here X is the input feature, μ and σ are the mean and standard deviation in X, ε is a small constant. After normalization, using linear affine transformation to obtain the indicator parameter γ for spatial information richness as described in Eq. (2). Here γ and β are trainable affine transformation. The higher value of γ, the richer spatial information contained in the feature maps.

$$X_{out} = \gamma X + \beta \tag{2}$$

Obtain ω through Eq. (3). The higher value of ω, the more important the layer is.

$$\omega_i = \frac{\gamma_i}{\sum_{k=0} \gamma_j}, \quad i,j = 1,2,\ldots,C \tag{3}$$

After determining the spatial distribution of information richness, the weights of the feature maps are reweighted and then mapped through the sigmoid function. Selective passage is conducted based on the threshold σ using the gate function Eq. (4), where W_1, W_2 represent the informative weights and non-informative weights.

$$\begin{gathered} X_1^w = \begin{cases} 1 & > \sigma \\ W_1 \otimes X_1^{(i)} & \leq \sigma \end{cases}, \\ X_2^w = \begin{cases} W_2 \otimes X_2^{(i)} & > \sigma \\ 0 & \leq \sigma \end{cases} \end{gathered} \tag{4}$$

The output X_1^W is rich in spatial information, while X_2^W with lower content can be considered redundant. The two are split from the channel and numerically added in an alternating manner to obtain X^{W1} and X^{W2}. These results are concatenated along the depth dimension to complete the spatial reconstruction. Spatial redundancy occurs when convolutional neural networks need to focus on all feature maps, but there is an uneven distribution of spatial information among them. Through conducting spatial reconstruction, the features are weighted based on their richness of information, resulting in the extraction of features with richer

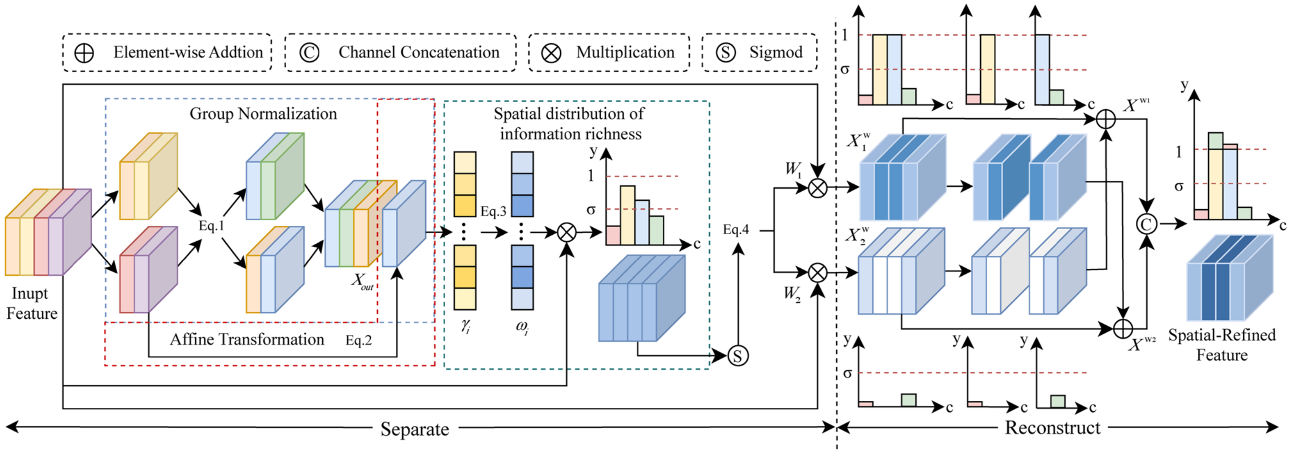

Fig. 4 The architecture of Spatial Reconstruction Unit and the flow of spatial information during separation and reconstruction. On the coordinate axis, **c** represents channel and **y** represents the richness of spatial information

information. This process enriches the feature information and reduces spatial redundancy by transferring feature information from layers with poor information to layers with abundant information (Fig. 4).

In the channel convolution, the output of the SRU undergoes channel compression when used as input. This method aims to reduce information redundancy by decreasing the information content in the feature channels. However, this compression process may result in the loss of some feature information. Therefore, the features with αC channels serve as the rich feature extractor, while those with $(1-\alpha)C$ channels serve as the shallow detail extractor. The rich feature extractor utilizes group convolution with a large receptive field to capture global information, while pointwise convolution operates on individual pixels to retain local detail information. This compensates for the loss of information in sparse convolution. The shallow detail extractor utilizes pointwise convolution with a single-element receptive field to capture local detail features. Additionally, channel concatenation fusion is employed to merge detail features from different levels and enhance the network's expressive capability. As shown in Eq. (5), global average pooling is employed to obtain global spatial information with channel-wise statistics S_m. Here, H and W represent the height and width of the feature map, while $Y_c(i,j)$ denotes the value corresponding to channel c at position (i,j) in the input feature map.

$$S_m = Pooling(Y_m) = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W} Y_c(i,j), \quad m = 1,2 \tag{5}$$

The values of two one-dimensional tensors are expanded along channels and normalized according to Eq. (6).

$$\beta_1 = \frac{e^{s_{1k}}}{e^{s_{1k}} + e^{s_{2k}}},\ \beta_2 = \frac{e^{s_{2k}}}{e^{s_{1k}} + e^{s_{2k}}}, \quad k \in [1, C] \tag{6}$$

Finally, the feature importance vector β_1, β_2 will be used to obtain channel-refined features Y according to Eq. (7), where Y_1, Y_2 represent the upper and lower features.

$$Y = \beta_1 Y_1 + \beta_2 Y_2 \tag{7}$$

In the upper segment of the channel reconstruction module, the rich global information is merged with local features. Strengthening common features through element-wise addition contributes to enhancing the expressiveness and generalization capability of the features. In the lower segment of the channel reconstruction module, richer semantic information is acquired by concatenating the extracted detailed features with the original image (Fig. 5).

