Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.061743

ARTICLE

Double Self-Attention Based Fully Connected Feature Pyramid Network for Field Crop Pest Detection

Zijun Gao*, Zheyi Li, Chunqi Zhang, Ying Wang and Jingwen Su

School of Information Science and Engineering, Dalian Polytechnic University, Dalian, 116034, China

*Corresponding Author: Zijun Gao. Email: gaozj@dlpu.edu.cn

Received: 02 December 2024; Accepted: 11 March 2025; Published: 19 May 2025

ABSTRACT: Pest detection techniques are helpful in reducing the frequency and scale of pest outbreaks; however, their application in the actual agricultural production process is still challenging owing to the problems of inter-species similarity, multi-scale, and background complexity of pests. To address these problems, this study proposes an FD-YOLO pest target detection model. The FD-YOLO model uses a Fully Connected Feature Pyramid Network (FC-FPN) instead of a PANet in the neck, which can adaptively fuse multi-scale information so that the model can retain small-scale target features in the deep layer, enhance large-scale target features in the shallow layer, and enhance the multiplexing of effective features. A dual self-attention module (DSA) is then embedded in the C3 module of the neck, which captures the dependencies between the information in both spatial and channel dimensions, effectively enhancing global features. We selected 16 types of pests that widely damage field crops in the IP102 pest dataset, which were used as our dataset after data supplementation and enhancement. The experimental results showed that FD-YOLO's mAP@0.5 improved by 6.8% compared to YOLOv5, reaching 82.6% and 19.1%–5% better than other state-of-the-art models. This method provides an effective new approach for detecting similar or multiscale pests in field crops.

KEYWORDS: Pest detection; YOLOv5; feature pyramid network; transformer; attention module

1 Introduction

Pest identification plays a crucial role in agricultural production, as pests account for a significant portion of global annual food losses [1]. Effective pest control is essential for enhancing agricultural productivity, ensuring food security, and safeguarding farmers' incomes. By implementing efficient pest monitoring systems, it is possible to achieve substantial results with reduced effort. Early and accurate detection and identification of pest populations are vital for managing their numbers, understanding their occurrence patterns, and implementing effective prevention and control measures. However, challenges such as environmental complexity and the high similarity among pest species [2] complicate the pest detection process, making it a challenging area of research.

Traditional pest identification techniques often rely on the expertise of trained taxonomists who identify pests based on morphological characteristics. This approach is labor-intensive, time-consuming, costly, and inefficient, often resulting in delayed feedback [3]. The advent of computer vision technology and machine learning algorithms has led to advancements in automated pest detection. Nevertheless, conventional machine learning methods depend on manually designed feature extractors, which tend to maintain stable detection accuracy only when dealing with a limited number of pest samples or a single pest species. When

faced with large-scale, multi-category pest data in practical applications, these methods often exhibit reduced accuracy and robustness [4].

In recent years, deep learning algorithms have emerged as efficient and accurate models for addressing computer vision tasks related to the identification and detection of agricultural pests. Among these, two-stage algorithms generate a series of object bounding boxes in the first stage, followed by classification in the second stage using convolutional neural networks. In contrast, one-stage algorithms directly convert the bounding box prediction problem into a regression task, simultaneously generating class probabilities and coordinates for detection. Consequently, while two-stage algorithms are known for their high accuracy, they tend to be slower compared to their one-stage counterparts [5]. For instance, Rong et al. [6] enhanced the feature pyramid structure of the Mask R-CNN model to improve recognition accuracy for small targets, providing a novel approach for pest identification and counting on yellow plates in the field. Wang et al. [7] employed a sensitive score matrix to enhance bounding box prediction performance for detecting small rice flies in rice fields, achieving an average accuracy of 81%. Ali et al. [8] proposed the Faster-PestNet method, which utilizes Faster R-CNN with MobileNet as a backbone network and depth-separable convolution to recognize multiple pest classes.

With the evolution of one-stage algorithms, some have surpassed two-stage algorithms in performance while also being more conducive to real-time applications. The YOLO [9] family exemplifies a one-stage detection algorithm that effectively balances detection accuracy and speed. Liu et al. [1] integrated a triple attention mechanism into a CNN network, resulting in an improved YOLOv4 model for tomato pest detection, achieving a mean Average Precision (mAP) of 95.2%. This model outperformed Faster R-CNN, SSD, the original YOLOv4, and YOLOv3 in terms of detection speed, computational cost, model size, and accuracy. Zhang et al. [10] introduced an enhanced YOLOv5 network that combines Bi-FPN feature fusion, channel attention, and a DenseNet backbone to accurately identify unopened bolls on cotton crops, aiding farmers in optimizing mechanized cotton harvesting and reducing crop loss while predicting yield. Wen et al. [11] developed Pest-YOLO based on YOLOv4, utilizing a focal loss function to enhance the loss weights of challenging samples and employing a confluence strategy to minimize the omission of tiny, dense pests, achieving a mAP of 69.6% on the Past24 dataset. Hu et al. [12] utilized GC Attention and Swin Transformer to enhance global sensing and feature extraction capabilities, employing Bi-FPN for feature fusion and adding detection heads to capture more scale features, thereby improving rice pest detection accuracy. Yin et al. [13] incorporated a multibranch CBAM structure into YOLOv8, introduced a minimum point distance intersection and merger ratio as a bounding box loss indicator, and utilized Ghost convolution to enhance the model's efficiency, facilitating automatic detection and counting of rice pests. Dong et al. [14] developed the PestLite model based on the YOLOv5 algorithm, introducing ECA attention to focus on multi-scale contextual information through multi-level spatial pyramid pooling and replacing up-sampling with feature content-aware restructuring, achieving a mAP@50 of 57.1% on the IP102 dataset, representing a 2.6% improvement over the baseline.

Despite the advancements in agricultural pest detection models, several significant challenges remain. The vast number of insect species and the diverse appearances of different pest species complicate detection efforts. Many existing pest detection algorithms are limited to specific pest categories with fewer instances, resulting in poor generalization and hindering predictions of pest outbreaks. Additionally, the high similarity in color and appearance among certain pests complicates feature extraction during model training. Furthermore, the substantial scale differences in pest images and the protective coloration of pests can lead to confusion or inaccurate localization against the background, significantly limiting the performance of pest detection algorithms.

To address these challenges, this paper builds upon the YOLOv5 baseline model by incorporating an attention mechanism to enhance feature extraction and replacing the neck Feature Pyramid Network with a multi-scale feature fusion approach. This results in the design of the FD-YOLO model, aimed at detecting pest targets in field crops characterized by interspecies similarity, multi-scale variations, and complex background environments. The main contributions and innovations of this paper are as follows:

(1) The introduction of the FC-FPN, which connects input feature layers at all scales with output feature layers at all scales, enabling adaptive fusion of multi-scale features and enhancing the multiplexing of effective features.
(2) The proposal of the DSA module, embedded within the C3 module to form DSA-C3, which captures spatial relationships between features and channel relationships, effectively enhancing global feature representation.
(3) The selection of sixteen pest categories that are significantly damaging to field crops from the IP102 large-scale pest dataset, along with the collection and annotation of images for underrepresented categories to achieve data balance. Additionally, data augmentation techniques were employed to expand the dataset, thereby improving the model's generalization ability and robustness.

2 Materials and Methods

2.1 Datasets

The dataset utilized in this paper is derived from the IP102 dataset [15] and various web resources. The IP102 dataset is the most comprehensive pest dataset available, featuring the largest number of images and the widest variety of pest species. It comprises 18,975 images across 102 pest categories, which are organized into two major groups and eight subgroups. The primary groups include pests affecting field crops (such as rice, corn, wheat, sugar beet, and alfalfa) and pests impacting cash crops (including grapes, citrus, and mango).

For this study, we selected 16 types of pests that significantly affect field crops, resulting in a total of 3893 images. Due to factors such as population size and environmental conditions, the dataset exhibits a natural long-tailed distribution. Direct training on this dataset can severely impact practical applications, as it is challenging to extract sufficient features from the limited number of samples in the tail classes. Consequently, these tail classes are often overlooked, while the abundance of samples in the head classes tends to dominate the classifier training. This unbalanced distribution typically results in improved performance for the head classes and diminished performance for the tail class [2]. To address this issue, we collected and annotated 825 images from the internet to supplement the five tail categories in the dataset, ultimately creating a multi-class field crop pest dataset containing 4718 images.

