

Article

A Lightweight Crop Pest Classification Method Based on Improved MobileNet-V2 Model

Hongxing Peng ^{1,2,3} , Huiming Xu ⁴, Guanjia Shen ¹, Huanai Liu ^{5,*}, Xianlu Guan ⁶ and Minhui Li ^{2,7,*}

- ¹ College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China; xyphx@scau.edu.cn (H.P.); sgj18300000677@gmail.com (G.S.)
- ² Guangdong Province Key Laboratory of Microbial Signals and Disease Control, South China Agricultural University, Guangzhou 510642, China
- ³ Key Laboratory of Smart Agricultural Technology in Tropical South China, Ministry of Agriculture and Rural Affairs, Guangzhou 510642, China
- ⁴ Zhuhai Wanlida Electric Automation Company with Limited Liability, Zhuhai 519000, China; x15979695993@163.com
- ⁵ School of Chemistry and Chemical Engineering, South China University of Technology, Guangzhou 510641, China
- ⁶ Guangdong Laboratory for Lingnan Modern Agriculture, College of Engineering, South China Agricultural University, Guangzhou 510642, China; guanxl@stu.scau.edu.cn
- ⁷ College of Plant Protection, South China Agricultural University, Guangzhou 510642, China
- * Correspondence: liuhn@scut.edu.cn (H.L.); liminhui@scau.edu.cn (M.L.)

Abstract: This paper proposes PestNet, a lightweight method for classifying crop pests, which improves upon MobileNet-V2 to address the high model complexity and low classification accuracy commonly found in pest classification research. Firstly, the training phase employs the AdamW optimizer and mixup data augmentation techniques to enhance the model's convergence and generalization capabilities. Secondly, the Adaptive Spatial Group-Wise Enhanced (ASGE) attention mechanism is introduced and integrated into the inverted residual blocks of the MobileNet-V2 model, boosting the model's ability to extract both local and global pest information. Additionally, a dual-branch feature fusion module is developed using convolutional kernels of varying sizes to enhance classification performance for pests of different scales under real-world conditions. Lastly, the model's activation function and overall architecture are optimized to reduce complexity. Experimental results on a proprietary pest dataset show that PestNet achieves classification accuracy and an F1 score of 87.62% and 86.90%, respectively, marking improvements of 4.20 percentage points and 5.86 percentage points over the baseline model. Moreover, PestNet's parameter count and floating-point operations are reduced by 14.10% and 37.50%, respectively, compared to the baseline model. When compared with ResNet-50, MobileNet V3-Large, and EfficientNet-B1, PestNet offers superior parameter efficiency and floating-point operation requirements, as well as improved pest classification accuracy.

Keywords: MobileNet-V2; deep learning; pest identification



Citation: Peng, H.; Xu, H.; Shen, G.; Liu, H.; Guan, X.; Li, M. A Lightweight Crop Pest Classification Method Based on Improved MobileNet-V2 Model. *Agronomy* **2024**, *14*, 1334. <https://doi.org/10.3390/agronomy14061334>

Academic Editor: Gniewko Niedbala

Received: 7 May 2024

Revised: 12 June 2024

Accepted: 17 June 2024

Published: 20 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

For a long time, agricultural pests have posed a serious threat to the growth of crops and the storage of agricultural products [1]. According to the Food and Agriculture Organization's (FAO) report, these pests cause global crop yield losses ranging from 20% to 40% annually. To control pests, farmers commonly use chemicals such as pesticides, which, however, have negative impacts on agricultural ecosystems [2]. Achieving integrated pest management involves optimizing combinations of mechanical, chemical, biological, and genetic tools to mitigate harmful impacts and enhance beneficial effects, rather than relying excessively on pesticides [3].

Timely and accurate detection and classification of diseases and pests are crucial for effective pest control. Early detection is a prerequisite for devising effective disease and pest management plans and can reduce contamination. Traditional crop pest and disease classification mainly relies on manual observation or expert guidance, which is slow, inefficient, costly, and subjective. With the development of machine learning methods and computer vision technology, researchers have begun to use information technology to identify images of agricultural pests.

Fina et al. [4] employed the k-means clustering algorithm to separate pests from the background and used corresponding filters to identify species. However, the filters were manually designed, and the number of images for training and testing was limited. Xie et al. [5] used sparse coding histograms to represent the appearance features of pests and fused various features through multi-kernel learning for classification. While their image dataset was larger, the backgrounds in the images were simpler. Ali et al. [6] used the ΔE color difference algorithm to separate areas affected by citrus diseases and classified diseases with 99.9% accuracy by combining color and texture features. However, their process was more complex. Ebrahimi et al. [7] designed an SVM classifier with a differential kernel function, employing region and color indicators for thrips identification, achieving a best classification average error rate of less than 2.25%. Bisgin et al. [8] constructed an SVM classifier using features such as size, color, basic patterns, and textures and experimented on a dataset containing 6900 images, demonstrating that the SVM performed well in classifying 15 species of beetle-contaminated foods. Pattnaik et al. [9] combined image processing techniques with machine learning, utilizing Local Binary Patterns (LBP) for feature extraction and constructing an SVM classifier for tomato pest classification, achieving an accuracy of 81.02%. Zhang et al. [10] employed BOF to extract pest features and improved the SVM model, obtaining good experimental results on four types of vegetable pests. The latest improvements in computer hardware and advancements in parallel computing have made it easier to build and train deep neural networks. Unlike traditional machine learning, deep convolutional neural networks (CNNs) are representational learning models that do not require handcrafted features for classifiers [11].

Deep learning models also receive more image data for training than traditional machine learning models, ensuring that they acquire more comprehensive knowledge and are applicable to real-world scenarios. Li et al. [12] selected five CNN models to identify 10 common agricultural pests in nature, with each pest category containing over 400 images. Fine-tuned GoogLeNet-V3 achieved the best performance. Chen et al. [13] created a CNN architecture named INC-VGGN for detecting diseases in rice and corn, leveraging pre-trained VGG-19 and replacing its last few layers with two Inception modules to further enhance feature extraction capabilities. Compared to other baseline networks, this new model achieved the highest classification accuracy (92%). Tetila et al. [14] employed transfer learning strategies to fine-tune models such as Inception-V3, ResNet-50, VGG-16, VGG-19, and Xception. They conducted experiments on a dataset containing 5000 soybean pest images, achieving classification results superior to traditional feature extraction methods like SIFT. Ayan et al. [15] utilized a weighted voting ensemble approach to integrate pre-trained models, including Inception-V3, Xception, and MobileNet. Their constructed GAEnsemble model achieved a classification accuracy of 67.13% on the IP102 dataset. Khanramaki et al. [16] constructed a citrus pest image dataset containing 1774 images and proposed a deep learning-based ensemble classifier to identify three common citrus pests, achieving an accuracy of 99.04%. Wu et al. [17] improved the ResNet-50 model using different attention mechanisms. Experimental results showed that networks embedded with ECA attention modules and CCO modules achieved higher prediction accuracy. Wang et al. [18] constructed a dataset called IP41 containing 41 classes and 46,567 images of pests and trained three different deep learning models using a combination of transfer learning and fine-tuning methods, all achieving classification accuracy above 80.00%. Peng et al. [19] constructed a high-quality dataset called HQIP102 consisting of 102 pest categories from eight crops with over 40,000 images. To address the problem of data imbalance, they

proposed a dynamic data augmentation method, and the average classification accuracy of the proposed MADN convolutional neural network model was 75.28%. Guan et al. [20] designed a novel network architecture based on the EfficientNetV2 model, called Dise-Efficient. This model achieved an accuracy of 99.80% on the Plant Village dataset for plant diseases and pests. Additionally, through transfer learning on the IP102 dataset, which represents real environmental conditions, the Dise-Efficient model achieved an accuracy of 64.40% in plant disease and pest identification. Xia et al. [21] proposed a novel pest classification method based on convolutional neural networks (CNN) and an improved visual Transformer model. They designed MMAINet to extract features of identified objects from different scales and finer granularities. Subsequently, a classification model named DenseNet Vision Transformer (DNVT) was proposed. Experimental results on the D0 and IP102 datasets demonstrate that the proposed methods achieved maximum classification accuracies of 99.89% and 74.20%, respectively. Wei et al. [22] proposed a crop pest classification method based on Multi-Scale Feature Fusion (MFFNet) to accurately identify and classify crop pests. They designed a Multi-Scale Feature Extraction (MFE) module using dilated convolutions and extracted deep feature information from images using a Deep Feature Extraction module (DFE). The features extracted by the MFE and DFE modules were then fused, enabling end-to-end, accurate classification and identification of crop pests. Experiments on a dataset of 12 types of pests demonstrated the excellent performance of MFFNet, achieving a classification accuracy (ACC) of 98.2%. Hechen et al. [23] introduced a novel architecture called Dilated Window Vision Transformer with Efficient Self-Attention (DWViT-ES), based on dilated windows in visual Transformer models and efficient suppression of self-attention. Experimental results demonstrate that DWViT-ES comprises only 19.6 million parameters and 3.5 billion FLOPs (reducing by over 20% compared to Swin-T, which has 19.6 million parameters and 4.5 billion FLOPs).