3.2.3 Efficient-MBConv

The standard configuration of the EfficientViT module comprises an input stem and four stages. Stages 3 and 4 of the EfficientViT consist of MBConv and EfficientViTBlock. The EfficientViTBlock can be substituted with MBConv and E-MBConv, resulting in different combinations. Table 2

Table 2 EfficientViT serves as the backbone network, with stages 3 and 4 employing different combinations of MBConv and EfficientViTBlock for comparative analysis of model performance

Combination	Top-1 mAP (%)	Latency (ms)	Params (M)
[EVB, EVB]	82.33	13.39	75.994
[MBC, EVB]	80.70 (− 1.63)	9.39	68.909
[MBC, MBC]	83.11 (+ 0.78)	11.31	74.506

Here, MBC represents the MBConv and EVB represents the EfficientViTBlock

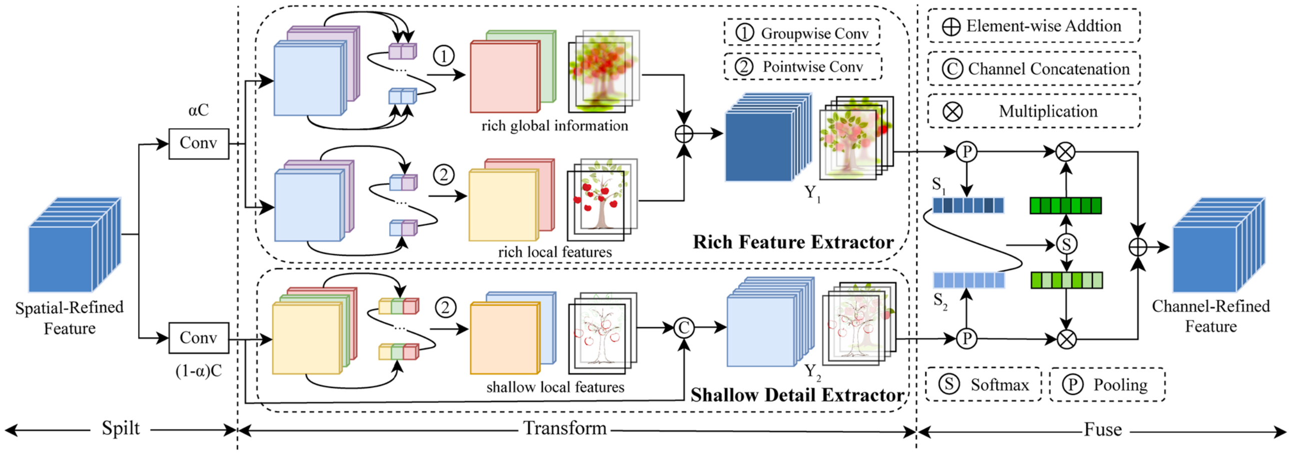

Fig. 5 The architecture of the Channel Reconstruction Unit. Using fruit trees to represent global information, while branches, fruits, and leaves to represent local information. This approach concretely demonstrates how convolutional operations extract different types of information

illustrates the impact of the backbone network on the model performance under different combinations.

The MBConv module has a simpler structure compared to the EfficientViTBlock, significantly reducing parameters and computations. However, its performance in complex tasks is inferior to that of the EfficientViTBlock. The combination of [MBC, EVB] achieves the highest accuracy, with reduced latency and parameters compared to the original [EVB, EVB]. Therefore, the combination of [MBC, EVB] is the more suitable option.

MBConv employs the SE [36] block to assign different weights to positions in the image based on the channel domain. This demonstrates great adaptability and can be applied to various deep-learning tasks. However, the SE block only considers encoding channel spatial information while neglecting the spatial positional information of feature maps. This limitation poses challenges in fruit recognition tasks. In this work, the ELA module was adopted to replace the SE attention mechanism within the MBConv structure. The result is a module called E-MBConv (Fig. 6), which demonstrates improved performance. ELA employs stripe pooling in the spatial dimension to capture long-range dependencies, replacing 2D convolutions with 1D convolutions in both horizontal and vertical directions to process sequential signals in channels. This not only endows E-MBConv with more precise positional attention but also reduces the model's parameters. With the optimal module combination of [MBC, EVB] previously, the initial MBConv module in stages 3 and 4 of the backbone was substituted with the E-MBConv module. This replacement enabled an evaluation of the impact of the SE and ELA attention mechanisms on performance (Table 3).

Table 3 shows that replacing the SE block in MBConv with ELA significantly reduces model parameters while improving detection accuracy. Therefore, ELA outperforms SE in terms of performance.

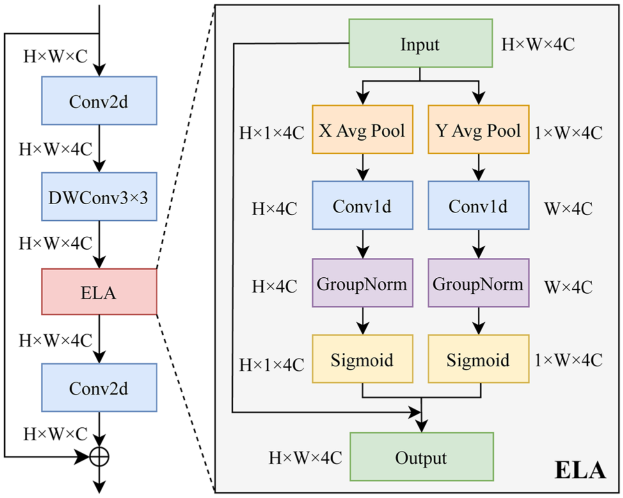

Fig. 6 The structure diagram of E-MBConv, obtained by replacing the SE in MBConv with ELA