Most images in this dataset feature real natural backgrounds that reflect actual field conditions. While this may lead to challenges in distinguishing targets from complex backgrounds, it significantly enhances the robustness and practicality of the model. Additionally, the dataset includes images depicting different life stages of pests. Although the appearances of larvae and adults of certain pest species can vary greatly, this diversity increases the recognition challenge while simultaneously enhancing the model's generalization ability. Fig. 1 illustrates some images of inter-species similar pests.

Deep learning models require extensive and varied data to make accurate predictions across different environments. Data augmentation techniques play a crucial role in enriching the dataset by generating multiple variants of existing data, thereby providing more training material and enabling the model to extract and utilize features more effectively. In this paper, we employed the Mosaic data augmentation method, applying transformations such as HSV adjustments, rotation, translation, cropping, scaling, combination,

flipping, and perspective transformations to the dataset images. After augmentation, each image is composed of four randomly transformed images, enhancing the detection capabilities for small targets. Furthermore, these new samples encompass various target orientations, angles, and lighting conditions, allowing the model to learn a richer set of target features. This approach not only improves prediction accuracy but also enhances generalization ability, increases robustness, and mitigates overfitting—factors that are critical for effective training.

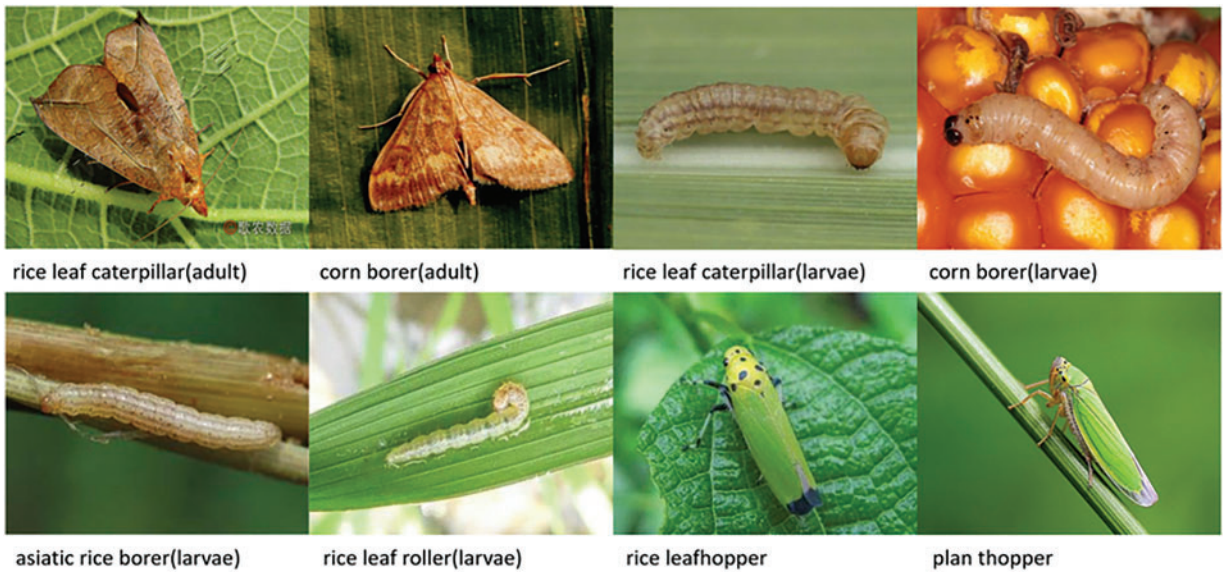

Figure 1: Some interspecific similar pests in the dataset

2.2 YOLOv5 Network Structure

The YOLO series algorithms divide the input image into small grids, generate the predicted bounding boxes by convolution operation, screen out the bounding boxes with low confidence using the non-maximum suppression (NMS) method, and then regress the bounding boxes with the highest confidence. Among them, the YOLOv5 architecture model not only has the advantages of fast speed and high accuracy, but is also widely used in many fields of computer vision because of its excellent ease of use, compatibility, and stability. YOLOv5 contains three key components: backbone, neck, and detection head. The backbone network performs feature extraction by stacking four C3 modules, which increase the width and depth of the network and better preserve image information to achieve the residual effect, as well as increase the receptive field to enhance the ability to focus on global information. Spatial Pyramid Pooling (SPP) is used at the end of the backbone to pool feature information at different scales to improve the accuracy and efficiency of the model. The neck uses a feature fusion module called Path Aggregation Network (PANet), which successively performs bottom-up and top-down multi-scale feature fusion to enhance the feature reuse of feature maps at each size, which helps to reduce the difficulty of parameter convergence and accelerate the learning speed of the model. Finally, the head was used as a prediction module to predict large, medium, and small targets using (80,80,256), (40,40,512) and (20,20,1024) feature maps of three different shapes with the category, confidence, and bounding box of the target. The structure of the YOLOv5 model is shown in Fig. 2.

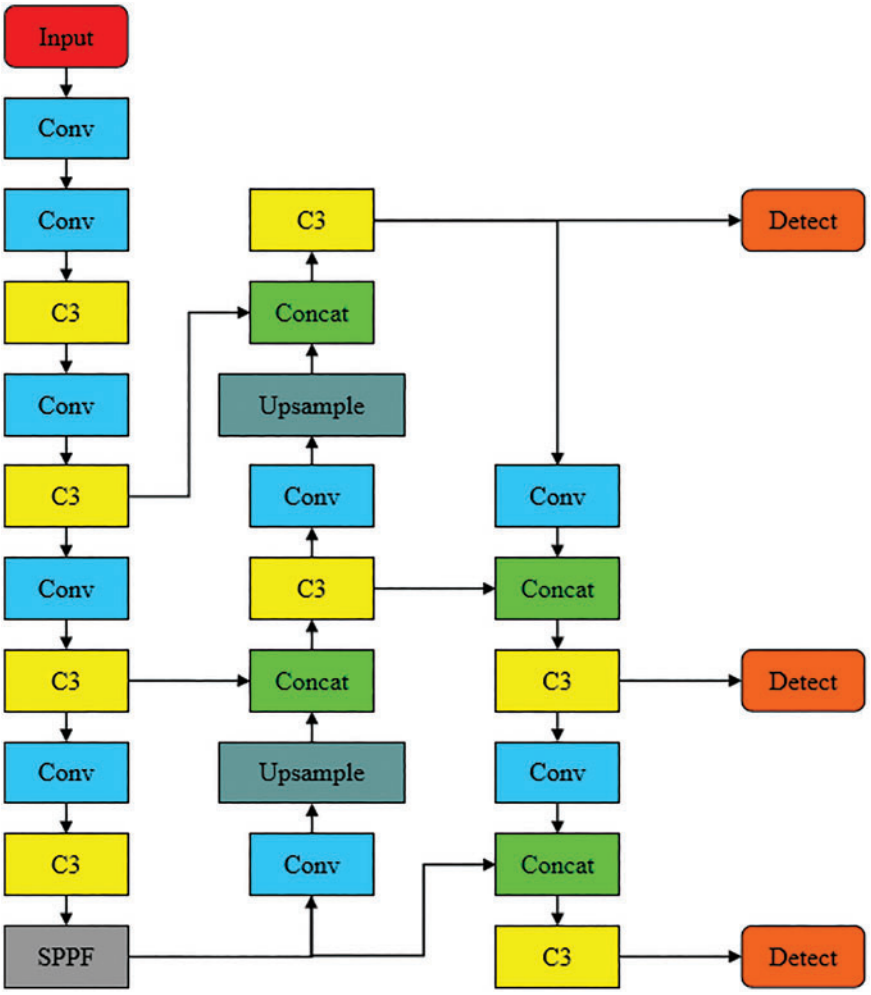

Figure 2: YOLOv5 network structure

2.3 FD-YOLO Network Structure

Due to the huge difference between pest detection and general target detection, most of the pest targets in the pest target detection task suffer from strong interspecies similarity, large scale difference or complex background, which leads to misdetection or omission. Therefore, this paper based on YOLOv5 proposes a FD-YOLO model for pest detection. Firstly, this paper proposes a Fully Connected FPN (FC-FPN) instead of PANet in YOLOv5, which fuses the input feature layers of all scales into the output feature layers of each scale, and performs feature extraction after each output feature layer. Then the Double Self-Attention (DSA) module is proposed, which is combined with the C3 module of the neck to enhance feature extraction. Finally, the FD feature fusion network is formed as the neck. The FD-YOLO model structure is shown in Fig. 3.

