MS-ALN: Multiscale Attention Learning Network for Pest Recognition

FUXIANG FENG, HANLIN DONG, YOUMEI ZHANG, YU ZHANG, BIN LI
School of Mathematics and Statistics, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353.
Corresponding author: Youmei Zhang (e-mail: zhangyoumei@qlu.edu.cn).

ABSTRACT Complex backgrounds, occlusions, and non-uniform classes present great challenges to pest recognition in practical applications. In this paper, we propose a multiscale attention learning network to address these problems. This network recursively locates discriminative regions and learns region-based feature representation in four branches. Three newly designed modules, which are target localization, attention detection, and attention removal connect two feature extracting sub-networks in adjacent branches to generate images of different scales. The target localization and attention detection modules locate the discriminative regions to filter out complex backgrounds while the attention removal module randomly removes the discriminative region to encourage the model to tackle occlusions. Thereafter, the parameter-shared classification sub-network follows the feature extracting sub-network in every branch for pest recognition. A decoupled learning strategy is adopted to address the problem of non-uniform classes. We experimented on the widely used IP-102 dataset and achieved state-of-the-art performance.

INDEX TERMS Pest Recognition, Multiscale Attention Learning Network, Target Localization Module, Attention Detection Module, Attention Removal Module.

I. INTRODUCTION

Recognition of pest categories plays an important role in pest control and agricultural management. Because of high similarity between different pests, only specialists can accurately distinguish them, which greatly reduces the efficiency of pest control and agricultural management. With the development of deep learning and large-scale pest datasets, pest recognition based on convolutional neural networks (CNNs) is gaining increasing attention in intelligent agriculture and computer vision [1]–[5]. Despite recent progress in CNNs, pest recognition remains challenging because of problems such as complex backgrounds, occlusions, and uneven distributions.

As shown in Fig. 1, complex backgrounds affect target feature extraction and decrease pest recognition accuracy. Filtering background information and strengthening target features improve pest recognition. Inter-class similarity and intra-class differences of pests are also obstacles to this task. Different types of pests are highly similar in appearance whereas congeneric pests show different morphologies at different growth stages. Furthermore, pests are usually small and easily shielded by leaves, which demands a high standard for model robustness. Traditional recognition algorithms cannot effectively extract pest characteristics and accurately classify them because of these challenges. Exploring the multiple fine-grained characteristics of pests improves recognition accuracy. In addition, the imbalanced distribution of pest numbers also hinders pest recognition. Limited to geographical distribution and growth environment, it is hard to capture some classes of pests whereas others are encountered frequently. Therefore, pest datasets always exist in long-tailed distribution.

Existing studies mostly designed approaches to tackle one of the above pest recognition problems, thereby restricting the recognition performance. Our study designs a coarse-to-fine pest recognition architecture named multiscale attention learning network (MS-ALN), which comprehensively
considers all these challenges. The architecture primarily consists of three modules, which are the target localization module (TLM), attention detection module (ADM), and attention removal module (ARM). The TLM filters out most of the backgrounds and locates the object regions. These object regions are further divided into multiple discriminative patches by ADM for fine-grained feature extraction. On this basis, we also proposed an ARM to encourage the model to learn multiple high response regions, thus improving the robustness of the model to occlusions. To address the problem of long-tailed distribution in pest data, a decoupling learning strategy is adopted to further improve the performance.

The contributions of our study are summarized as follows:

1. We design MS-ALN to tackle complex backgrounds, occlusions, and uneven distribution of data in pest recognition.

2. TLM effectively enhances the features in the target location and filters background information. ADM and ARM further encourage the networks to enhance fine-grained features that distinguish pest categories, thus extracting stable features.

3. To our best knowledge, we make the first attempt to tackle the non-uniform class problem by introducing a decoupling strategy into the field of pest recognition.

4. We conduct experiments on a challenging dataset, IP102, and obtain state-of-the-art performance with an accuracy of 74.61%, which demonstrates the effectiveness of the proposed method.