Table 3 Based on the combination of [MBC, EVB], the performance comparison of models using MBConv and E-MBConv

Module	Top-1 mAP (%)	Latency (ms)	Params (M)
MBConv	82.33	13.39	75.994
E-MBConv	83.56 (+1.23)	14.34	34.192

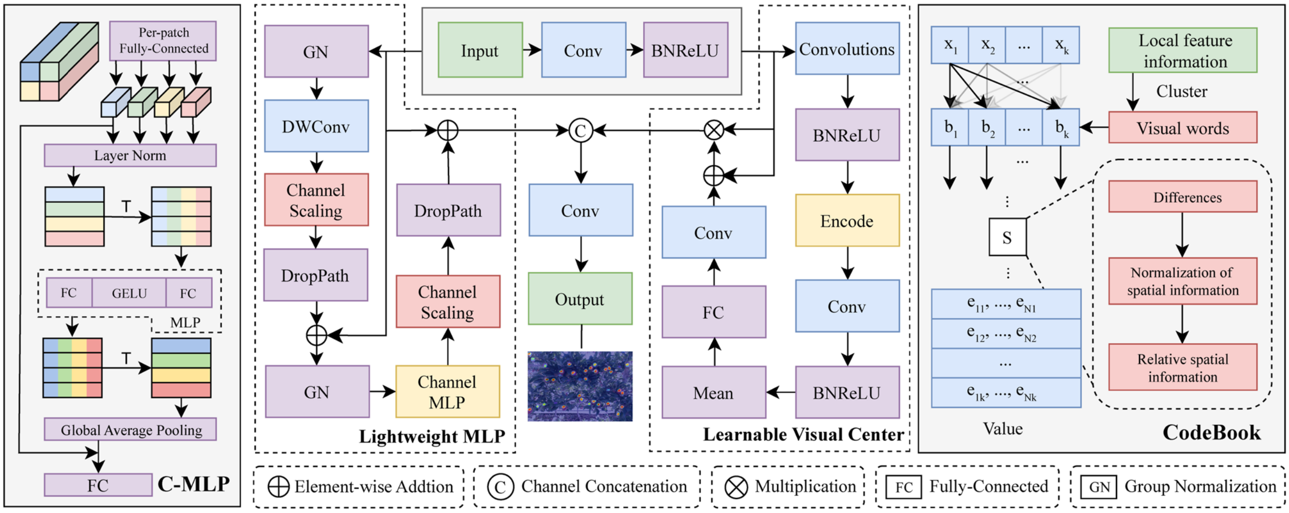

Fig. 7 The structure diagram of EVC

3.3 Neck network improvement and optimization

3.3.1 Centralized feature pyramid

The feature pyramid is a fundamental network structure used to detect targets with different scales and effectively leverage various scale feature information. In YOLOX, PAFPN serves as the neck network, while ESC-YOLO utilizes CFPNet for improved feature fusion. Compared to PAFPN, CFPNet has a broader receptive field, more comprehensive feature interaction, and a stronger emphasis on local information. It not only captures global long-range dependencies but also efficiently acquires comprehensive and distinctive features.

In CFPNet, the top-down and bottom-up feature paths are similar to PAFPN. However, different from PAFPN directly transmits and integrates global information at different scales, CFPNet initially extracts the most abundant global information and easily lost local information separately using an Explicit Visual Center (EVC) (Fig. 7). Following dimension concatenation, Global Centralized Regulation (GCR) is implemented to normalize cross-layer features across the entire feature pyramid. This allows the feature pyramid to demonstrate spatial weights that reflect global information at various scales, achieving comprehensive and discriminative feature representation. The Lightweight Multilayer Perceptron (MLP) is employed to capture global information. The first part achieves spatial information fusion on different dimensions of input feature maps through Depthwise Separable Convolution (DWConv) [37], while the second part achieves channel-wise information fusion via the Channel MLP [38]. The combination of these two parts completes the capture of global long-range dependencies, thereby enhancing the model's generalization capability and robustness under channel scaling and stochastic depth-wise dropout operations. The Learnable Visual Center (LVC) is used to focus on the local features. Firstly, a series of convolutional layers is utilized to encode the input feature maps. A group of trainable scale factors is set in the dictionary to adjust the weights of the encoded features, continuously fitting with a fixed codebook. Subsequently, a fully connected layer and a 1 × 1 convolutional layer are employed to predict the salient features. The output from these layer structures is used for channel-wise multiplication, addition with the original input, and extraction of local regional features.

The neck networks of YOLOX-S respectively employ PAFPN and CFPNet to analyze performance changes (Table 4). Serving CFPNet as the neck network exhibits a 1.88% improvement in Top-1 mAP compared to PAFPN. This increase in accuracy is attributed to CFPNet's capability to focus on both global and local features. However, due to the more comprehensive feature fusion conducted by CFPNet, there was a slight increase in both parameters and latency.