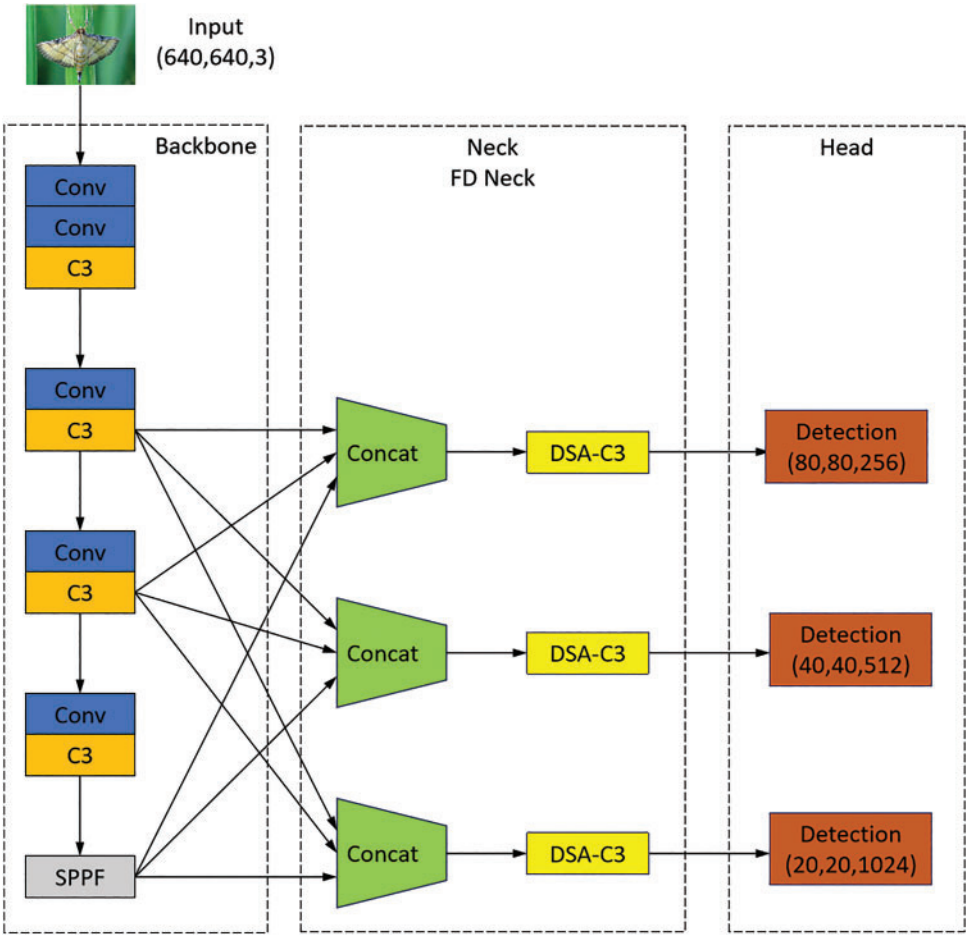

Figure 3: FD-YOLO network structure

2.4 FC-FPN

Multiscale feature fusion using the proposed FC-FPN replacement of PANet in the YOLOv5 neck. PANet was proposed in 2018 by Liu et al. [16]. By adding a bottom-up fusion path, the shallow information is more conveniently delivered to the top-level features through PANet rather than passing through all the backbone networks before reaching them, thus preserving more shallow texture information and position information. However, with the improvement of YOLOv5, the deeper CSPDarknet53 and ResNet backbone networks are replaced by a wider and lighter C3 module, and PANet becomes less convenient and efficient. Currently, BiFPN and ASFF are the widely used FPN networks. BiFPN adds fusion weights during feature fusion, which makes it easier for the network to learn appropriate information; however, using only adjacent layers is not conducive to capturing multi-scale information about the target. In addition, using bidirectional feature fusion and superimposing the BiFPN several times increases the complexity of the model to a certain degree, which is not conducive to network learning. ASFF adaptively fuses multiple backbone feature layers of different scales into multiple output feature layers of different scales for detection, which can directly transfer information between deep and shallow layers and simplify the network structure. However, a larger number of parameters are used during adaptive fusion, which increases the computational cost, and the detection

is performed directly without feature extraction after feature fusion, which results in a limit. Fig. 4 shows a schematic of the PANet, BiFPN, and ASFF network structures.

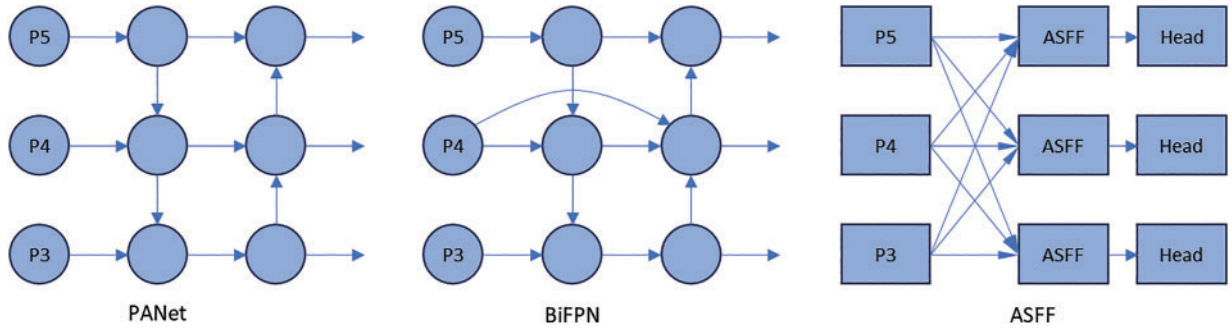

Figure 4: PANet, BiFPN and ASFF structure

The structure of our proposed FC-FPN is illustrated in Fig. 5. FC-FPN Downsampling or Upsampling all the three input feature maps into three sizes of 80 × 80, 40 × 40 and 20 × 20 feature maps, and then sets the trainable weights w to weight and fuse the feature maps of each size, and finally performs the feature extraction through the C3 module. The output equations of each layer of the FC-FPN are as follows:

$$\begin{cases} P_3^{out} = C3\left(Concat\left(\frac{w_{11}}{\sum w_{i1}+\varepsilon}\cdot P_3^{in}, \frac{w_{21}}{\sum w_{i1}+\varepsilon}\cdot Resize\left(P_4^{in}\right), \frac{w_{31}}{\sum w_{i1}+\varepsilon}\cdot Resize\left(P_5^{in}\right)\right)\right) \\ P_4^{out} = C3\left(Concat\left(\frac{w_{12}}{\sum w_{i2}+\varepsilon}\cdot Resize\left(P_3^{in}\right), \frac{w_{22}}{\sum w_{i2}+\varepsilon}\cdot P_4^{in}, \frac{w_{32}}{\sum w_{i2}+\varepsilon}\cdot Resize\left(P_5^{in}\right)\right)\right) \\ P_5^{out} = C3\left(Concat\left(\frac{w_{13}}{\sum w_{i3}+\varepsilon}\cdot Resize\left(P_3^{in}\right), \frac{w_{23}}{\sum w_{i3}+\varepsilon}\cdot Resize\left(P_4^{in}\right), \frac{w_{33}}{\sum w_{i3}+\varepsilon}\cdot P_5^{in}\right)\right) \end{cases} \quad (1)$$

where w is the trainable weight, the *Resize* operation is a Downsampling or Upsampling operation, and ε is a minimal value.