The above research demonstrates the effectiveness of CNN models in pest classification. However, most studies have made few improvements to the model structure, and the used models do not have lightweight characteristics, resulting in low recognition efficiency. Lightweight CNN models with faster inference speeds and fewer parameters are gradually favored by researchers. Ashwinkumar et al. [24] proposed an automatic classification model for plant leaf diseases based on optimal mobile networks on the basis of the MobileNet series models, achieving a recognition accuracy of 0.985. Among the research based on lightweight CNN models, there are relatively few studies on pest classification. In addition, existing lightweight CNN models have lower accuracy in classifying pests with complex backgrounds, varying sizes, diverse postures, and fine granularity. Considering the practical application needs, this paper improves the MobileNet-V2 model from the perspectives of increasing classification accuracy, reducing model parameter count, and floating-point computational complexity, proposing a lightweight crop pest classification method called PestNet.

2. Related Work

2.1. Pest Dataset

Xing and Lee [25] pointed out in their paper that the IP102 pest dataset contains many invalid image data. The main types of invalid image data include incorrectly labeled images and images with missing target objects. As an example, Figure 1 illustrates some of the invalid image data contained in the *Orseolia oryzae* category of the IP102 dataset.

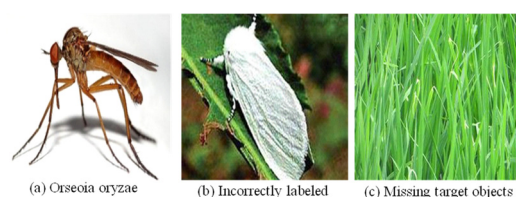


Figure 1. Some examples of images of rice gall midges from the IP102 dataset.

From the above image, it is evident that the category of pest in Figure 1b is not *Orseolia oryzae*, and Figure 1c does not contain any target objects of *Orseolia oryzae*. We invited agricultural pest experts and volunteers to screen the IP102 dataset, removing invalid pest images from the dataset, resulting in 13,848 images covering 32 common pest categories. To further supplement pest image data, team members captured 2043 pest images in the fields and litchi orchards in Guangzhou, Guangdong Province, using Canon EOS-60D cameras and mobile devices. This included two types of pests harmful to rice, *Pomacea canaliculata* and *Spodoptera frugiperda*, and three types of pests harmful to litchi: *Cryptotympana atrata*, the Scarab beetle, and *Tessaratoma papillosa*. Through the above two methods, a total of 15,891 images covering 37 pest categories were collected, forming the Pest37 pest classification dataset. The overall composition of this dataset is shown in Table 1, and some pest images are shown in Figure 2.

Table 1. The composition of the Pest37 dataset.

Crop Type	Pest Species Name	Number of Images
Rice	<i>Cnaphalocrocis medinalis</i>	178
	<i>Scirpophaga incertulas</i>	143
	<i>Orseolia oryzae</i>	291
	<i>Pomacea canaliculata</i>	523
	<i>Spodoptera frugiperda</i>	255
	<i>Lissorhoptus oryzophilus</i>	401
Corn	Grubs	840
	Mole Cricket	1621
	Black Cutworm	191
	<i>Protaetia brevitarsis</i> Lewis	459
	<i>Dichocrocis punctiferalis</i>	199
Mango	<i>Scirtothrips dorsalis</i> Hood	417
	<i>Lawana imitata</i> Melichar	573
Beet	<i>Loxostege sticticalis</i>	190
	Oriental Serica	140
Alfalfa	Acrididae	1359
	<i>Epicauta gorhami</i> Marseul	957
	<i>Mylabris phalerata</i> Pallas	275
	<i>Pieris canidia</i>	453
Grape	<i>Brevipalpus lewisi</i> McGregor	74
	<i>Oides decempunctata</i>	280
	<i>Paranthrene regalis</i> Butler	70
	<i>Ampelophaga</i>	405
	<i>Cicadella viridis</i>	1224
	Whitefly	575
	<i>Erythroneura apicalis</i>	65
Wheat	<i>Sitodiplosis mosellana</i>	196
Citrus	<i>Panonychus citri</i> McGregor	403
	<i>Phyllocoptruta oleivora</i>	383
	<i>Phyllocoptruta oleivora</i> Ashmead	36
	<i>Unaspis yanonensis</i>	420
	<i>Ceroplastes rubens</i>	257
	<i>Aleurocanthus spittiferus</i> Quaintance	690
	<i>Bactrocera minax</i>	83
Litchi	<i>Cryptotympana atrata</i>	256
	Scarab beetle	428
	<i>Tessaratoma papillosa</i>	581
Total		15,891

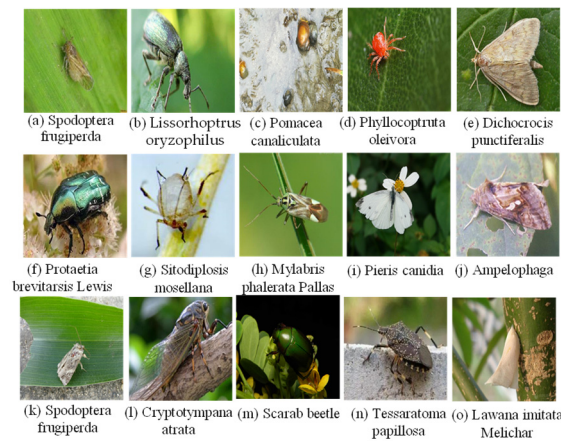


Figure 2. Some examples of images of pests from the Pest37 dataset.

2.2. MobileNet

With the rapid development of mobile technology, the demand for AI applications on mobile devices continues to grow. To meet this market demand, Google introduced a lightweight convolutional neural network designed specifically for mobile and embedded systems called MobileNetV1 [26]. MobileNetV1 effectively reduces the size and computational complexity of the model through depthwise separable convolutions, enabling AI technology to be applied to resource-constrained mobile devices.

Despite the significant achievements in lightweight and efficiency, the performance and computational efficiency of MobileNetV1 still have room for improvement. Based on this, Google further improved MobileNetV1 and introduced the more efficient MobileNetV2 [27]. This model builds upon MobileNet-V1 by adding linear bottleneck layers and inverted residual blocks, improving recognition accuracy, and reducing the model's parameter count. The linear bottleneck layers use linear activation functions instead of ReLU6 activation functions, reducing the model's influence on feature information. On the other hand, MobileNetV2 borrows the $1 \times 1 \rightarrow 3 \times 3 \rightarrow 1 \times 1$ pattern from ResNet and introduces the inverted residual structure with Shortcut connections that add the output to the input. This not only reduces the parameter count and computational complexity but also improves the model's performance. Unlike the ResNet residual modules, which first reduce dimensionality, convolve, and then increase dimensionality, the inverted residual modules in MobileNetV2 first increase dimensionality, convolve, and then reduce dimensionality, using depthwise convolutions instead of regular convolutions, simplifying the model's structure. The core structure of the MobileNet-V2 is illustrated in Figure 3.

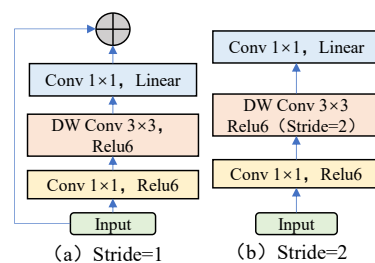


Figure 3. The core structure of the MobileNet-V2.

In summary, MobileNetV2, with its clever lightweight architecture, cleverly integrates residual connection technology while maintaining the model's compactness and efficiency. This design strategy not only optimizes the training stability of the network, promoting better convergence performance, but also enhances recognition accuracy and processing speed. Thanks to these characteristics, MobileNetV2 demonstrates wide applicability while maintaining low model size and memory requirements, making it easy to deploy

in diverse hardware environments, including but not limited to mobile and embedded platforms. Such features make it one of the key technologies driving the development of AI applications in mobile and edge computing.

3. Materials and Methods

To improve the model's classification ability on complex backgrounds, varying sizes, diverse poses, and fine-grained pests, this paper proposes a lightweight crop pest classification method called PestNet, building upon the MobileNet-V2 model. The aim is to achieve higher classification accuracy with lower complexity.

3.1. Adaptive Spatial Group-Wise Enhance (ASGE)

When classifying multiple categories of pests, the model can be disturbed by irrelevant features such as background, leading to inaccurate recognition results. The use of attention mechanisms allows the model to focus on task-relevant information while ignoring other irrelevant information. Unlike the SE attention mechanism [28], which focuses only on global information between channels, the Spatial Group-Wise Enhance (SGE) attention mechanism groups feature maps and considers each group as representing a semantic feature. It generates an attention mask based on the similarity between local and global features, thereby enhancing the spatial distribution of semantic features [29]. Its structure is illustrated in Figure 4a.