Table 4 Performance comparison of YOLOX-S neck network using PAFPN and CFPNet respectively

Neck network	Top-1 mAP (%)	Latency (ms)	Params (M)
PAFPN	80.41	9.54	8.951
CFPNet	82.29 (+1.88)	11.44	13.155

3.3.2 Parameter-free attention mechanism

SimAM is a module that evaluates the importance of each neuron by measuring the linear separability between neurons, aiming to enhance the attention mechanisms. First, calculate the differences d of each element in the input features X as Eq. (8), where n, c, h, and w represent the batch size, channels, height, and width of a tensor, respectively. $\overline{X_{n,c}}$ represents the calculation of the mean on dimensions h and w for channel c.

$$d_{n,c,h,w} = \left(X_{n,c,h,w} - \overline{X_{n,c}}\right)^2 \tag{8}$$

Then, compute the variance v of each channel.

$$v_{n,c} = \frac{1}{N \times H \times W - 1} \sum_{h=1}^{H} \sum_{w=1}^{W} d_{n,c,h,w} \tag{9}$$

Calculate the attention weights E_{inv} and model the importance of each pixel as Eq. (10), where λ is a configurable constant.

$$E_{\mathrm{inv}n,c,h,w} = \frac{d_{n,c,h,w}}{4\left(v_{n,c} + \lambda\right)} + 0.5 \tag{10}$$

Finally, perform element-wise multiplication between the input features X and the attention weights E_{inv} to obtain the adjusted features Y. Here, $\delta(\cdot)$ represents the sigmoid function.

$$Y_{n,c,h,w} = X_{n,c,h,w} \times \delta\left(E_{\mathrm{inv}n,c,h,w}\right) \tag{11}$$

Compared to traditional one-dimensional and two-dimensional attention mechanisms, SimAM combines spatial and channel attention to facilitate the selection of information during visual processing. SimAM does not introduce additional parameters and is a parameter-free attention mechanism, which is friendly and effective in reducing model parameters and computational cost.

4 Experimental results analysis

4.1 Backbone ablation experiment

In this work, improvements were made to the backbone feature extraction network and the neck-enhanced feature extraction network of YOLOX. The neck network was enhanced by replacing PAFPN with CFPNet, which achieves a more comprehensive feature fusion. EfficientViT employs three strategies to optimize and serve as the backbone network of ESC-YOLO. First, it incorporates the SCConv module into the ResBlock of the input stem. Second, it utilizes the optimal combination of MBConv and EfficientViT-Block in stages 3 and 4. Third, it replaces the SE module in MBConv with ELA.

To assess the impact of the three backbone network optimization strategies on model performance, the enhanced results of the combination were analyzed through a backbone network ablation experiment (Table 5). The models were trained until convergence with modifications made solely to the backbone network while maintaining the overall structure and parameter settings unchanged.

Comparing experiment Methods 1, 2 and 7, it shows that the single SCConv module has limited impact on the overall model performance, with only a 0.04% improvement. However, when the reconstructed spatial and channel features receive the focus of ELA, there is a more comprehensive utilization of spatial-channel information, resulting in a 1.03% increase in Top-1 mAP and a significant improvement in performance. In Methods 1 and 4, replacing the SE attention mechanism with ELA significantly reduces the model's parameters. Furthermore, when compared to solely enhancing local attention in the backbone network, the joint attention of ELA and CFPNet on both the backbone and neck networks facilitates the extraction of local feature information. This joint attention also enhances the flow and coherence of information within the model, promotes feature fusion and collaborative learning, ultimately resulting in a Top-1 mAP increase up to 83.65%. In Methods 1 and 3, the combination of [MB, EVB] shows superior performance, smaller parameters, and faster inference speed compared to the original configuration. According to Method 8, the improved EfficientViT has advantages in all three optimization strategies. These strategies complement each other, significantly enhancing the performance of ESC-YOLO and promoting the effective utilization of spatial and channel features. As a result, the model achieves a Top-1 mAP of 84.82%.

Table 5 Ablation study on the improved model's backbone network. Here, **A** represents the utilization of RSBlock, **B** represents the use of the optimal combination of [MBC, EVB], and **C** represents using E-MBConv as the MBC in the combination

Method	A	B	C	Top-1 mAP (%)	Latency (ms)	Params (M)
1	×	×	×	82.68	15.91	80.198
2	√	×	×	82.72 (+0.04)	17.75	80.199
3	×	√	×	83.14 (+0.46)	13.38	78.710
4	×	×	√	83.65 (+0.97)	16.17	38.396
5	√	√	×	83.60 (+0.92)	14.82	78.712
6	×	√	√	84.29 (+1.61)	13.97	36.908
7	√	×	√	83.71 (+1.03)	18.26	38.398
8	√	√	√	84.82 (+2.14)	14.92	36.910

Table 6 Comparison of performance with different attention mechanisms embedded in the neck network of ESC-YOLO

Attention module	Top-1 mAP (%)	Latency (ms)	Params (M)
Base	84.82	14.92	36.910
+ Coord. A [39]	84.76 (− 0.06)	15.43	36.946
+ CBAM [40]	85.01 (+ 0.25)	16.44	36.932
+ SimAM	84.93 (+ 0.17)	15.28	36.910

4.2 Comparative experiment of attention mechanisms

In this work, attention mechanism modules are embedded at the outputs of Layers 1, 2 and 3 in the neck network of ESC-YOLO. This improvement aims to enhance the model's focus on input features while improving its performance, generalization ability, and interpretability.

According to Table 6, the comparison of attention mechanisms embedded in the neck network shows that their inclusion has a noticeable impact on directing feature extraction towards specific regions and improving network performance. The CBAM demonstrates strong performance in enhancing network accuracy, but it also leads to a significant increase in model inference time. In contrast, the SimAM does not introduce additional parameters and shows excellent performance in both inference speed and accuracy, achieving a balanced performance.

To clearly illustrate the model's focus on different regions, this work employs heatmaps to visualize the attention mechanisms. Models without weights are trained for 10 epochs under a consistent setting and the visualization results are shown in Fig. 8. After embedding attention mechanisms into the model, the improved models exhibit heightened focus on target fruits in orchard environments characterized by direct sunlight exposure, dense fruit distribution, and no occlusion. Compared to ESC-YOLO without attention mechanisms and those embedding CA or CBAM, SimAM demonstrated a higher degree of attention to primary regions and local features, facilitating improved learning and classification. Furthermore, SimAM demonstrates an improvement in reducing missed detections, demonstrating greater versatility and robustness. Therefore, it exhibits superior performance.