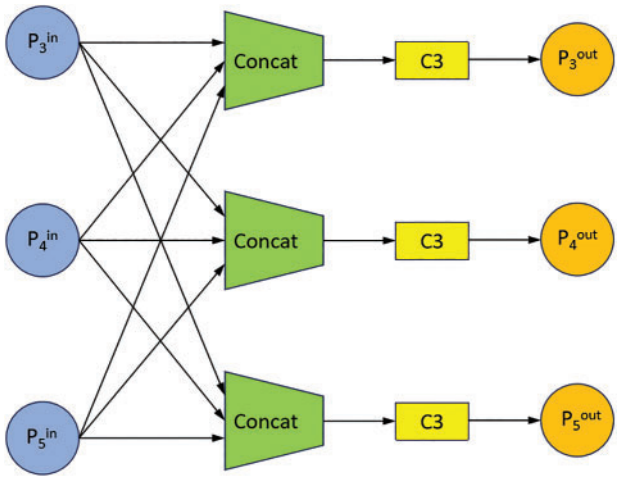

Figure 5: FC-FPN structure

FC-FPN summarizes the advantages of BiFPN and ASFF networks. Each layer integrates the characteristics of three different scale levels in the backbone network, and uses a fully connected information transmission mode to enable pest image information to be transmitted across layers. Large-scale target information can be strengthened in the shallow layer, and small-scale target information can be retained in the deep layer. Deepen the semantic understanding of pest targets. And the weighted Concat approach is used, with only nine parameters, so that the network can learn the importance of different resolution feature layers. This fully-connected weighted fusion of feature layer fusion not only improves the recognition accuracy of pests at all scales, but also multiple feature reuse of useful features in the input data and multiple utilization of feature information of pests in the image to discriminate between the pest target and the background and to reduce the bounding box loss. In addition, the single-layer network structure not only reduces the complexity of the network, but also reduces the number of convolution and C3 modules, reduces the number of parameters, and makes the network more lightweight. Finally, the C3 module used after Concat can integrate the existing data features to extract the effective information and reduce the dimensionality and complexity of the data in order to the detection header predicts the target information.

2.5 DSA Module

The Transformer self-attention mechanism is a method for calculating the correlation between the information of each element in the input sequence, which associates each input information with all the information to obtain new self-attention-weighted information. This improves the ability of the model to integrate global information and thus enhances the model's ability to understand and express the input data. For example, the Vision Transformer and Swin Transformer accomplish spatial dimensional self-attention operations by segmenting or downsampling an image into a series of patch elements and by performing interactive computations between patches. The use of spatial self-attention enhances the overall attentional weight of the pest target and reduces the attention to the background region, allowing the model to focus on the spatial region of the pest target. However, many excellent attention mechanisms not only focus on the relationship between information in the spatial dimension but also on the different degrees of importance of each channel feature, such as CBAM [17] and Triple Attention [18]. Using the Transformer self-attention mechanism in the channel dimension, the dependencies between features in the channel dimension can be captured, thus integrating all the feature information of the pests and enhancing the model's classification ability for similar pests.

The DSA module uses a Transformer self-attention mechanism to successively perform channel self-attention and spatial self-attention operations. The overall structure of the DSA module is illustrated in Fig. 6. The overall structure of the C3 module is shown in Fig. 7, which is an easily extensible module whose design flexibility allows multiple modules to replace the BottleNeck Block. In this study, the DSA module was replaced with the BottleNeck Block in the C3 module, thus embedding the DSA module into the C3 module to form DSA-C3. The overall structure of DSA-C3 module is shown in Fig. 8.

In the channel self-attention module, the information of each channel of the input feature matrix is converted into one-dimensional vectors and then used as inputs to the Transformer encoder, which is implemented using the Multihead Self-Attention (MSA) Module and the Multi-Layer Perceptron (MLP). Layer Norm is applied before each module to speed up the training. Residual Connection is applied after each module to avoid information loss. Vision Transformer and Swin Transformer enhance the spatial information of the object, while using Transformer in the channel dimension, it is possible to focus on the correlation between different features on the channel. Fig. 9 shows the structure of the channel self-attention module.

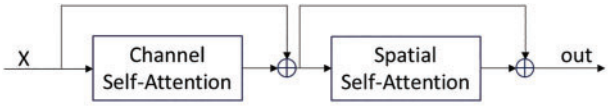

Figure 6: Overall structure of the DSA module

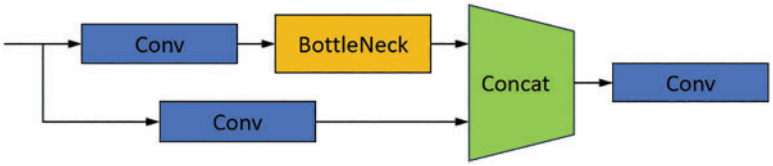

Figure 7: Overall structure of the C3 module

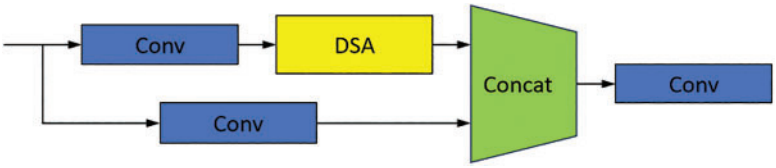

Figure 8: Overall structure of the DSA-C3 module

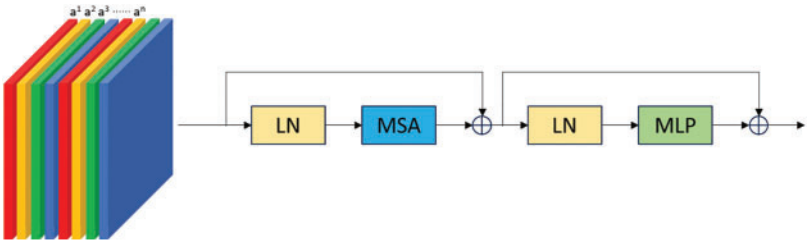

Figure 9: Structure of channel self-attention module3 equations and mathematical expressions

The self-attentive mechanism passes the information $a^1, a^2, \ldots, a^n$ from each channel through three transformation matrices W^q, W^k, W^v to obtain q^i, k^i, v^i, defined as follows:

$$\begin{cases} q^i = W^q \cdot a^i \\ k^i = W^k \cdot a^i \\ v^i = W^v \cdot a^i \end{cases} \tag{2}$$

The similarity score b_{ij} between the channel features is obtained by the inner product operation of Formula (3), which is normalized using the softmax function in order to stabilize the gradient during training, where d is the vector dimension, and is therefore divided by the square root of d in order to prevent the inner product from being too large.

$$b_{ij} = Softmax\left(\frac{q^i \cdot k^j}{\sqrt{d}}\right) \tag{3}$$

The similarity score b_{ij} is used as the weight of the channel information v_j, which is accumulated to obtain the attention result c_i of the channel for all the channel information, which is expressed by the

following formula:

$$c_i = \sum b_{ij} \times v_j \tag{4}$$

For input feature matrix $A = (a^1, a^2, \ldots a^n)$, set matrix $Q = (q^1, q^2, \ldots q^n)$, $K = (k^1, k^2, \ldots k^n)$, $V = (v^1, v^2, \ldots v^n)$ then we get overall Formula (5).

$$Attention\,(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{5}$$

The Multihead Self-Attention (MSA) module uses a multi-group transformation matrix W^q, W^k, W^v to obtain a multi-group of Query, Keys, and Values, which are computed in parallel for each head. The resultant matrix is computed using the parameter vector W0 in a weighted Concat computation, and the output of the MSA is then obtained. The spatial self-attention module uses a Swin Transformer, which computes self-attention in non-overlapping local windows instead of global self-attention, by improving the MSA module into a window-based self-attention W-MSA module and a shifted-window-based self-attention SW-MSA module, shrinking some of the receptive field but drastically reducing the computation of high-resolution images. Subsequently, the moving-window computation method was introduced to enable information linkage between windows while maintaining the computational efficiency.

3 Experiments and Analysis of Results

3.1 Model Performance

FD-YOLO model gives an initial indication of the model's performance by comparing it with the YOLOv5 baseline model on multiple metrics shown in Fig. 10. The figure shows three loss curves for the training and validation sets. Bounding box loss evaluates the model's ability to localize the target and predict the target size, and its improvement reflects the model's enhanced ability to regress the bounding box for targets of different scales. Target loss and recall reflect the loss of targets during detection, which is not ideal for YOLOv5 because of the presence of many instances of small targets, such as midges and aphids in the sample, and is improved by the FD-YOLO model. Category loss and precision reflect whether the model has the ability to discriminate between similar categories of targets, and for our complex multicategory dataset, the FD-YOLO model performs well. The mAP is the area under the P-R curve, which balances the effects of precision and recall on the model. The increase in the mAP reflects the comprehensive performance of the FD-YOLO model.