Based on these insights, this paper enhances the SGE attention mechanism by incorporating parallel GAP channels and introducing adjustable parameters to balance the influence of GAP and GMP, thus forming the ASGE attention mechanism. This modification aims to enhance the model's capability to represent information more effectively. The structure of the ASGE attention mechanism, as depicted in Figure 4b, involves several key steps:

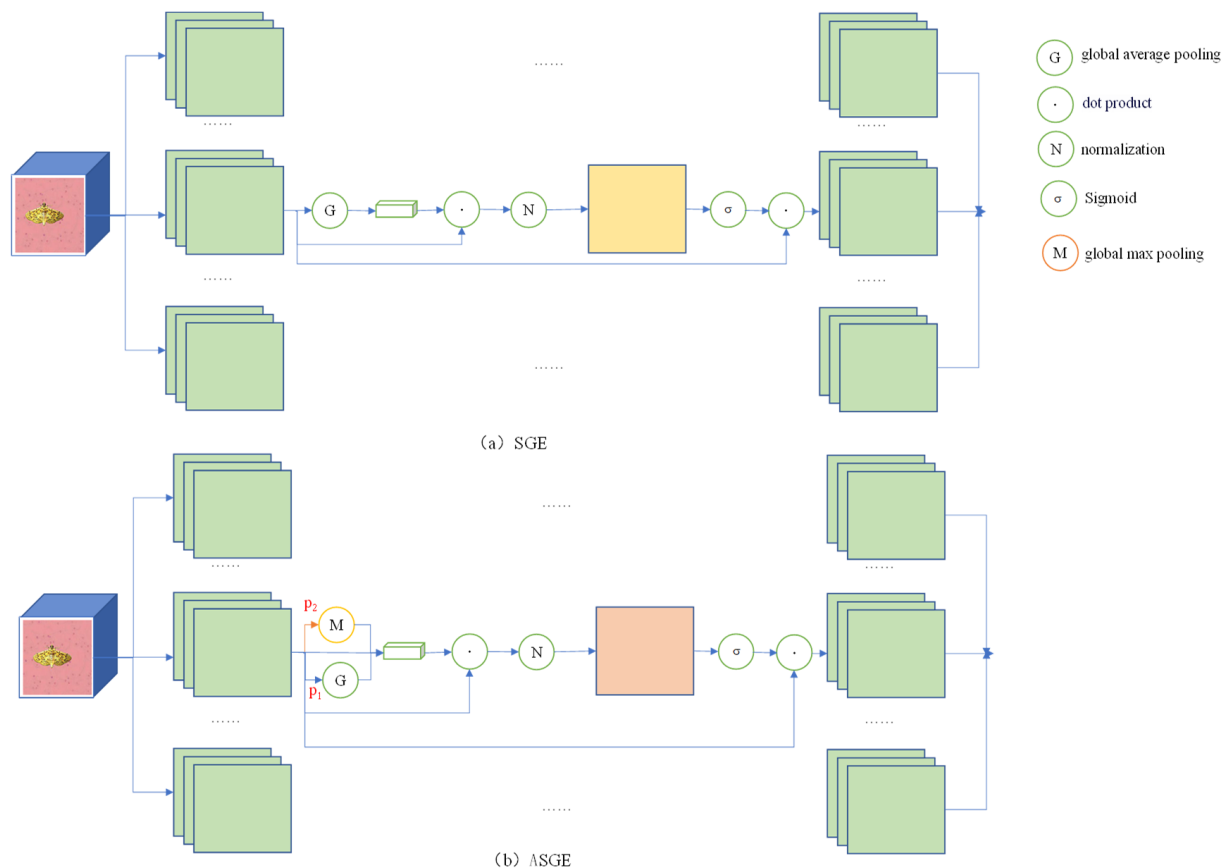


Figure 4. The structure of SGE and ASGE attention mechanisms.

The performance of using attention mechanisms in a model depends on their placement. For example, the possible insertion points for the SGE (Spatial Group-wise Enhance) attention mechanism in the inverted residual blocks of the MobileNet-V2 are shown in Figure 5.

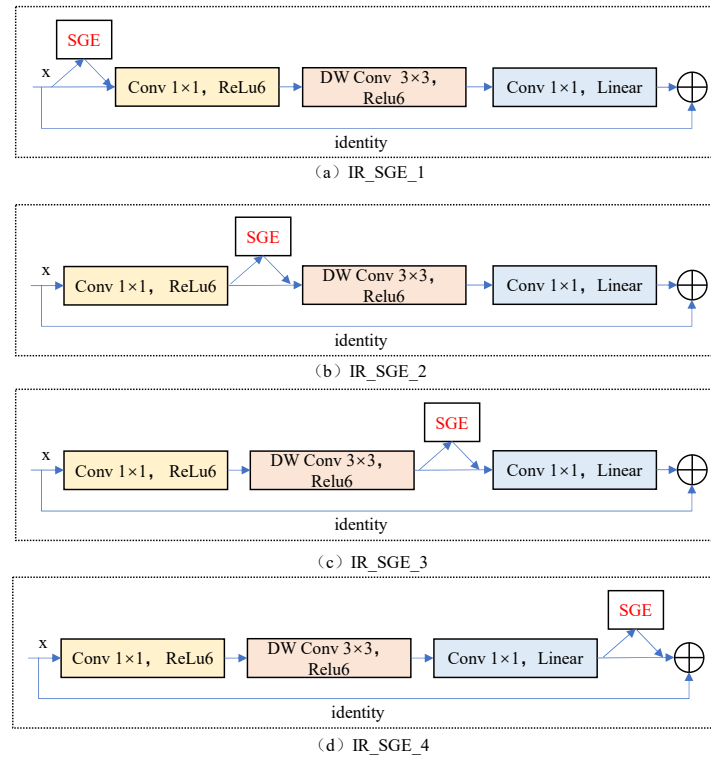


Figure 5. SGE attention mechanism insertion points in the inverted residual blocks.

Figure 5 illustrates four possible points for incorporating the SGE attention mechanism within the inverted residual blocks of the MobileNet-V2 architecture. Subsequent experiments were conducted to compare the classification performance at these different points and to determine the most effective location for integration. The SGE attention mechanism employs Global Average Pooling (GAP) to capture global information from grouped feature maps. However, relying solely on GAP often fails to extract meaningful information from complex inputs [30]. Zhou and colleagues [31] demonstrated that GAP, along with Global Max Pooling (GMP), performs well in feature localization through class activation map analysis. Building on this concept, the CBAM attention mechanism [32] employs a simple linear combination of GAP and GMP to glean richer features.

Firstly, the ASGE attention mechanism divides the input feature map $X \in R^{C \times H \times W}$ into G groups, and each group's feature map is processed separately, denoted as $X_g \in R^{\frac{C}{G} \times H \times W}$.

Then, each group's feature map undergoes both Global Average Pooling (GAP) and Global Max Pooling (GMP) operations, and learnable parameters p_1 and p_2 are introduced to update the weights of the parallel channels. The element-wise multiplication of each group's input feature map with the feature matrices obtained from these operations results in the initial attention mask c , as shown in Equation (1).

$$\begin{aligned} n &= p_1 \cdot F_{GAP}(X_g) + p_2 \cdot F_{GMP}(X_g) \\ c &= n \cdot X_i \end{aligned} \quad (1)$$

Next, the attention mask c is normalized, and learnable scaling and shifting parameters γ and β are introduced, as shown in Equations (2) and (3).

$$\dot{c}_i = \frac{c_i - \mu_c}{\sigma_c + \varepsilon}, \mu_c = \frac{1}{m} \sum_j^m c_j, \sigma_c^2 = \frac{1}{m} \sum_j^m (c_j - \mu_c)^2 \quad (2)$$

$$a_i = \gamma \dot{c}_i + \beta \quad (3)$$

Finally, use the product of the Sigmoid activation function and the input feature of the group as the corresponding output feature, as shown in Formula (4).

$$\dot{X} = X \cdot \sigma(\alpha) \quad (4)$$

3.2. Dual-Branch Feature Fusion Module (DFFM)

GoogLeNet widened the network and increased receptive fields by parallelizing multiple convolutional kernels of different sizes, achieving feature extraction from multi-scale inputs, and enhancing model performance [33]. However, the multi-scale feature fusion structure in GoogLeNet contains a large number of branches and uses regular convolutions for feature extraction, resulting in a high number of parameters and computational complexity, which affects the lightweight nature of the model. Therefore, in this study, deep convolutions were used instead of regular convolutions, and the number of branches was reduced when constructing the feature fusion module. Taking into account the structural characteristics of the MobileNet-V2 model's reverse residual module, four DFFM modules were designed, and the optimal structure was selected through experiments, as shown in Figure 6. The selection of an appropriate DFFM structure mainly follows the subsequent principles, which can better improve the performance of pest identification without excessively increasing the model complexity.