4.3 ESC-YOLO vs. YOLOX performance comparison

4.3.1 Feature map visualization

Feature map visualization is a method used to visualize the intermediate feature maps of deep neural networks. As convolutional neural networks increase in depth, the size of the image decreases and the contained information becomes more abstract. This work extracts three different-sized and dimensioned feature maps from stages 2, 3 and 4 of ESC-YOLO-S, as well as the output of the YOLOX backbone network with depths of 128, 256, and 512. This

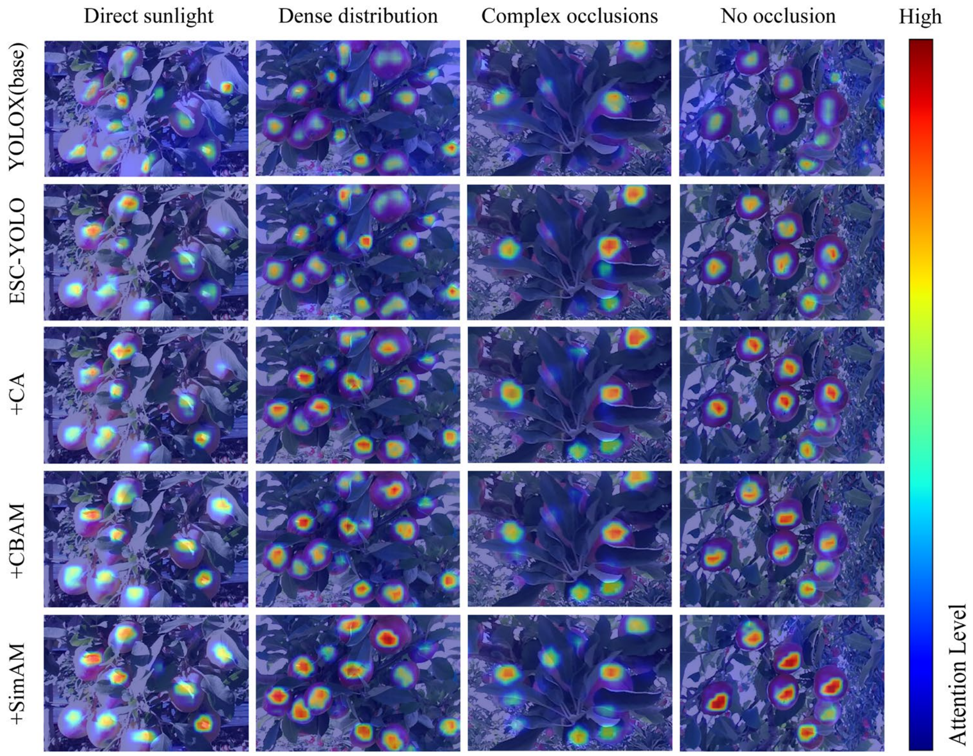

Fig. 8 Comparison of heatmap visualization includes YOLOX (base), ESC-YOLO, and ESC-YOLO with different attention mechanisms embedded in CFPNet

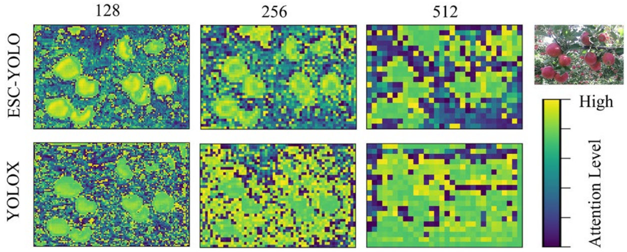

Fig. 9 ESC-YOLO and YOLOX learn the information about Large-area fruits in channel 128, 256 and 512

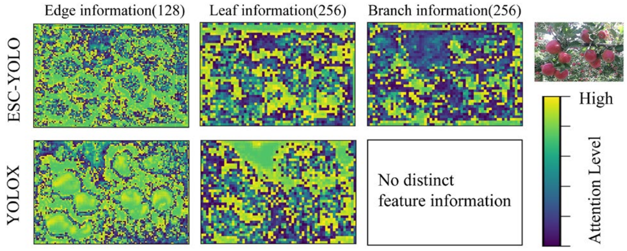

Fig. 10 ESC-YOLO and YOLOX learn the information about different types of features

method helps observe the learning behavior and focus of the two models on different information and features.

In Figs. 10 and 11, a portion of the image containing a clearly defined main branch is selected. The marked bounding boxes of all the fruits in this area are labeled as Large. This input image contains clear and rich local details, which are used to analyze the detection and attention capabilities of the two models regarding local details. Figure 11 randomly selects an image from the dataset, which is used to analyze the detection and attention capabilities of the two models regarding global information.

Figure 9 illustrates that ESC-YOLO demonstrates a more precise focus on fruit information at the dimension of 128. Even at deeper dimensions, ESC-YOLO maintains its attention on fruit information, while YOLOX's focus on fruit information becomes dispersed and mixed with attention to other features.

The information regarding occlusion types includes details about fruit edges, leaves, and branches. Accurate identification of occlusion types requires models to effectively distinguish and learn these features.

Figure 10 reveals that ESC-YOLO accurately identifies the contours of fruits, while YOLOX's contour ranges are blurred and contain fruit information. In terms of leaf recognition, ESC-YOLO effectively identifies and focuses on the leaves near the fruits, and successfully segments the areas containing both fruits and leaves, even in cases where occlusion may occur. In contrast, YOLOX only achieves blurry recognition in certain areas. In terms of branch information, ESC-YOLO maintains attention to the tiny local details of branches within various feature information. In contrast, YOLOX lacks distinct branch feature information, which is a major factor in false detections.