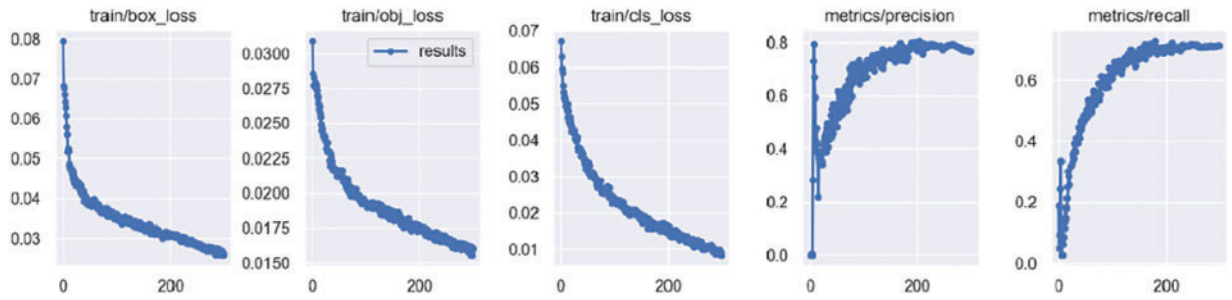

Figure 10: (Continued)

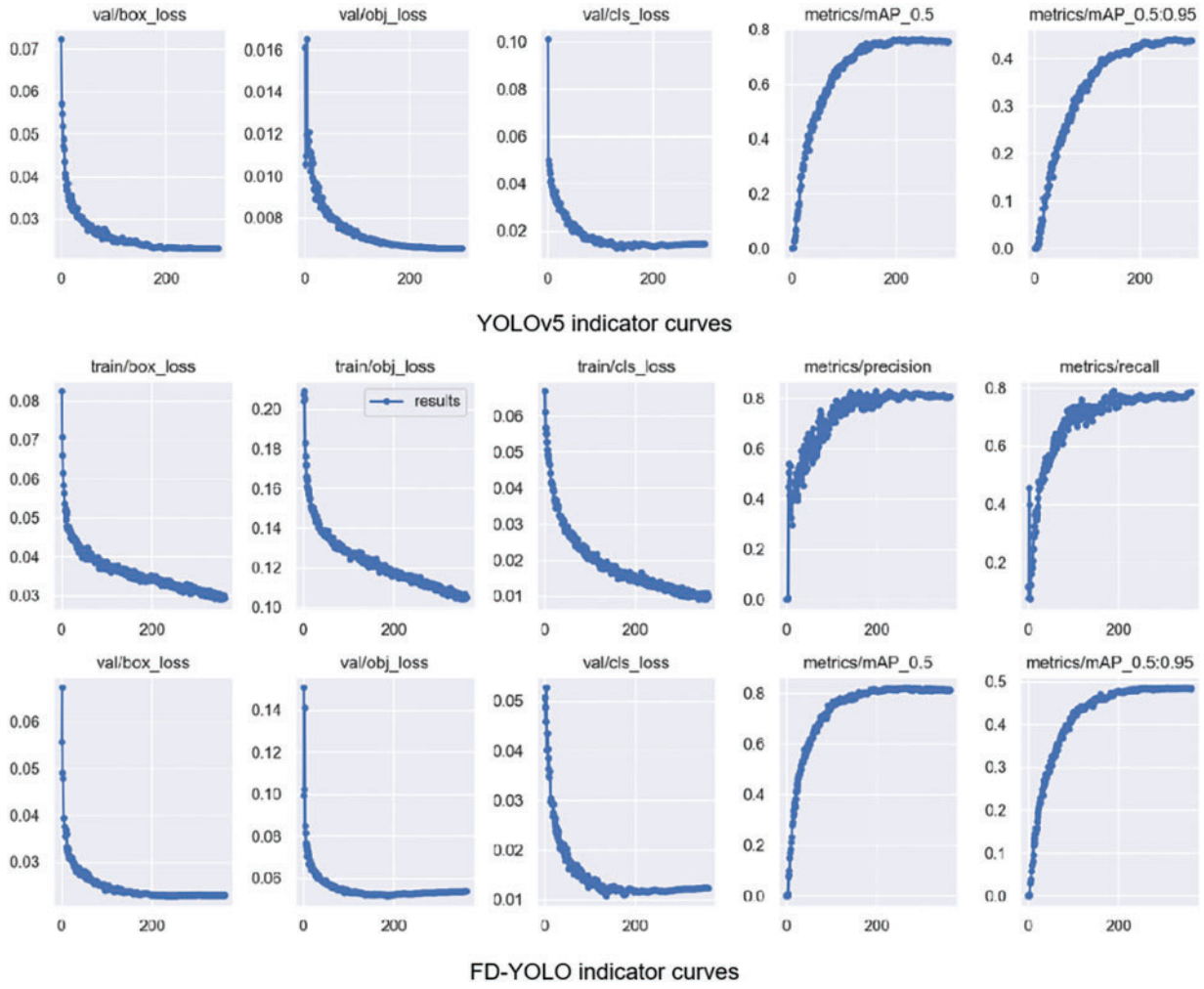

Figure 10: Comparison of YOLOv5 and FD-YOLO indicators

3.2 Comparative Tests

To verify whether the FD-YOLO model has advanced performance, we conducted controlled experiments. A total of seven advanced algorithms for target detection, YOLOv5, SSD, Faster R-CNN, Pest-YOLO, YOLOX, YOLOv8, and YOLOv10, were selected for comparison with FD-YOLO. All algorithms were applied to the augmented dataset, which was divided in a ratio of 8:2. The results of the comparison tests of the different models are listed in Table 1. The FD-YOLO model shows an improvement in all the metrics with 82.6% for mAP@0.5, 48.7% for mAP@0.5–0.95, 85% accuracy, and 76.8% recall. It improves by 6.8%, 4.8%, 5.6%, and 5.9% over the benchmark model YOLOv5, and 19.1%–2.3%, 11.3%–2.2%, 17.6%–2.7%, and 17.8%–2.9%, respectively, compared to the other models. These data demonstrate that the FD-YOLO model provides better detection results for multiscale field pests in complex situations.

Table 2 presents the performance of the double self-attention-based fully connected feature fusion network (FD) across different backbone networks. The results indicate that replacing the common backbone network does not significantly enhance the performance of the FD-YOLO model. Specifically, the precision, recall, and mAP values obtained by combining these feature extraction networks with the FD feature fusion network are slightly lower than those achieved using the original backbone network. The lightweight

architectures, such as EfficientNet v2, ShuffleNet v2, and ConvNeXt, are less effective, while the advanced Swin Transformer feature extraction architecture proves to be more effective, achieving an mAP@0.5 of 80.8%, although this is still slightly lower than that of FD-YOLO. Therefore, we incorporate the FD feature fusion network into the YOLO network with the C3 feature extraction architecture, resulting in our proposed FD-YOLO model based on YOLOv5.

Table 1: Comparison of the results of the models

	mAP@0.5	mAP@0.5–0.95	Precision	Recall	F1
YOLOv5	75.8	43.9	79.4	70.9	74.9
SSD	63.5	37.4	67.4	59.0	62.9
Faster RCNN	71.5	42.1	73.7	65.3	69.2
Pest-YOLO	76.4	44.8	80.2	71.7	75.7
YOLOX	73.0	41.7	77.2	68.6	72.6
YOLOv7	75.2	44.1	77.9	69.3	73.3
YOLOv8	80.3	46.5	82.3	73.9	77.9
YOLOv10	79.1	45.9	80.5	73.4	76.8
FD-YOLO	82.6	48.7	85.0	76.8	80.7

Table 2: FD performance in different backbone networks

	mAP@0.5	mAP@0.5–0.95	Precision	Recall	F1
EfficientNet v2+FD	78.3	45.3	81.5	71.0	75.9
ShuffleNet v2+FD	75.7	43.7	79.2	69.5	74.0
ResNet-50+FD	79.2	44.2	81.4	71.3	76.0
ConvNeXt+FD	73.9	41.9	76.4	68.6	72.3
Swin Transformer+FD	80.8	45.3	81.6	73.1	77.1
FD-YOLO	82.6	48.7	85.0	76.8	80.7

3.3 Generalizability and Stability

In order to thoroughly explore the generalization potential of the model in the pest detection task, we used the full IP102 dataset to evaluate its generalization ability, and the experimental results are shown in Table 3. Even when facing the challenge of a larger and more diverse dataset, FD-YOLO is still able to demonstrate excellent pest recognition capabilities. Compared with similar models, FD-YOLO achieves the best performance in the key metric of mAP@0.5. Therefore, FD-YOLO has a wide range of application potential and can be flexibly adapted to detect various types of pests on different crops, showing strong practical value and adaptability.