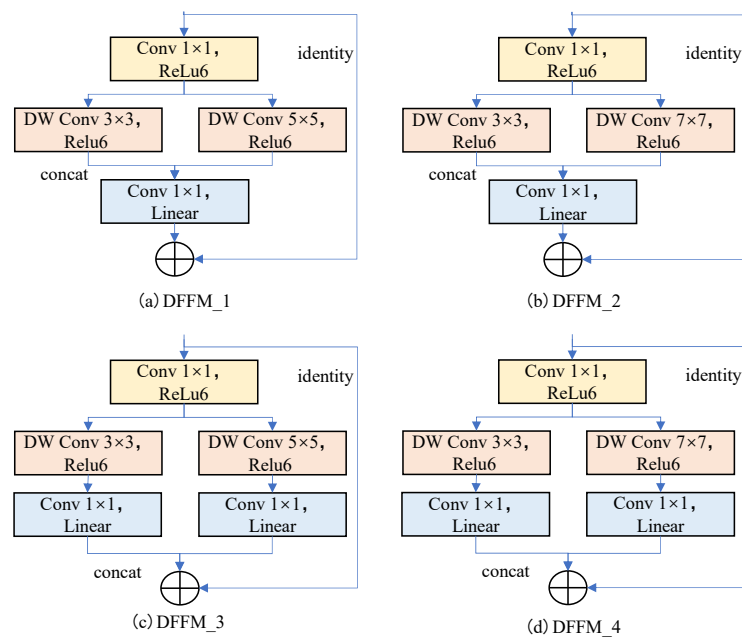


Figure 6. The four structures of the dual-branch feature fusion module (DFFM).

This paper integrates the ASGE attention mechanism and DFFM structure mentioned above to improve the reverse residual module in MobileNet-V2, proposing a new bottleneck structure called Bottleneck_ASGE_DFFM, as illustrated in Figure 7.

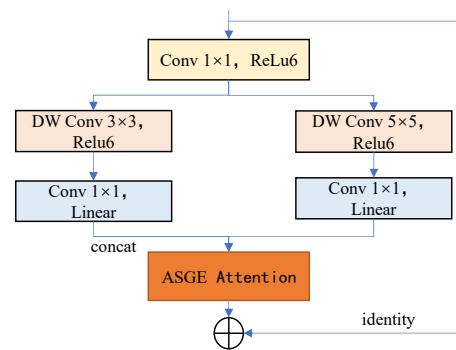


Figure 7. Bottleneck_ASGE_DFFM.

3.3. Optimizing the Activation Function

The Pest37 dataset exhibits significant differences in the quantity of data among different pest categories, making the model prone to overfitting on pests with fewer samples during training. The ReLU activation function can sparsify the model's structure, thereby alleviating overfitting issues, and it is widely used in CNN models. However, the ReLU activation function is not differentiable at zero, which affects the model's performance to some extent. The GELU activation function possesses characteristics such as continuity and stochastic regularity, and the gradient descent process is relatively stable. It demonstrates good generalization and convergence capabilities. Its expression is shown in Equation (5), and its function graph is illustrated in Figure 8.

$$GELU(x) = x\Phi(x) \quad (5)$$

In the Equation (5), $\Phi(x)$ represents the cumulative distribution function of the Gaussian distribution.

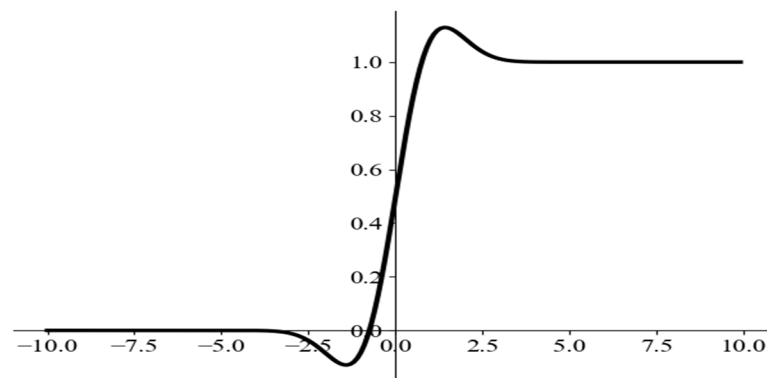


Figure 8. The function graph of GELU.

This paper replaces the ReLU6 activation function in the MobileNet-V2 with the GELU activation function to enhance the stability and generalization ability of model training.

3.4. Model Architecture Adjustments and the Structure of PestNet

Compared to the reverse residual module in MobileNet-V2, the bottleneck structure proposed in this paper, Bottleneck_ASGE_DFFM, can improve the model's classification performance. However, the dual-branch structure, to some extent, increases the model's parameters and computational complexity. To reduce the model's complexity, this paper redefines the internal parameters by adjusting the iteration times of the core bottleneck structure. Through the series of improvements mentioned above, a lightweight pest classification method named PestNet is obtained. The internal parameters of PestNet are shown in Table 2, and the overall structure is depicted in Figure 9. In the table, “->” indicates

the adjustment process (“2->1” means adjusting 2 to 1), t represents the magnification factor of input channels, c represents the number of output channels, n represents the number of iterations for the corresponding operator, and s denotes the convolution operation stride when the corresponding operator is first used, with a stride of 1 in other cases.

Table 2. The internal parameters of PestNet.

Input	Layer	t	c	n	s
$224 \times 224 \times 3$	conv2d	-	32	1	2
$112 \times 112 \times 32$	Bottleneck_ASGE_DFFM	1	16	1	1
$112 \times 112 \times 16$	Bottleneck_ASGE_DFFM	6	24	2->1	2
$56 \times 56 \times 24$	Bottleneck_ASGE_DFFM	6	32	3->1	2
$28 \times 28 \times 32$	Bottleneck_ASGE_DFFM	6	64	4->1	2
$14 \times 14 \times 64$	Bottleneck_ASGE_DFFM	6	96	3->1	1
$14 \times 14 \times 96$	Bottleneck_ASGE_DFFM	6	160	3	2
$7 \times 7 \times 160$	Bottleneck_ASGE_DFFM	6	320	1	1
$7 \times 7 \times 320$	conv2d 1 × 1	-	1280	1	1
$7 \times 7 \times 1280$	avgpool 7 × 7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1 × 1	-	37	-	-

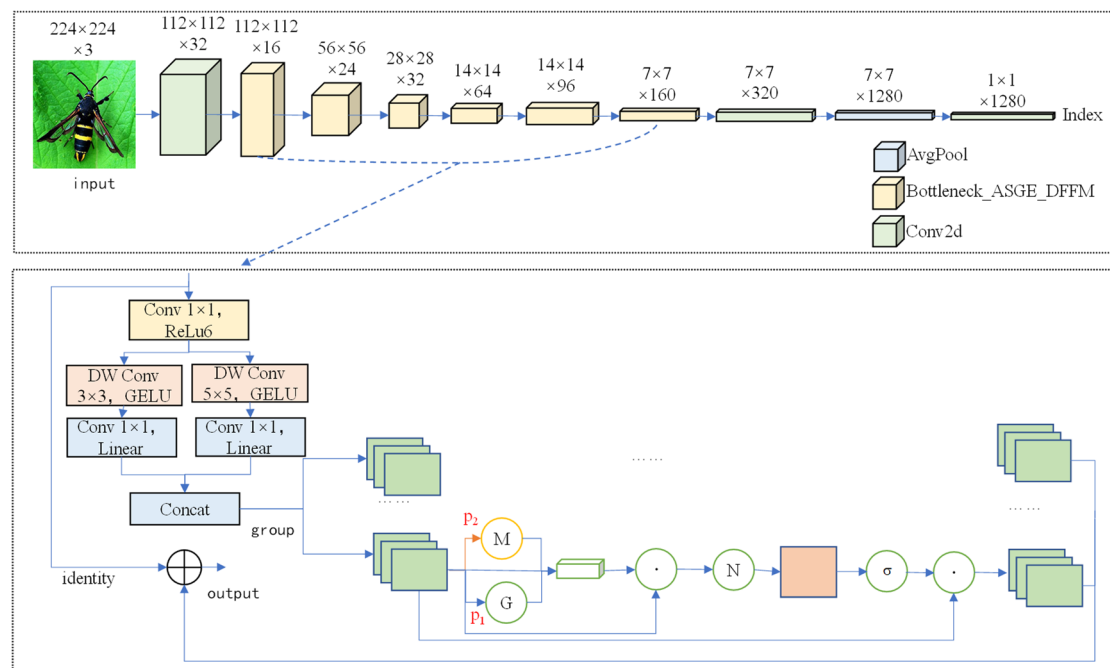


Figure 9. The structure of PestNet.