In Fig. 11, ESC-YOLO accurately identifies the positions of fruit, clarifying the layout and interrelation between fruits and leaves at dimension 256. In contrast, YOLOX's feature map is chaotic, with blurred attention regions. This demonstrates ESC-YOLO's significant advantage in detecting the global information and local details, learning various types of feature information, and attending to global context compared to YOLOX. Both in terms of local or global detection, ESC-YOLO outperforms YOLOX.

4.3.2 Visualization of test results

To assess ESC-YOLO's performance in fruit detection and occlusion recognition, we selected samples from the test set with different fruit distributions and environmental conditions. The selected samples include: a. abundant and densely distributed fruits, b. sparsely distributed fruits with complex occlusions, and c. direct sunlight and shadows from branches and leaves.

Figure 12 demonstrates that ESC-YOLO accurately locates and identifies the spatial distribution of fruits, regardless of density. In comparison, YOLOX shows lower accuracy with a higher rate of false and missed detections. ESC-YOLO maintains clear classification and higher confidence levels even under special lighting conditions. It also exhibits more precise spatial object recognition and an accurate grasp of local detailed features compared to YOLOX. Additionally, ESC-YOLO outperforms YOLOX in occlusion recognition, accurately identifying occlusions in various scenarios where YOLOX fails to do so. Therefore, ESC-YOLO outperforms YOLOX in both detection accuracy and occlusion recognition.

To further compare the performance differences between the two models, two indicators are used for the data analysis of four different occlusion types. The definitions of False Detection Rate (FDR) and Missed Detection Rate (MDR) are provided in Eqs. (12) and (13) to quantify the occurrence

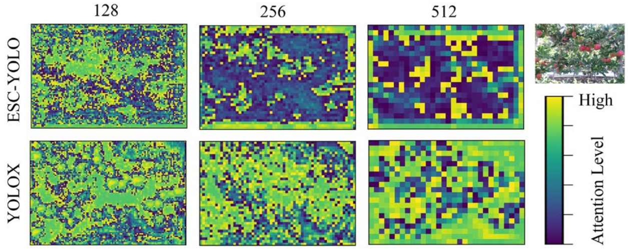

Fig. 11 ESC-YOLO and YOLOX learn the global information about Medium-area in channel 128, 256 and 512

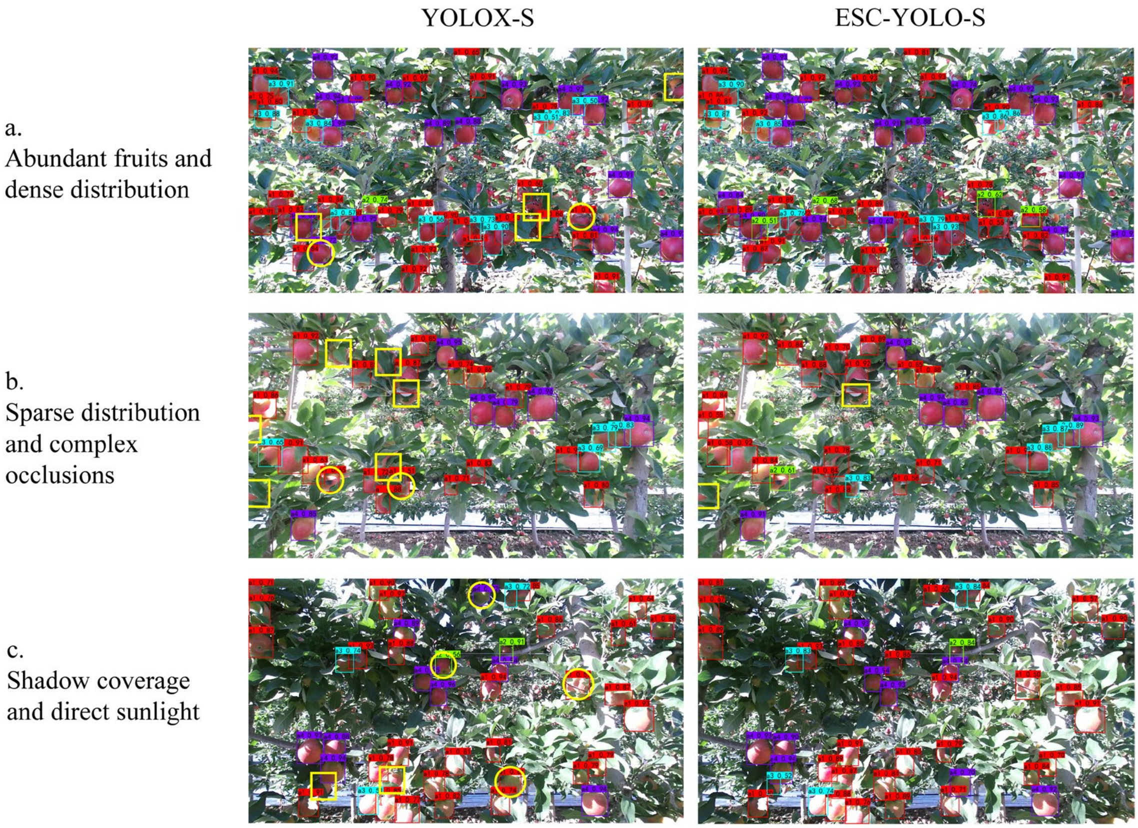
YOLOX-S
ESC-YOLO-S
a.
Abundant fruits and dense distribution
b.
Sparse distribution and complex occlusions
c.
Shadow coverage and direct sunlight