To verify the stability of the FD-YOLO model, we performed cross-validation. We divided the dataset into five subsets, rotating each subset as a test set and using the rest as a training set, and tested the model five times independently. This validation can comprehensively and objectively assess the generalization ability and stability of the FD-YOLO model on the field pest dataset. The cross-validation results are shown in Table 4. By comparing the results of the five experiments, we can find that the performance of the model is very close to one another when different subsets are used as test sets, which indicates that the FD-YOLO model has excellent stability and consistency.

Table 3: Comparison of results across models on the IP102 dataset

	mAP@0.5	mAP@0.5–0.95	Precision	Recall	F1
YOLOv5	56.2	34.1	51.8	55.1	53.4
SSD	47.2	21.5	43.9	47.0	45.4
Faster RCNN	48.0	28.4	41.6	45.4	43.4
YOLOX	52.1	31.1	48.2	52.3	50.2
YOLOv8	58.8	39.4	51.7	56.7	54.1
FD-YOLO	60.5	40.7	55.0	58.2	56.6

Table 4: Cross-validation results

	mAP@0.5	mAP@0.5–0.95	Precision	Recall	F1
Experiment 1	82.6	48.7	85.0	76.8	80.7
Experiment 2	83.4	48.9	86.6	77.9	82.0
Experiment 3	83.1	49.2	84.5	76.6	80.4
Experiment 4	81.9	48.0	85.3	77.4	81.2
Experiment 5	82.3	48.5	84.8	75.1	79.7

3.4 Confusion Matrix

The confusion matrix generated by applying the FD-YOLO model to our dataset is shown in Fig. 11. The horizontal axis represents the true label of the target, while the vertical axis represents the predicted category of the model for the target, and the value on the diagonal represents the recall of the category. Compared to the YOLOv5 model, the FD-YOLO model has improved recall for all categories. Categories such as mole cricket, thrips, and rice water weevil are not easily confused with other categories, so they achieve recalls of 0.98%, 0.94%, and 0.93%, respectively. However, there are still some objective reasons that lead to the unsatisfactory classification of some categories. For example, the army worm is more similar to the cutworm, and 20% of army worms were predicted to be cutworms. Due to the varying degrees of similarity between the larvae or adults of the rice leaf roller, rice leaf caterpillar, Asiatic rice borer, and corn borer, there is a high rate of misclassification among them, resulting in low recall. In addition, aphids and planthoppers suffer from small target scales and poorly characterized features, respectively, resulting in these two class instars remaining relatively easy to confuse with background badlands.

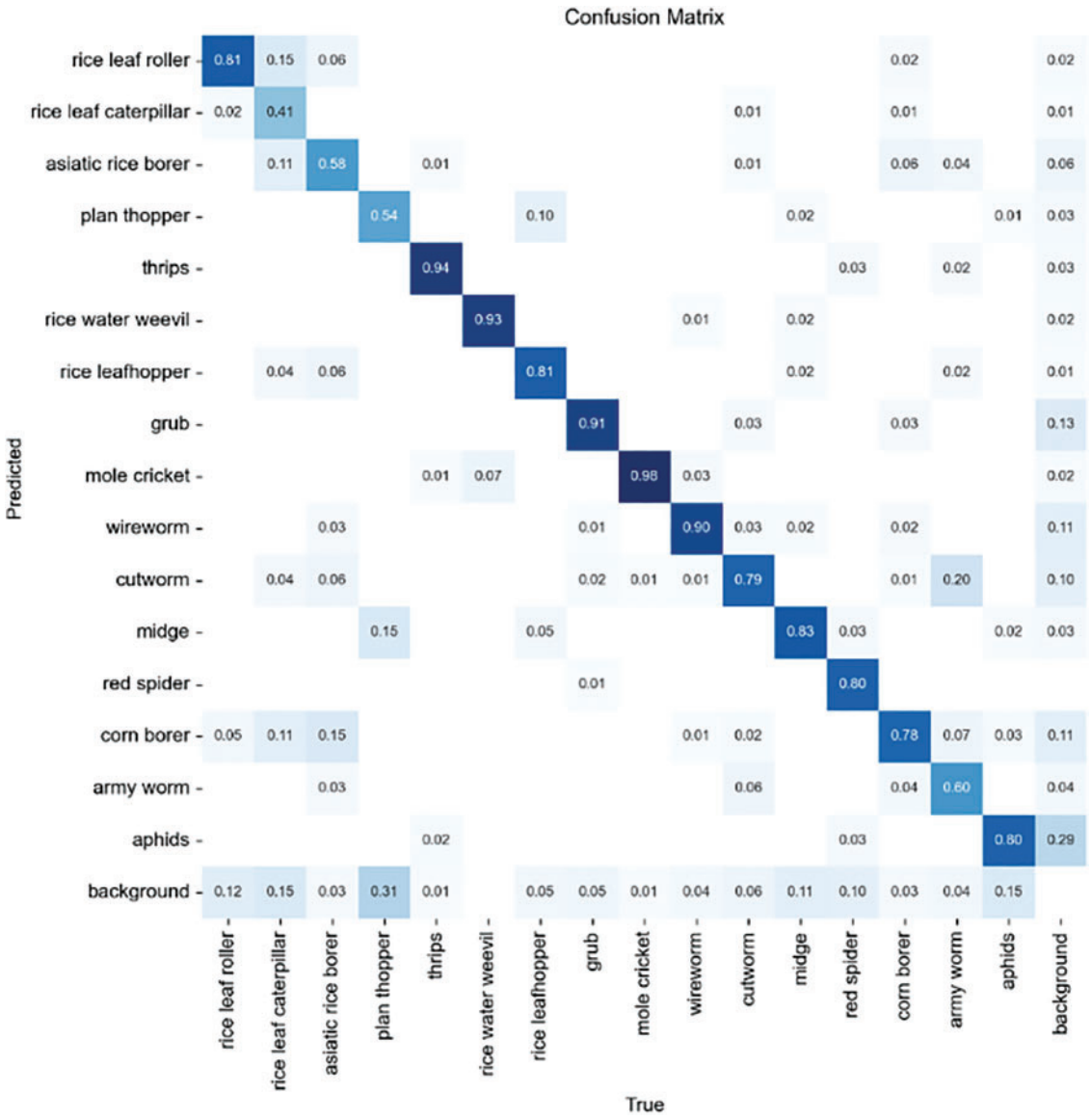

Figure 11: Confusion matrix

3.5 Qualitative Analysis

Six example images with high error rates were selected from the detection results of YOLOv5 and FD-YOLO to qualitatively analyze the specific cases of model failure, as shown in Fig. 12. In Fig. 12a, both YOLOv5 and FD-YOLO have different degrees of missed detection in the images due to the multi-scale variation of the pest target and its similarity to the background. In Fig. 12b, the background environment is complex, and YOLOv5 misdetects branches in the background as pest targets, while FD-YOLO corrects this situation. The rice leaf caterpillar is easily misclassified as other categories due to the small number of samples and the similarity of its characteristics with those of other multi-category pests. In Fig. 12c, FD-YOLO is correctly categorized, but it has low confidence. In Fig. 12d, both YOLOv5 and FD-YOLO are misclassified in the images, but the confidence level of FD-YOLO for the wrong category is reduced. In Fig. 12e, both YOLOv5 and FD-YOLO are recognized correctly in the images, and FD-YOLO not only has improved confidence, but

also has a better bounding box and reduced bounding box loss. In Fig. 12f, YOLOv5 misidentifies asiatic rice borer as rice leaf caterpillar, and FD-YOLO has incorrect prediction frames that identify asiatic rice borer as corn borer in addition to those that correctly identify the target. Although it improved the recall of asiatic rice borer, this diminished the accuracy of corn borer.