4. Experiments and Results

The Pest37 dataset is divided into a training set (11,140 images), a validation set (1585 images), and a test set (3166 images) in a ratio of 7:1:2. All input images undergo standardization during the preprocessing stage. Considering the large overall data volume of pest images in Pest37, using offline data augmentation techniques would generate a considerable number of data samples, thereby increasing training time. Therefore, this paper employs online data augmentation techniques to process input samples. In addition to random cropping and random horizontal flipping adopted by the model during training, the mixup data augmentation method is introduced in this paper [34]. To comprehensively evaluate the model’s performance, accuracy, precision, recall, and F1 score are used as metrics for measuring the model’s classification performance, while parameter count, floating-point operations, and training time are used as metrics for evaluating model complexity.

4.1. Experimental Setup

To minimize the influence of experimental environments on model performance and ensure the reliability of experimental results, all experiments in this chapter were conducted on the same system. The hardware and software configurations of this system are shown in Table 3.

Table 3. Experimental Environment.

Hardware	Software
CPU: Intel(R) Core (TM) i5-8400 @2.80 GHz	Ubuntu 18.04
RAM: 32 GB DDR4	CUDA 11.4
GPU: 12GB GeForce GTX 1080 Ti	PyTorch 1.11.0

MobileNet-V2 uses SGD as the optimization algorithm, but SGD has strict requirements on the learning rate and tends to converge slowly, making it prone to local optima and affecting the accuracy of pest classification. Therefore, this paper chooses AdamW, a commonly used optimization algorithm in deep learning, for the model. During the experiments, the input image size is adjusted to 224×224 , the batch size is set to 32, the number of epochs is set to 100, the learning rate is initialized to 0.001, the weight decay coefficient is set to 0.0001, and the cosine annealing learning rate decay strategy is used during iterations. For model comparison, besides MobileNet-V2, other advanced models such as ResNet-50, ResNet-101, DenseNet-121, EfficientNet-B1, ShuffleNet-V2, and MobileNetV3-Large are also experimented with, and relevant evaluation metrics are used to comprehensively assess the pest classification performance of PestNet. The optimization algorithms and data augmentation methods used by all compared models are consistent.

4.2. Optimization Algorithm Comparison Experiment

To verify the effectiveness of the AdamW optimization algorithm, we conducted experiments using the MobileNet-V2 baseline model, where only the type of optimization algorithm was changed. The optimization algorithms compared in the experiment were SGD, Adam, and RMSprop. The variation of model loss on the Pest37 training set during model iteration is shown in Figure 10, and the classification results on the test set are presented in Table 4.

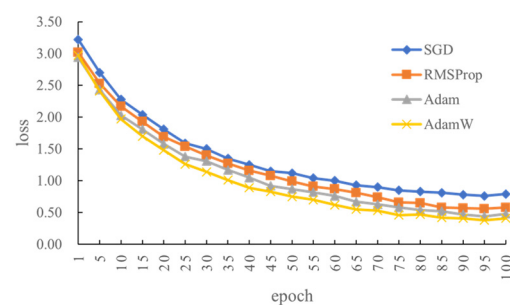


Figure 10. The variation of model loss.

Table 4. Experimental Comparison Using Different Optimization Algorithms.

Optimization	Acc (%)	P (%)	R (%)	F1 (%)
SGD	76.09	77.84	70.84	73.46
RMSProp	82.25	82.05	78.39	79.68
Adam	83.39	83.04	78.90	80.29
AdamW	83.42	83.06	79.60	81.04

The AdamW optimization algorithm modifies the weight decay calculation method based on Adam and integrates the adaptive gradient mechanism and momentum gradient

mechanism from RMSprop. Compared to SGD, AdamW is less sensitive to learning rates and decouples weight decay. As shown in Figure 10, after 100 iterations of training, the loss values of the MobileNet-V2 model using these four optimization algorithms tend to stabilize. Among them, the model using SGD as the optimization algorithm exhibits a slow decline in loss values and poor convergence. In contrast, models using Adam and AdamW optimization algorithms show faster decline rates and smaller loss values compared to those using RMSprop and SGD. The model using AdamW exhibits the best performance in terms of loss value, decline rate, and convergence.

From the experimental results in Table 4, it can be observed that the MobileNet-V2 model using SGD performs poorly on the Pest37 test set, with an accuracy of only 76.09%. Models using RMSprop, Adam, and AdamW optimization algorithms achieve better results compared to SGD. Particularly, the MobileNet-V2 model with the AdamW optimization algorithm performs the best, with accuracy and an F1 score of 83.42% and 81.04%, respectively, representing an improvement of 7.33 and 7.58 percentage points over the model using SGD and outperforming Adam and RMSprop in overall performance. This shows, to some extent, that the AdamW optimization algorithm has better performance and results in deep learning model training compared to optimization algorithms such as SGD by improving the regularization method, adopting an adaptive learning rate, incorporating momentum, avoiding locally optimal solutions, optimizing large datasets, and reducing parameter tuning.

4.3. Determination of Mixup Data Augmentation Hyperparameters

The mixup data augmentation method follows a beta distribution with parameters α and α , and its performance is significantly influenced by the α value. To validate the effectiveness of mixup data augmentation and determine suitable hyperparameters, contrasting experiments were conducted using the MobileNet-V2 baseline model. The experimental results of the Pest37 test set are shown in Table 5.

Table 5. Experiment Results of Mixup Data Augmentation Hyperparameters.

Mixup	α	Acc (%)	P (%)	R (%)	F1 (%)
×	None	83.42	83.06	79.60	81.04
✓	0.1	83.89	83.42	80.53	81.63
✓	0.2	83.83	83.46	80.07	81.48
✓	0.5	84.14	86.37	80.68	82.95
✓	0.8	82.79	84.05	77.58	80.20
✓	1.0	82.69	84.17	78.30	80.66

Note: The “✓” in the table indicates the use of the mixup data augmentation strategy during model training, while “×” indicates no use of the mixup data augmentation strategy.

From the experimental results in Table 5, it can be observed that when the value of α is between 0.8 and 1, using mixup data augmentation did not improve the model’s classification performance. The classification accuracy on the Pest37 test set was lower than that of the original MobileNet-V2 model. However, when α is set to 0.1, 0.2, and 0.5, the models with mixup data augmentation showed improvements in classification accuracy and F1 score compared to the original MobileNet-V2 model. Specifically, when α is set to 0.5, the model achieves the best experimental results. Compared to the MobileNet-V2 baseline model, the model’s classification accuracy, precision, recall, and F1 score improved by 0.72%, 3.31%, 1.02%, and 1.91%, respectively. The experimental results show that the use of the mixup data augmentation method is greatly affected by the hyperparameter α , and it is essential to select a suitable hyperparameter α . When the appropriate hyperparameter α is chosen, there is a good improvement in the generalization ability of the model.

4.4. Impact of Activation Functions on Model Performance

To investigate the effect of replacing the ReLU6 activation function in MobileNet-V2 with the GELU activation function on the model's classification performance, contrasting experiments were conducted with baseline models using the Swish, PReLU, LeakyReLU, Hswish, and ELU activation functions. The experimental results of the Pest37 test set are presented in Table 6.

Table 6. Experimental Results of Activation Functions Comparison.

Activation Function	Acc (%)	P (%)	R (%)	F1 (%)
ReLU6	83.42	83.06	79.60	81.04
Swish	84.49	84.95	80.93	82.42
PReLU	83.45	84.11	79.08	80.95
LeakyReLU	84.33	84.38	80.71	82.17
Hswish	84.37	85.31	81.36	83.01
ELU	81.90	82.36	77.33	79.29
GELU	84.74	85.08	81.67	83.15

The results from Table 6 reveal noticeable differences in pest classification performance when using different types of activation functions on the MobileNet-V2 baseline model. Among them, the model using the ELU activation function experienced varying degrees of decline in accuracy, precision, recall, and F1 score on the Pest37 test set. Replacing the ReLU6 activation function with PReLU slightly improved the accuracy and precision of the model, but the recall and F1 scores decreased. Models using Swish, LeakyReLU, Hswish, and GELU activation functions all exhibited comprehensive improvements in pest classification performance. Compared to other activation functions, the model using the GELU activation function demonstrated the best classification performance, with accuracy and F1 scores of 84.74% and 83.15%, respectively, representing an increase of 1.32 and 2.11 percentage points over the MobileNet-V2. Different activation functions may be suitable for different tasks and datasets, and from the experimental results, it is shown that the advantages of the GELU activation function in the gradient vanishing problem, smoothing and conductivity, nonlinear transformations, approximate constancy, and stochastic regularity can help to improve the training effect and performance of the neural network proposed in this paper.