Fig. 12 Comparison of fruit recognition performance between YOLOX-S and ESC-YOLO-S. False detections are indicated by yellow circles, while missed detections are indicated by yellow boxes

Table 7 Comparison of FDR and MDR between the original and the improved models

Model	FDR (%)				MDR (%)			
	*a*1	*a*2	*a*3	*a*4	*a*1	*a*2	*a*3	*a*4
YOLOX-S	18.34	27.81	18.73	16.94	10.08	51.50	15.95	7.80
ESC-YOLO-S (ours)	16.34	21.14	18.54	15.12	9.97	39.00	11.11	6.80

Here, **a1–a4** correspond to different types of occlusions

of false and missed detections in fruit recognition models. Here, N_{ai} represents the number of ground truth bounding boxes for each class, FN_{ai} represents False Negatives for each class and TP_{ai} represents True Positives for each class.

$$FDR_{ai} = \frac{FN_{ai}}{N_{ai}}, \quad i = 1, 2, 3, 4 \tag{12}$$

$$MDR_{ai} = 1 - FDR_{ai} - \frac{TP_{ai}}{N_{ai}}, \quad i = 1, 2, 3, 4 \tag{13}$$

The smaller the values of FDR and MDR, the lower the frequency of false and missed detections, indicating superior performance of the fruit recognition model. According to Table 7, ESC-YOLO achieved lower FDR and MDR for all four types of occlusions. Particularly noteworthy is its significant improvement in detecting type a2 occlusions. It can be inferred that ESC-YOLO has the capability to identify small-scale objects and features. There has been an improvement in model performance due to a significant reduction in false and missed detections. According to Table 8, our ESC-YOLO exhibits superior accuracy and generality in detecting multi-scale objects. Compared with YOLOv7-L, YOLOv8-S, YOLOv8-L, Gold-YOLO-S, and RT-DETR-R50, ESC-YOLO achieves improvements of 5.89%, 5.56%, 3.48%, 3.31%, and 0.38% in Top-1 mAP$_{0.5}$. Whether it is the horizontal comparison of the *S* models of YOLO or the horizontal comparison of models with similar parameters, ESC-YOLO significantly enhances the detection accuracy of multi-scale objects under the criterion of real-time detection. Compared to the original YOLOX-S model, ESC-YOLO-S demonstrates improvements of 4.26%, 2.96%, 4.10%, and

Table 8 Comprehensive comparison of model performance

Model	$mAP_{0.5}$ (%)	$mAP_{0.5:0.95}$ (%)	mAP_M (%)	mAP_L (%)	Latency (ms)	Params (M)
YOLOX-S	80.67	62.89	59.36	69.83	9.54	8.939
YOLOv7-L	79.04	57.22	54.71	65.93	12.53	37.216
YOLOv8-S	79.37	62.31	60.01	69.25	8.45	11.138
YOLOv8-L	81.45	65.25	63.06	73.59	14.77	43.634
Gold-YOLO-S [41]	81.62	64.73	62.18	72.03	6.33	21.541
RT-DETR-R50 [42]	84.55	66.06	63.39	74.31	13.26	42.707
ESC-YOLO-S (Ours)	84.93	65.85	63.46	74.28	15.28	36.910

Here, **L** and **M** in **mAP** represent the size of the region for detecting targets, **0.5** and **0.5:0.95** in **mAP** are the thresholds of IoU when calculating mAP, **S** and **L** in **Model** denote the models of YOLO detectors, and **R50** indicates the backbone of RT-DETR. All **mAP** are adopted with the Top-1 values

4.45% in Top-1 $mAP_{0.5}$, $mAP_{0.5:0.95}$, mAP_M, and mAP_L respectively. This confirms that the improved model is more effective in capturing global information and implies higher precision and robustness in recognizing local features.

5 Conclusion

This work presents an enhanced ESC-YOLO detection model that fully leverages and emphasizes spatial channel information. The model achieves precise classification learning of multi-type and multi-scale features through a more comprehensive fusion of high fluidity and coherence feature information. The model emphasizes the integrity of multi-scale features and achieves fruit recognition under complex occlusion based on multi-level semantic information, such as global information and local features. Compared to YOLOX, ESC-YOLO exhibits higher learning efficiency and faster convergence speed, with improvements of 4.26%, 2.96%, 4.10%, and 4.45% in Top-1 $mAP_{0.5}$, $mAP_{0.5:0.95}$, mAP_M, and mAP_L, respectively. Moreover, ESC-YOLO significantly improves the false and missed detections under all types of occlusions, especially reducing the FDR and MDR of strip-shaped occlusions by 6.67% and 12.50%. This work meets the demand for high-precision fruit recognition in complex orchard environments and provides new insights into the application of real-time fruit recognition, thereby assisting in automated harvesting decisions and offering effective technical support for the automation of apple picking.

Author contributions Jun Sun: Conceptualization, Supervision, Resource. Yifei Peng: Writing – original draft, Software, Methodology. Chen Chen: Writing – review & editing. Bing Zhang: Supervision. Zhaoqi Wu: Writing – review & editing. Yilin Jia: Data Curation. Lei Shi: Methodology.

Funding Funding was provided by the National Natural Science Foundation of China (Grant No. 32102598) and Project of Postgraduate Research & Practice Innovation Program of Jiangsu Province (Grant No. SJCX24_2390).

Data availability No datasets were generated or analysed during the current study. The Apple dataset and executable files used in this paper will be available upon request at: https://github.com/fu3lab/Scifresh-apple-RGB-images-with-multi-class-label.

Declarations

Conflict of interests The authors declare no competing interests.