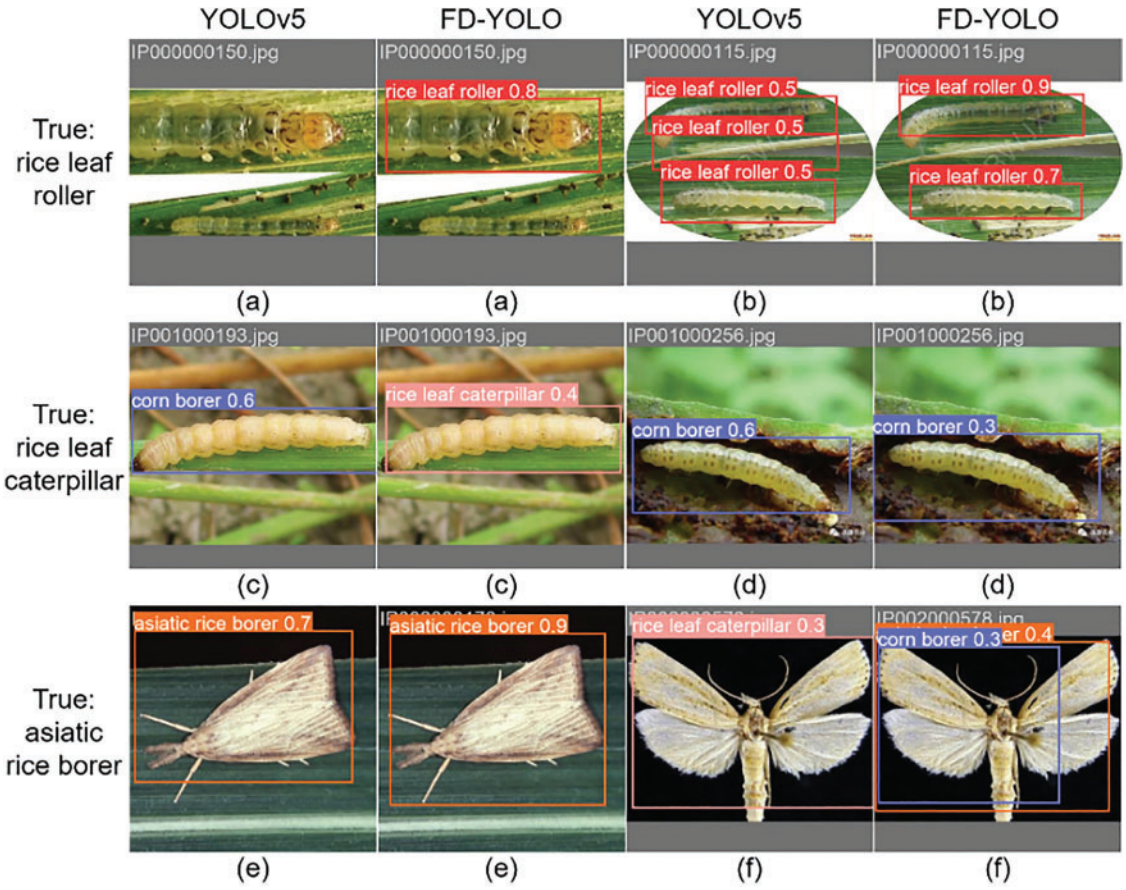

Figure 12: Example of comparison of test results

3.6 Ablation Study

In order to verify the effectiveness and advancements of each component in the FD-YOLO model, we conducted the following ablation experiments. The FC-FPN and DSA modules were first removed one by one to observe the effectiveness of these two modules, and the experimental results are shown in Table 5. When only FC-FPN is used, mAP@0.5 reaches 81.3%, which is 5.5% higher than YOLOv5. This proves that FC-FPN can better fuse multi-scale features and strengthen the model's ability to detect targets across all scales. When only the DSA module is used, the precision and recall are 81.2% and 72.9%, respectively, which are 1.8% and 2% higher. This proves that the DSA module enhances feature extraction and improves the ability to discriminate between similar pests and to distinguish pest targets from the background. The mAP@0.5 of FD-YOLO is 82.6%, which is 6.8% higher than YOLOv5. This demonstrates that using the FC-FPN and DSA modules together can better detect multi-scale and similar field pest targets in complex backgrounds.

In order to verify the advancements and lightweight nature of FC-FPN, we compare it with PANet, ASFF, and BiFPN feature fusion necks, and the experimental results are shown in Table 6, which reflects

that FC-FPN successfully combines the advantages of both ASFF and BiFPN, reducing the model size and accelerating the speed of detection, making it a new and effective feature fusion method.

Table 5: Results of ablation experiments

	mAP@0.5	mAP@0.5–0.95	Precision	Recall	F1
YOLOv5	75.8	43.9	79.4	70.9	74.9
YOLO5+FC-FPN	81.3	47.5	84.4	75.8	79.9
YOLOv5+DSA	78.2	45.7	81.2	72.9	76.8
FD-YOLO	82.6	48.7	85	76.8	80.7

Table 6: Comparison of results of each neck

	mAP@0.5	mAP@0.5–0.95	Precision	Recall	F1	FPS	GFLOPs
YOLOv5 (PANet)	75.8	43.9	79.4	70.9	74.9	99.0	15.9
YOLOv5+ASFF	76	44.8	78.2	70.1	73.9	89.6	24.3
YOLO5+BiFPN	77.6	45.3	79.7	72.2	75.8	98.4	17.2
YOLO5+FC-FPN	81.3	47.5	84.4	75.8	79.9	103.5	15.1

We also compare the DSA module with a variety of attention modules, including the DSA module, to quantitatively analyze the performance, and the experimental results are shown in Table 7. The DSA module outperforms both the Triplet attention module and the CBAM module, reflecting its advanced global information capture and integration capabilities, which stem from the Transformer self-attention mechanism. The enhancement of the DSA module over the STR module reflects that the channel self-attention mechanism achieves improved feature extraction and integration.

Table 7: Comparison of attention modules

	mAP@0.5	mAP@0.5–0.95	Precision	Recall	F1
YOLOv5+Triplet	74.6	43.7	79.0	70.5	74.5
YOLOv5+CBAM	77.8	44.6	78.4	71.2	74.6
YOLOv5+STR	77.1	45.1	80	71.9	75.7
YOLOv5+DSA	78.2	45.7	81.2	72.9	76.8

To visualize the effects of the various components added to the FD-YOLO model, we plotted heat maps for schemes I, III, V, and VI using Gradient-weighted Class Activation Mapping (Grad-CAM) [19], as shown in Fig. 13. The heat map demonstrates the distribution of the model's attention during detection through the color shades, reflecting the important data to which the model pays attention during prediction. From the figure, it can be seen that the YOLOv5 model is deficient in feature extraction and information integration for pest data, and fails to effectively capture key information from pest images. With the use of FC-FPN, the model enhances feature reuse and fuses multi-scale information, which does not ignore pest information but instead focuses on varying degrees of background regions unrelated to the target. With the use of the DSA module, the model enhances its ability to discriminate between background and target, further focusing

on pest-related features. The FD-YOLO model, on the other hand, more clearly highlights the pest target region in the input image, which reflects the model's success in focusing on task-critical information during the decision-making process and demonstrates that the model learns to effectively utilize and integrate key features of the input data during training.

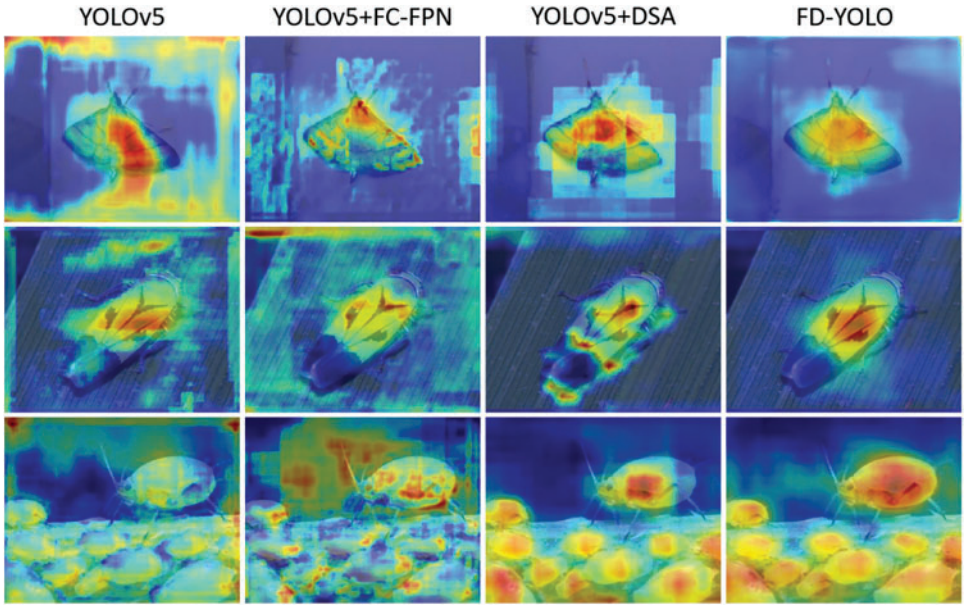

Figure 13: Heat map for each scheme

4 Discussion

This study focuses on the problem of field pest detection, where early and accurate monitoring of pest populations is crucial for pest control and prevention, effectively reducing food losses. Our proposed FD-YOLO model significantly improves the identification and localization of target pests by addressing three common problems: interspecific similarity, multiscale, and background complexity in pest detection tasks.