4.5. Model Performance Evaluation with Attention Mechanisms

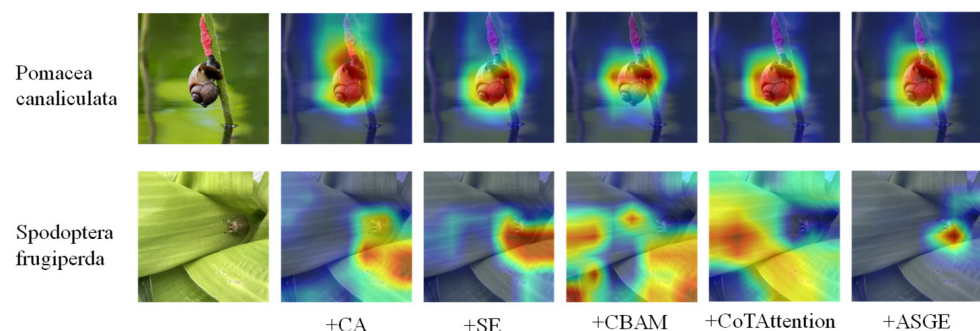
The performance of the model is influenced by the insertion of attention mechanisms into different positions of the MobileNet-V2 inverted residual module. By introducing the SGE attention mechanism into four different positions of the inverted residual module, this section compares the performance of the SGE attention mechanism at different insertion positions through experiments and determines the optimal insertion position. Additionally, experiments were conducted using the improved ASGE attention mechanism on the basis of the optimal insertion position. These were compared with baseline models incorporating SGE, CA, SE, CBAM, and CoTAttention [35], validating the effectiveness of the improved ASGE attention mechanism. The experimental results of the Pest37 test set are shown in Table 7.

From the results in Table 7, it can be observed that introducing the attention mechanism at positions IR_SGE_1 and IR_SGE_2 resulted in poorer performance, with noticeable decreases in all evaluation metrics. When the SGE attention mechanism was introduced after the 3×3 depthwise convolution (IR_SGE_3), there was a slight improvement in accuracy, precision, and F1 score, while recall experienced a slight decrease. Introducing the SGE attention mechanism after the 1×1 channel compression convolution (IR_SGE_4) led to a comprehensive improvement in model performance compared to positions IR_SGE_1, IR_SGE_2, and IR_SGE_3.

Table 7. Experiment with Attention Mechanisms.

Attention	Insertion Position	Acc (%)	P (%)	R (%)	F1 (%)
-	-	83.42	83.06	79.60	81.04
SGE	IR_SGE_1	82.47	82.39	78.21	79.79
SGE	IR_SGE_2	81.90	80.60	77.19	78.48
SGE	IR_SGE_3	84.14	83.42	79.47	81.08
SGE	IR_SGE_4	84.62	84.80	80.85	82.44
ASGE	IR_SGE_4	85.85	85.16	82.38	83.45
CA	IR_SGE_4	84.02	83.37	80.63	81.72
SE	IR_SGE_4	84.52	84.85	81.17	82.75
CBAM	IR_SGE_4	82.69	83.47	77.43	79.48
CoTAttention	IR_SGE_4	84.62	84.84	80.65	82.33

After determining the optimal insertion position, the pest classification performance of the baseline model with different attention mechanisms was compared. Compared to the SGE attention mechanism, introducing the improved ASGE attention mechanism resulted in an increase of 1.23%, 0.36%, 1.53%, and 1.01% in accuracy, precision, recall, and F1 score, respectively, indicating the effectiveness of the improvement in the SGE attention mechanism. Compared to the MobileNet-V2 model, introducing the ASGE attention mechanism resulted in an increase of 2.43% and 2.41% in accuracy and F1 score, respectively. Compared to the baseline models with CA, SE, CBAM, and CoTAttention attention mechanisms, the model with ASGE also achieved better experimental results. To visually demonstrate the performance of the MobileNet-V2 model with the introduced ASGE attention mechanism in classifying pests, this paper utilized the Grad-CAM [36] technique to visualize class activation maps for selected pests and compared them with MobileNet-V2 with CA, SE, CBAM, and CoTAttention, as shown in Figure 11.

**Figure 11.** The class activation maps of models with introduced attention mechanisms.

The class activation maps in Figure 11 reveal that for *Pomacea canaliculata*, which has relatively low background interference and a large main body of the pest, the performance of MobileNet-V2 models with introduced CBAM and SE attention mechanisms is poor, as they only focus on a small part of the pest's main body. The model with the CA attention mechanism focuses on the pest's main body but is also affected by background interference. In contrast, models with CoTAttention and the improved ASGE attention mechanism proposed in this paper perform better.

For *Spodoptera frugiperda*, which has a larger background interference and a smaller main body of the pest, models with CA, SE, CBAM, and CoTAttention attention mechanisms are all heavily affected by the background interference. Among them, the models with CBAM and CoTAttention even fail to respond to the main body of the pest. However, the model with the improved ASGE attention mechanism alleviates the background interference and accurately focuses on the main body of the pest, extracting more effective pest information. This partly verifies the effectiveness of introducing the improved ASGE attention mechanism into the MobileNet-V2 baseline model.

The synthesis of experimental results and analysis, not any introduction of an attention mechanism in the model, can achieve better results. We need to consider the characteristics of the model and the specific location of the introduction. The ASGE attention mechanism proposed in this study can focus on both local and global features, which is more suitable for the pest identification task.

4.6. Model Performance Comparison with Multi-Scale Feature Fusion Structures

To investigate the performance of the four multi-scale feature fusion structures proposed in this paper, comparative experiments were conducted on the MobileNet-V2 baseline model. In addition to using classification performance metrics, parameters count and floating-point operations count were added as evaluation metrics for model complexity. The experimental results of the Pest37 test set are shown in Table 8.

Table 8. Performance of the Model with the Introduction of a Multi-Scale Feature Fusion Structure.

Structure	Acc (%)	P (%)	R (%)	F1 (%)	Parameters (M)	FLOPs (G)
-	83.42	83.06	79.60	81.04	2.27	0.32
DFFM_1	84.37	83.72	80.92	82.04	1.95	0.26
DFFM_2	84.05	83.61	79.26	80.93	2.03	0.28
DFFM_3	85.19	85.02	81.67	83.14	2.47	0.37
DFFM_4	84.43	85.15	80.82	82.63	2.64	0.42

From the results in Table 8, it can be observed that introducing the four multi-scale feature fusion structures designed in this paper all led to improvements in classification accuracy. Among them, introducing the DFFM_2 feature fusion structure resulted in a decrease in recall rate and F1 score. Introducing DFFM_1 led to an increase of 0.95 and 0.66 percentage points in classification accuracy and precision, respectively. Introducing DFFM_3 resulted in a classification accuracy of 85.19%, an improvement of 1.77 percentage points, with only a slight increase of 0.05 G in floating-point operations and 0.2 M in parameters. Introducing DFFM_4 did not significantly improve the classification performance compared to DFFM_3. Considering both classification performance and model complexity, the multi-scale feature fusion structure adopted in PestNet is DFFM_3.

The combined experimental results show that the introduction of a multi-scale fusion structure has a greater potential to improve the model's recognition performance, but it also requires careful design of the specific structure and the location of its introduction. In addition, when designing the multi-scale fusion structure, the characteristics of the original model should also be considered, and the complexity of the model should not be increased excessively. The DFFM structure proposed in this study takes into account the characteristics of the original structure of MobileNet V2, which improves the recognition performance of the model while increasing the complexity of the model slightly and achieving a more balanced result.

4.7. Ablation Study

In this section, we investigated the performance improvements achieved by introducing individual enhancement factors into MobileNet-V2. Furthermore, to further explore the pest classification performance of introducing multiple enhancement factors into the MobileNet-V2 baseline model and the overall performance of PestNet obtained by integrating all improvements, an ablation study was conducted using the Pest37 test set. The results are summarized in Table 9.

From the experimental results in Table 9, it can be observed that incorporating the mixup data augmentation method, replacing the ReLU6 activation function with GELU, and introducing the ASGE attention mechanism into the reverse residual units of the model all improved the model's classification accuracy and F1 score without increasing the number of parameters or floating-point operations. Introducing the multi-scale feature

fusion structure, DFFM_3 increased the model's classification performance while adding a small number of model parameters and floating-point operations.

Table 9. The results of the ablation study.

Improvement	Acc (%)	F1 (%)	Parameters (M)	FLOPs (G)
-	83.42	81.04	2.27	0.32
GELU	84.74	83.15	2.27	0.31
ASGE	85.85	83.45	2.27	0.32
mixup	84.14	82.95	2.27	0.32
DFFM_3	85.19	83.14	2.47	0.37
GELU+ASGE	86.10	84.00	2.27	0.31
GELU+ASGE+mixup	87.27	85.93	2.27	0.31
GELU+ASGE+mixup+DFFM_3	87.81	86.26	2.47	0.36
PestNet	87.62	86.90	1.95	0.20

Combining the GELU and ASGE attention mechanisms, the MobileNet-V2 model achieved an accuracy of 86.10% and an F1 score of 84.00% on the Pest37 test set, surpassing the individual improvements of GELU and ASGE. Further combining GELU, ASGE, and mixup data augmentation, the model's pest classification performance was further enhanced, with an accuracy of 87.27%.