II. RELATED WORKS

Traditional recognition methods: Most traditional pest recognition algorithms use classifiers such as support vector machine and K-means based on handcrafted features. Ebrahimi et al. [7] used image processing techniques to extract color and regional features, and further used a support vector machine with different kernel functions to classify images based on these features. Fina et al. [8] extracted the variant distinctive attributes between pests and their habitat and further recognized plant pests using the K-means clustering algorithm. Larios et al. [9] improved the feature extraction strategy based on SIFT. The authors adopted the bag-of-features approach [10] to extract region-based features of stonfly larvae and represented those regions as SIFT vectors. Subsequently, they classified the features via ensemble classification algorithms. The above methods classify pests based on hand-crafted features, which is cumbersome and time consuming. Samanta et al. [11] designed an algorithm to detect tea insect pests based on artificial neural networks. Incremental backpropagation learning networks were constructed to realize automatic feature extraction. Traditional recognition algorithms require manual extraction of image features, so these methods only perform well on pest datasets with few classes and are not conducive to large-scale application.

CNN-based recognition methods: CNN-based recognition algorithms have shown great success in various computer vision tasks. Some researchers explored the application of CNN architecture in pest recognition. Alfarisy et al. [12] designed CaffeNet for paddy pest recognition and realized precision detection for both paddy pests and diseases. Inspired by residual block [13], Ren et al. [14] designed a feature reuse residual block to extract more comprehensive information and the block improved pest recognition. Ayan et al. [15] proposed a GAEnsemble comprising multiple CNN branches to improve the recognition accuracy. The prediction probabilities of different branches were weighted as final results by a genetic algorithm to enable this method to consider multiple models to make decisions. Ung et al. [16] integrated attention, feature pyramid and fine-grained models together and proposed a new pest recognition method that effectively dealt with the problem of high similarity between different pests. Although these methods improved the accuracy of the models for pest recognition to a certain extent, the performance of the models can still be improved. Different from the method in [16], we improve the generalization ability of the model by encouraging the model to find multiple highly responsive regions and fine-tune the classifier of the model to solve the problems caused by uneven data distribution.

Recently, some researchers explored the use of data preprocessing to address the problems of complex backgrounds and occlusions. To reduce environmental interference, Liu et al. [17] used a global contrast region-based approach for saliency map generation and then located pests using this saliency map. In [18], channel and spatial attention modules were introduced to further locate pest parts. The authors utilized a region proposal network (RPN) to distinguish features of pests from backgrounds. Li et al. [19] proposed two methods to remove complex backgrounds in situations where high similarity and large differences between pests and backgrounds exist. Reza et al. [20] considered the variability of the environment and used data augmentations to improve the generalization ability and stability of the network. An integration strategy combining the saliency method and CNN was proposed in [21]. The authors used different saliency methods to enhance the data and increased the importance of pest information in images. Aiming at light instability objects in the field, Chen et al. [22] proposed a three-step deep learning method to identify and count wheat mites. They solved the problems caused by environmental factors in some way.

The above methods trained CNNs end-to-end and thus avoiding the cumbersome handcrafted feature extraction process. However, they did not consider the long-tailed distribution of pest data; therefore, the performance of models can be improved.

Long-tailed learning methods: The long-tailed distribution of datasets phenomenon is common in natural scenes. Owing to the habits and diversity of pests, the long-tailed phenomenon is particularly common in pest datasets. Many studies tackled the long-tailed distribution of pest datasets [23], [24]. Wang et al. [23] proposed a dynamic feature weighting method to re-weight head and tail classes based on feature centroids. Yang et al. [25] claimed a novel con-
volitional rebalancing network to classify rice pests. This network extracted more comprehensive features from unbalanced datasets using the convolutional rebalancing module. Although these algorithms alleviated the problems caused by the long-tailed distribution of pest data, most of the existing methods are complicated, which hinders their practical application. In contrast, our proposed method does not add any additional parameters while fine-tuning the model classifier during training.