References

1. Fu, L., Gao, F., Wu, J., Li, R., Karkee, M., Zhang, Q.: Application of consumer RGB-D cameras for fruit detection and localization in field: a critical review. Comput. Electron. Agric. **177**, 105687 (2020)
2. Zhang, Z., Igathinathane, C., Li, J., Cen, H., Lu, Y., Flores, P.: Technology progress in mechanical harvest of fresh market apples. Comput. Electron. Agric. **175**, 105606 (2020)
3. Wang, D., Song, H., He, D.: Research advance on vision system of apple picking robot. Trans. Chin. Soc. Agric. Eng. **33**(10), 59–69 (2017)
4. Zhang, C., Kang, F., Wang, Y.: An improved apple object detection method based on lightweight YOLOv4 in complex backgrounds. Remote Sens. **14**(17), 4150–4150 (2022)
5. Divyanth, L.G., Rathore, D., Senthilkumar, P., Patidar, P., Zhang, X., Karkee, M., Machavaram, R., Soni, P.: Estimating depth from RGB images using deep-learning for robotic applications in apple orchards. Smart Agric. Technol. **6**, 100345 (2023)
6. Wu, L., Ma, J., Zhao, Y., Liu, H.: Apple detection in complex scene using the improved YOLOv4 model. Agronomy **11**(3), 476 (2021)
7. Sun, J., Yang, K., Chen, C., Shen, J., Yang, Y., Wu, X., Tomas, N.: Wheat head counting in the wild by an augmented feature pyramid networks-based convolutional neural network. Comput. Electron. Agric. **193**, 106705 (2022)
8. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: single shot multibox detector. In: Computer Vision-ECCV 2016: 14th European Conference, Amsterdam, Proceedings, Part I, pp. 21–37. The Netherlands, October 11–14 (2016)

9. Bochkovskiy, A., Wang, C.-Y., Liao, H.-Y.M.: YOLOv4: optimal speed and accuracy of object detection (2020). arXiv preprint arXiv:2004.10934
10. Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., Nie, W., Li, Y., Zhang, B., Liang, Y., Zhou, L., Xu, X., Chu, X., Wei, X., Wei, X.: YOLOv6: a single-stage object detection framework for industrial applications (2022). arXiv preprint, arXiv:2209.02976
11. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake, USA (2018)
12. Wang, C.-Y., Bochkovskiy, A., Liao, H.-Y. M.: YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7464–7475 (2023)
13. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587 (2014)
14. Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
15. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2015)
16. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2961–2969 (2018)
17. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J.: YOLOX: exceeding YOLO series in 2021 (2021). arXiv preprint arXiv:2107.08430
18. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8759–8768 (2018)
19. Liu, X., Peng, H., Zheng, N., Yang, Y., Hu, H., Yuan, Y.: EfficientViT: memory efficient vision transformer with cascaded group attention. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14420–14430 (2023)
20. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
21. Li, J., Wen, Y., He, L.: SCConv: spatial and channel reconstruction convolution for feature redundancy. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
22. Xu, W., Wan, Y.: ELA: efficient local attention for deep convolutional neural networks (2024). arXiv preprint, arXiv:2403.01123
23. 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), pp. 4510–4520 (2018)
24. Tan, M., Le, Q.: EfficientNet: rethinking model scaling for convolutional neural networks. In: Proceedings of the 36th International Conference on Machine Learning, PMLR, vol. 97, pp. 6105–6114 (2019)
25. Quan, Y., Zhang, D., Zhang, L., Tang, J.: Centralized feature pyramid for object detection. IEEE Trans. Image Process. **32**, 4341–4354 (2022)
26. Yang, L., Zhang, R.-Y., Li, L., Xie, X.: SimAM: a simple, parameter-free attention module for convolutional neural networks. In: Proceedings of the 38th International Conference on Machine Learning, pp. 11863–11874 (2021)
27. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: a metric and a loss for bounding box regression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 658–666 (2019)
28. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization (2017). arXiv preprint, arXiv:1412.6980
29. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: transformers for image recognition at scale (2020). arXiv preprint, arXiv:2010.11929
30. Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., Xie, S.: A ConvNet for the 2020s. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11976–11986 (2022)
31. Tan, M., Le, Q.: EfficientNetV2: smaller models and faster training. In: Proceedings of the 38th International Conference on Machine Learning, PMLR, vol. 139, pp. 10096–10106 (2021)
32. Vasu, P.K.A., Gabriel, J., Zhu, J., Tuzel, O., Ranjan, A.: MobileOne: an improved one millisecond mobile backbone. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7907–7917 (2023)
33. Wang, A., Chen, H., Lin, Z., Han, J., Ding, G.: RepViT: revisiting mobile CNN from ViT perspective. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 15909–15920 (2024)
34. Xie, S., Girshick, R., Dollar, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1492–1500 (2017)
35. Wu, Y., He, K.: Group normalization. In: Proceedings of the European Conference on Computer Vision, pp. 3–19 (2018)
36. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
37. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: MobileNets: efficient convolutional neural networks for mobile vision applications. arXiv preprint, arXiv:1704.04861 (2017)
38. Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Keysers, D., Uszkoreit, J., Lucim, M.:. MLP-mixer: an all-MLP architecture for vision (2021). arXiv preprint, arXiv:2105.01601
39. Hou, Q., Zhou, D., Feng, J.: Coordinate attention for efficient mobile network design. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2021)
40. Woo, S., Park, J., Lee, J.-Y., Kweon, I.S.: CBAM: convolutional block attention module. In: Proceedings of the European Conference on Computer Vision, pp. 3–19 (2018)
41. Wang, C., He, W., Nie, Y., Guo, J., Liu, C., Han, K., Wang, Y.: Gold-YOLO: efficient object detector via gather-and-distribute mechanism. In: Advances in Neural Information Processing Systems, vol. 36 (2023)
42. Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., Chen, J.: DETRs beat YOLOs on real-time object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 16965–16974 (2024)