By integrating GELU, ASGE, mixup data augmentation, and the multi-scale feature fusion structure DFFM_3, the model achieved a significant improvement in accuracy and F1 score on the Pest37 test set, with an increase of 4.39 and 5.22 percentage points, respectively, albeit with a slight increase in complexity. Adjusting the model architecture to obtain PestNet, the model achieved an accuracy and F1 score of 87.62% and 86.90% on the Pest37 test set, respectively.

Compared to the baseline model, PestNet reduced the number of parameters and floating-point operations by 14.10% and 37.50%, respectively, while maintaining or even improving classification performance. When using MobileNet-V2 and PestNet for pest image classification, the predicted labels and true labels were statistically analyzed, and visual analysis was conducted using Grad-CAM technology to output class activation maps, as shown in Figure 12.

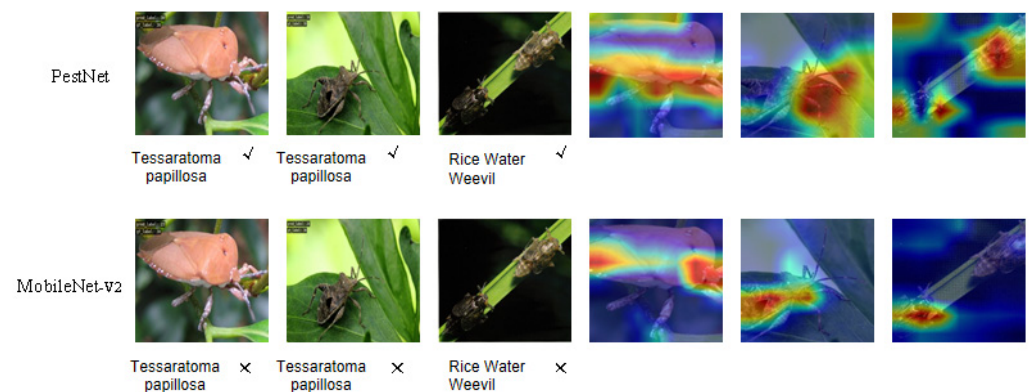


Figure 12. The prediction results and corresponding class activation maps of PestNet and MobileNet-V2.

In Figure 12, ✓ represents correct predictions, while × indicates incorrect ones. The subsequent three columns show the class activation maps corresponding to the first three columns of images. From Figure 12, the following observations can be made: In the first column, the main part of the litchi bug image is clearly visible. Compared to MobileNet-V2, PestNet not only focuses on the head and tail of the pest but also attends to the bottom part and edge curves, which may contribute to PestNet's accurate identification. In the second column, the main part of the litchi bug image is relatively small and subject to background

interference. While MobileNet-V2 focuses on the body of the pest, PestNet concentrates on the head of the pest, resulting in a correct prediction, indicating the importance of the litchi bug's head features in model recognition. In the third column, the image contains two pests of approximately the same size. MobileNet-V2 only focuses on the main part of one pest, while PestNet performs better, focusing on the main parts of both pests. Overall, the experimental results from Table 9 and Figure 12 suggest that the improved PestNet pest classification method is effective.

Synthesizing the results and analysis of the ablation experiments, the various parts of the improvements of MobileNet V2 in this study have shown good results, and the PestNet model proposed by synthesizing all the improvements has achieved good results in terms of pest recognition effect and model complexity. The reduction in model complexity is due to our careful design of the number of iterations of the core structure, and the improvement of the recognition effect is mainly due to our proposed DFFM multi-scale fusion structure and ASGE attention mechanism.

4.8. Performance Comparison between PestNet and Other Advanced Models

To fully evaluate the performance of the improved PestNet in pest classification and model complexity, comparisons are made with classical conventional models such as ResNet-50, ResNet-101, DenseNet-121, and ResNest-50, as well as lightweight models like MobileNetV3-Large, ShuffleNet-V2, and EfficientNet-B1. During training, the accuracy changes in each comparison model on the Pest37 validation set are depicted in Figure 13, while the classification performance on the Pest37 test set and the model complexity are presented in Tables 10 and 11, respectively.

Table 10. Classification Performance of Various Models on the Pest37 Test Set.

	Model	Acc (%)	P (%)	R (%)	F1 (%)
Lightweight Models	ShuffleNet-V2	83.73	85.28	80.19	82.33
	MobileNetV3-Large	83.89	85.16	80.31	82.25
	EfficientNet-B1	84.24	85.43	80.31	82.26
	PestNet	87.62	89.82	84.71	86.90
Conventional Models	ResNet-50	84.08	84.79	80.95	82.44
	ResNet-101	85.60	87.76	82.38	84.54
	DenseNet-121	83.95	86.08	80.15	82.59
	ResNest-50	87.49	89.37	84.68	86.62

Table 11. Comparison of Model Complexity.

	Model	Parameters (M)	FLOPs (G)	Training Time
Lightweight Models	ShuffleNet-V2	5.42	0.59	1 h 10 min
	MobileNetV3-Large	4.25	0.23	1 h 13 min
	EfficientNet-B1	6.56	0.59	2 h 32 min
	PestNet	1.95	0.20	1 h 19 min
Conventional Models	ResNet-50	23.58	4.12	2 h 15 min
	ResNet-101	42.58	7.85	3 h 32 min
	DenseNet-121	6.99	2.88	2 h 27 min
	ResNest-50	25.51	5.41	6 h 11 min

The curve graph in Figure 13 illustrates that during the training process, ResNest-50 and PestNet exhibited relatively high accuracy, with a rapid increase in accuracy. ResNet-50, ResNet-101, ShuffleNet-V2, MobileNetV3-Large, and EfficientNet-B1 showed comparable accuracy rates with similar trends. However, DenseNet-121 performed poorly in terms of accuracy.

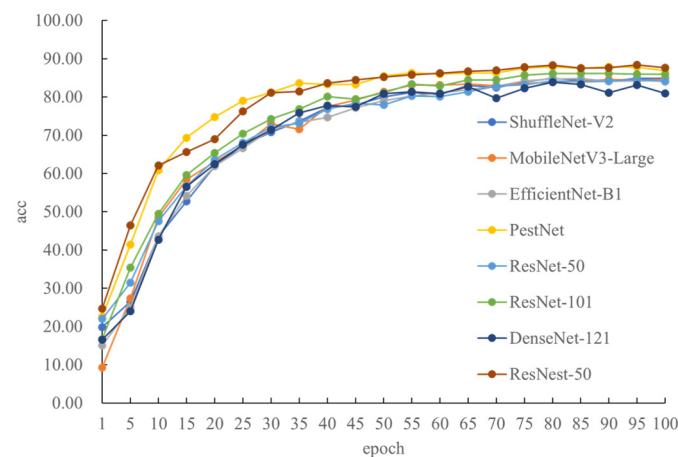


Figure 13. The accuracy of each model during iterations on the Pest37 validation set.

From the results in Tables 10 and 11, it can be observed that, compared to conventional models, lightweight models have relatively fewer parameters and floating-point operations. Among the conventional models, DenseNet-121 has lower complexity, but its classification performance is poorer. ResNeSt-50 achieves good pest classification results, but it has a large number of parameters and floating-point operations and requires a long training time. In the lightweight models, PestNet proposed in this paper achieves comparable classification performance to the ResNeSt-50 conventional model, with a reduction of 92.36% in parameters and 96.30% in floating-point operations, accompanied by a substantial reduction in training time. Compared to EfficientNet-B1, ShuffleNet-V2, and MobileNetV3-Large models, PestNet demonstrates the best classification performance with the fewest parameters and floating-point operations. Additionally, its training time is shorter than that of EfficientNet-B1 and slightly higher than ShuffleNet-V2 and MobileNetV3-Large. In conclusion, the PestNet proposed in this paper exhibits high accuracy and low complexity in pest classification, making it suitable for quick and accurate pest identification, especially in resource-constrained environments and portable devices.

From the experimental results and analysis, it can be seen that the simple transplantation of a model that performs well on other tasks to another specific task may not necessarily achieve good results, and it is also necessary to improve and enhance the model in many aspects to adapt to a specific task, in which ideas such as the attention mechanism and multi-scale fusion play an important role.

5. Conclusions

Our research focuses on agricultural pest classification and addresses the challenges of low accuracy and high complexity in existing models. We constructed the Pest37 dataset, tailored specifically for pest classification tasks. We improved the MobileNet-V2 baseline model by enhancing optimization algorithms, data augmentation techniques, activation functions, attention mechanisms, multi-scale feature fusion, and architectural adjustments. This resulted in a lightweight agricultural pest classification method named PestNet. We conducted multiple experiments to validate the effectiveness of both individual and combined improvements in enhancing model performance. We compared PestNet with lightweight and conventional models, evaluating classification performance metrics such as accuracy, precision, recall, and F1 score. Additionally, we measured model complexity in terms of parameters, floating-point operations, and training time. The results confirmed the superior performance and potential applications of the improved PestNet in agricultural pest image classification tasks. PestNet effectively reduces the complexity of the MobileNet-V2 model while enhancing its performance in classifying agricultural pests. This research lays the foundation for further advancements in methods for classifying agricultural pests. In future research, we will deploy the PestNet model on a resource-constrained Raspberry

Pi to test the effectiveness of the model proposed in this study when used in practice. Moreover, we will continue to pay attention to the popular development directions in deep learning and actively introduce them to crop pest recognition.