Compared with the baseline model YOLOv5, the FD-YOLO model replaces the neck with FC-FPN and subsequently adds the DSA module. FC-FPN improves the ability to recognize pests at multiple scales by using a fully connected approach for multi-scale feature fusion. Heatmap Fig. 13 shows that with the use of FC-FPN, the model enhances feature reuse and fuses multiscale information, without essentially ignoring pest information. The enhancement of mAP in Table 6 indicates that FC-FPN has improved the model's ability to locate targets and the detection accuracy of pests at various scales. The DSA module not only uses the advanced Swin Transformer for spatial self-attention, enhancing the overall attention weight of the pest target while reducing attention to the background region, but also employs channel self-attention to capture the dependencies between features in the channel dimension, to deeply understand the global features of pests and enhance the model's classification performance. Heatmap Fig. 13 shows that with the use of the DSA module, the model further focuses on the relevant features of the pest target while ignoring background information, reflecting the model's enhanced ability to detect pests that are easily confused with the background. The improvement in model accuracy shown in Table 7 demonstrates that the DSA module strengthens feature extraction and improves the ability to discriminate between similar pests. The combined use of the FC-FPN and DSA modules enables the model to better understand field pests, and the metrics of

FD-YOLO outperform those of other state-of-the-art networks, indicating that the algorithm has significant potential and value in the field of pest detection.

However, the FD-YOLO model also has some limitations. For example, for small-scale pest categories, such as aphids, the model still confuses targets with the background due to limited target information. In the next step, we plan to input the output of the first C3 module into FC-FPN and add a (160, 160)-sized small-target detection head to improve the model's detection ability for small-scale pests. However, improving small-target recognition ability while maintaining recognition ability for normal targets is a challenging aspect of this work. In addition, improving the DSA module to integrate both spatial and channel dimension information to further enhance feature extraction capability and improve detection performance for pests similar to other pests or the background is also a future priority. The FD-YOLO algorithm also has many potential applications. For example, through domain adaptation technology, fine-tuning and transfer learning on different pest datasets can be conducted to identify more types of pests, strengthening the model's recognition ability in harsh scenarios, reducing the impact of dataset bias, and adapting to tasks in more fields.

5 Conclusion

In order to solve the interspecies similarity, multi-scale and background complexity problems of pest target detection tasks and promote the practical application of computer vision technology in agriculture, this paper proposes the FD-YOLO model. By improving the neck feature fusion network, designing FC-FPN, and proposing the DSA module based on the self-attention mechanism, we enhance the detection effect of the model for different scales of targets and improve the problems of similar pests being difficult to categorize and easy to confuse with the background. We selected 16 pests in the IP102 pest dataset, which are widely harmful to field crops, for data supplementation and enhancement in experiments. The experimental results show that the FD-YOLO model's mAP@0.5 is 82.6%, mAP@0.5–0.95 is 48.7%, precision is 85%, and recall is 76.8%. It shows improvements of 6.8%, 4.8%, 5.6%, and 5.9% respectively over the benchmark model YOLOv5, and the mAP@0.5 is 19.1% to 2.3% higher than other state-of-the-art models.

Acknowledgement: Thanks to all the authors cited in this article and the referee for their helpful comments and suggestions.

Funding Statement: This research was funded by Liaoning Provincial Department of Education Project, Award number JYTMS20230418.

Author Contributions: Zijun Gao proposed research and secured funding. Zheyi Li conducted analyses and wrote the paper. Chunqi Zhang and Ying Wang revised and commented on the article. Jingwen Su reviewed the article. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the corresponding author, Z.G., upon reasonable request.

Ethics Approval: Not applicable.

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

References

1. Liu J, Wang X, Miao W, Liu G. Tomato pest recognition algorithm based on improved YOLOv4. Front Plant Sci. 2022;13:814681. doi:10.3389/fpls.2022.814681.

2. Feng F, Dong H, Zhang Y, Zhang Y, Li B. MS-ALN: multiscale attention learning network for pest recognition. IEEE Access. 2022;10(2):40888–98. doi:10.1109/ACCESS.2022.3167397.
3. Liu J, Wang X. Tomato diseases and pests detection based on improved Yolo V3 convolutional neural network. Front Plant Sci. 2020;11:898. doi:10.3389/fpls.2020.00898.
4. Wang X, Du J, Xie C, Wu S, Ma X, Liu K, et al. Prior knowledge auxiliary for few-shot pest detection in the wild. Front Plant Sci. 2023;13:1033544. doi:10.3389/fpls.2022.1033544.
5. Zhu H, Wei H, Li B, Yuan X, Kehtarnavaz N. A review of video object detection: datasets, metrics and methods. Appl Sci. 2020;10(21):7834. doi:10.3390/app10217834.
6. Rong M, Wang Z, Ban B, Guo X. Pest identification and counting of yellow plate in field based on improved mask R-CNN. Discrete Dyn Nat Soc. 2022;2022(1):1913577. doi:10.1155/2022/1913577.
7. Wang F, Wang R, Xie C, Zhang J, Li R, Liu L. Convolutional neural network based automatic pest monitoring system using hand-held mobile image analysis towards non-site-specific wild environment. Comput Electron Agric. 2021;187:106268. doi:10.1016/j.compag.2021.106268.
8. Ali F, Qayyum H, Iqbal MJ. Faster-PestNet: a lightweight deep learning framework for crop pest detection and classification. IEEE Access. 2023;11:104016–27. doi:10.1109/ACCESS.2023.3317506.
9. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016 Jun 27–30; Las Vegas, NV, USA: IEEE; 2016. p. 779–88. doi:10.1109/CVPR.2016.91.
10. Zhang Y, Yang G, Liu Y, Wang C, Yin Y. An improved YOLO network for unopened cotton boll detection in the field. J Intell Fuzzy Syst. 2022;42(3):2193–206. doi:10.3233/JIFS-211514.
11. Wen C, Chen H, Ma Z, Zhang T, Yang C, Su H, et al. Pest-YOLO: a model for large-scale multi-class dense and tiny pest detection and counting. Front Plant Sci. 2022;13:973985. doi:10.3389/fpls.2022.973985.
12. Hu Y, Deng X, Lan Y, Chen X, Long Y, Liu C. Detection of rice pests based on self-attention mechanism and multi-scale feature fusion. Insects. 2023;14(3):280. doi:10.3390/insects14030280.
13. Yin J, Huang P, Xiao D, Zhang B. A lightweight rice pest detection algorithm using improved attention mechanism and YOLOv8. Agriculture. 2024;14(7):1052. doi:10.3390/agriculture14071052.
14. Dong Q, Sun L, Han T, Cai M, Gao C. PestLite: a novel YOLO-based deep learning technique for crop pest detection. Agriculture. 2024;14(2):228. doi:10.3390/agriculture14020228.
15. Wu X, Zhan C, Lai YK, Cheng MM, Yang J. IP102: a large-scale benchmark dataset for insect pest recognition. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2019 Jun 15–20; Long Beach, CA, USA: IEEE; 2019. p. 8779–88. doi:10.1109/cvpr.2019.00899.
16. Liu S, Qi L, Qin H, Shi J, Jia J. Path aggregation network for instance segmentation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA: IEEE; 2018. p. 8759–68. doi:10.1109/CVPR.2018.00913.
17. 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. Lecture notes in computer science. Vol. 11211. Cham: Springer; 2018. p. 3–19. doi:10.1007/978-3-030-01234-2_1.
18. Misra D, Nalamada T, Arasanipalai AU, Hou Q. Rotate to attend: convolutional triplet attention module. In: 2021 IEEE Winter Conference on Applications of Computer Vision (WACV); 2021 Jan 3–8; Waikoloa, HI, USA: IEEE; 2021. p. 3138–47. doi:10.1109/WACV48630.2021.00318.
19. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-CAM: visual explanations from deep networks via gradient-based localization. In: 2017 IEEE International Conference on Computer Vision (ICCV); 2017 Oct 22–29; Venice, Italy: IEEE; 2017. p. 618–26. doi:10.1109/ICCV.2017.74.