Author Contributions: Methodology, H.P. and H.X.; software, H.X. and G.S.; validation, H.P., H.X. and G.S.; formal analysis, H.L.; investigation, H.L. and M.L.; resources, G.S.; data curation, M.L.; supervision, X.G.; funding acquisition, X.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Guangzhou Science and Technology Plan Project (Grant No. 2024E04J1242, 2023B01J0046), Dongguan Agricultural Plant Quarantine Pests and Diseases Monitoring and Inspection Project (Grant No. WT-202304-39).

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: Author Huiming Xu was employed by the company Zhuhai Wanlida Electric Automation Company with Limited Liability. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

- Cheng, X.; Zhang, Y.; Chen, Y.; Wu, Y.; Yue, Y. Pest identification via deep residual learning in complex background. *Comput. Electron. Agric.* **2017**, *141*, 351–356. [\[CrossRef\]](#)
- Geiger, F.; Bengtsson, J.; Berendse, F.; Weisser, W.W.; Emmerson, M.; Morales, M.B.; Ceryngier, P.; Liira, J.; Tschardtke, T.; Winqvist, C.; et al. Persistent negative effects of pesticides on biodiversity and biological control potential on European farmland. *Basic Appl. Ecol.* **2010**, *11*, 97–105. [\[CrossRef\]](#)
- Liu, H.; Lee, S.-H.; Chahl, J.S. A review of recent sensing technologies to detect invertebrates on crops. *Precis. Agric.* **2017**, *18*, 635–666. [\[CrossRef\]](#)
- Fina, F.; Birch, P.; Young, R.; Obu, J.; Faithpraise, B.; Chatwin, C. Automatic plant pest detection and recognition using k-means clustering algorithm and correspondence filters. *Int. J. Adv. Biotechnol. Res.* **2013**, *4*, 189–199.
- Xie, C.; Zhang, J.; Li, R.; Li, J.; Hong, P.; Xia, J.; Chen, P. Automatic classification for field crop insects via multiple-task sparse representation and multiple-kernel learning. *Comput. Electron. Agric.* **2015**, *119*, 123–132. [\[CrossRef\]](#)
- Ali, H.; Lali, M.I.; Nawaz, M.Z.; Sharif, M.; Saleem, B.A. Symptom based automated detection of citrus diseases using color histogram and textural descriptors. *Comput. Electron. Agric.* **2017**, *138*, 92–104. [\[CrossRef\]](#)
- Ebrahimi, M.; Khoshtaghaza, M.H.; Minaei, S.; Jamshidi, B. Vision-based pest detection based on SVM classification method. *Comput. Electron. Agric.* **2017**, *137*, 52–58. [\[CrossRef\]](#)
- Bisgin, H.; Bera, T.; Ding, H.; Semey, H.G.; Wu, L.; Liu, Z.; Barnes, A.E.; Langley, D.A.; Pava-Ripoll, M.; Vyas, H.J. Comparing SVM and ANN based machine learning methods for species identification of food contaminating beetles. *Sci. Rep.* **2018**, *8*, 6532. [\[CrossRef\]](#)
- Pattnaik, G.; Parvathy, K. Machine learning-based approaches for tomato pest classification. *TELKOMNIKA (Telecommun. Comput. Electron. Control)* **2022**, *20*, 321–328. [\[CrossRef\]](#)
- Zhang, C.; Ye, Y.; Xiao, D.; Qi, L.; Yin, J. Rapid detection and identification of major vegetable pests based on machine learning. *Int. J. Wirel. Mob. Comput.* **2022**, *22*, 223–235. [\[CrossRef\]](#)
- LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [\[CrossRef\]](#)
- Li, Y.; Wang, H.; Dang, L.M.; Sadeghi-Niaraki, A.; Moon, H. Crop pest recognition in natural scenes using convolutional neural networks. *Comput. Electron. Agric.* **2020**, *169*, 105174. [\[CrossRef\]](#)
- Chen, J.; Chen, J.; Zhang, D.; Sun, Y.; Nanehkaran, Y.A. Using deep transfer learning for image-based plant disease identification. *Comput. Electron. Agric.* **2020**, *173*, 105393. [\[CrossRef\]](#)
- Tetila, E.C.; Machado, B.B.; Astolfi, G.; de Souza Belete, N.A.; Amorim, W.P.; Roel, A.R.; Pistori, H. Detection and classification of soybean pests using deep learning with UAV images. *Comput. Electron. Agric.* **2020**, *179*, 105836. [\[CrossRef\]](#)
- Ayan, E.; Erbay, H.; Varçın, F. Crop pest classification with a genetic algorithm-based weighted ensemble of deep convolutional neural networks. *Comput. Electron. Agric.* **2020**, *179*, 105809. [\[CrossRef\]](#)
- Khanramaki, M.; Asli-Ardeh, E.A.; Kozegar, E. Citrus pests classification using an ensemble of deep learning models. *Comput. Electron. Agric.* **2021**, *186*, 106192. [\[CrossRef\]](#)
- Wu, J.; Liang, S.; Bi, N.; Tan, J. Attention Embedding ResNet for Pest Classification. In Proceedings of the International Conference on Pattern Recognition and Artificial Intelligence, Cham, Switzerland, 2 June 2022; pp. 591–602.
- Wang, K.; Chen, K.; Du, H.; Liu, S.; Xu, J.; Zhao, J.; Chen, H.; Liu, Y.; Liu, Y. New image dataset and new negative sample judgment method for crop pest recognition based on deep learning models. *Ecol. Inform.* **2022**, *69*, 101620. [\[CrossRef\]](#)

19. Peng, H.; Xu, H.; Gao, Z.; Zhou, Z.; Tian, X.; Deng, Q.; He, H.; Xian, C. Crop pest image classification based on improved densely connected convolutional network. *Front. Plant Sci.* **2023**, *14*, 1133060. [[CrossRef](#)]
20. Guan, H.; Fu, C.; Zhang, G.; Li, K.; Wang, P.; Zhu, Z. A lightweight model for efficient identification of plant diseases and pests based on deep learning. *Front. Plant Sci.* **2023**, *14*, 1227011. [[CrossRef](#)]
21. Xia, W.; Han, D.; Li, D.; Wu, Z.; Han, B.; Wang, J. An ensemble learning integration of multiple CNN with improved vision transformer models for pest classification. *Ann. Appl. Biol.* **2023**, *182*, 144–158. [[CrossRef](#)]
22. Wei, D.; Chen, J.; Luo, T.; Long, T.; Wang, H. Classification of crop pests based on multi-scale feature fusion. *Comput. Electron. Agric.* **2022**, *194*, 106736. [[CrossRef](#)]
23. Hechen, Z.; Huang, W.; Yin, L.; Xie, W.; Zhao, Y. Dilated-Windows-based Vision Transformer with Efficient-Suppressive-self-attention for insect pests classification. *Eng. Appl. Artif. Intell.* **2024**, *127*, 107228. [[CrossRef](#)]
24. Ashwinkumar, S.; Rajagopal, S.; Manimaran, V.; Jegajothi, B. Automated plant leaf disease detection and classification using optimal MobileNet based convolutional neural networks. *Mater. Today Proc.* **2022**, *51*, 480–487. [[CrossRef](#)]
25. Xing, S.; Lee, H.J. Crop pests and diseases recognition using DANet with TLDP. *Comput. Electron. Agric.* **2022**, *199*, 107144. [[CrossRef](#)]
26. 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* **2017**, arXiv:1704.04861.
27. 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*, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
28. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
29. Li, B.; Liu, Y.; Wang, X. Gradient harmonized single-stage detector. In *Proceedings of the AAAI conference on Artificial Intelligence*, Hong Kong, China, 17 July 2019; pp. 8577–8584.
30. Qin, Z.; Zhang, P.; Wu, F.; Li, X. Fcanet: Frequency channel attention networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Hangzhou, China, 22 December 2020; pp. 783–792.
31. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, 17–21 June 2016; pp. 2921–2929.
32. Sanghyun, W.; Jongchan, P.; Joon-Young, L.; In, S. CBAM: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV)*, Munich, Germany, 8–14 September 2018.
33. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, 17–21 June 2016; pp. 1–9.
34. Zhang, H.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv* **2017**, arXiv:1710.09412.
35. Pham, Q.; Liu, C.; Sahoo, D.; Steven, H. Contextual transformation networks for online continual learning. In *Proceedings of the International Conference on Learning Representations*, Virtual, 26 April–1 May 2020.
36. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, Atlanta, GA, USA, 22–29 October 2017; pp. 618–626.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.