Some recent research has been carried out to solve the model learning problem caused by long-tailed distribution. Among the solutions, re-sampling and re-weighting have been widely applied with success. Mainly, re-sampling methods [26], [27] reset the sampling strategy so that the adjusted strategy selects more tail samples. In this manner, the effect of uneven sample distribution will be offset as much as possible. Re-weighting methods [28], [29] apply different parameters to the head and tail classes by a loss function. In addition, other methods such as fine-tuning [30], meta-learning [31] and knowledge transfer learning [32], [33], etc. are popular for long-tailed distributions. Although the above methods can improve the model performance on tail classes, they also have problems such as difficulty in algorithm designing and model training, as well as the risk of damaging normal feature learning. Decoupled learning [34] proposed in recent years performs outstandingly on the long-tailed problem; it decouples feature learning and classifier training. In the following research, we conduct comparative experiments to evaluate the decoupled learning and traditional methods.

III. METHOD

In this study, we propose a coarse-to-fine convolutional neural network named MS-ALN to tackle the challenges of pest recognition. The network primarily consists of three modules: TLM, ADM, and ARM for fine-grained feature extraction. As Fig. 2 shows, coarse features \( F_r \) are firstly extracted by the CNN architecture and are further fed to TLM to locate pest objects. The located objects \( X_o \) are cropped for fine-grained feature extraction. ADM is then designed to guide the network to focus on important parts based on features \( F_o \). The important parts \( X_p \) are used to further extract features \( F_p \). To encourage the network to learn multiple high response regions, we design ARM to randomly delete discriminative regions and use the CNN architecture to further extract features \( F_d \). Finally, we obtain four branches to extract features with different scales for classification. The CNN feature extraction architecture and classifiers in different branches share parameters. To deal with the problem of long-tailed distribution of pest data, a decoupling strategy is introduced to the model training process. By separating classifier training and feature learning processes, the effect of uneven distribution can be effectively alleviated.

A. MULTISCALE ATTENTION LEARNING NETWORK

1) Target Localization Module

TLM locates targets and crops the corresponding regions to filter the backgrounds. Features with higher response contain more information and thus are more critical to classification. [35], [36] proved that the aggregation of convolution features is beneficial in describing the position of objects. Therefore, we design two branches in TLM and clip the overlap regions output by the branches to obtain the pest patch. The TLM structure is shown in Fig. 3.

Coarse feature maps extracted from raw image \( X_r \) are defined as \( F_r \) with size \( C \times H \times W \), where \( C, H, W \) indicate the number of feature channels, height, and width. The coarse feature extraction process is formulated as Eq. 1:

\[
\text{Loss}_r = \sum_{i=1}^{n} w_i \cdot \text{Loss}_r^i
\]

where \( w_i \) are weights assigned to each class, \( \text{Loss}_r^i \) is the loss function for class \( i \).
The shape of \( R \) can well represent the data characteristics of the raw image, and the spatial data in the feature maps are distributed in multiple channels. Therefore, we further obtain characteristic response map \( R \) by summing the data in the feature maps along the channel dimension. The size of \( R \) is \( 1 \times H \times W \), as shown in Eq. 2:

\[
F_r^{C \times H \times W} = f_e(X_r)
\]

where \( f_e(\cdot) \) represents a series of convolution and pooling layers in feature extraction network.

\( F_r \) can well represent the data characteristics of the raw image, and the spatial data in the feature maps are distributed in multiple channels. Therefore, we further obtain characteristic response map \( R \) by summing the data in the feature maps along the channel dimension. The size of \( R \) is \( 1 \times H \times W \), as shown in Eq. 2:

\[
R^{1 \times H \times W} = \sum_{i=1}^{C} F_r^i
\]

where \( F_r^i \) represents the \( i \)-th feature map of the corresponding channel.

As the response map \( R \) is obtained through aggregating feature maps, it contains data distribution on all channels. More data in the response map imply denser information in corresponding areas and these regions are more likely to be the target regions. We use global average pooling (GAP) to calculate the information threshold \( \theta \) to judge whether the area contains pests. \( \theta \) is calculated according to Eq. 3:

\[
\theta = GAP(R)
\]

\( \theta \) is used to divide the response map into \( R_p \), which represents the pest regions with positive values. As shown in Eq. 4

\[
R_p = \{(x, y) \mid (x, y) \in R, R(x, y) > \theta\}
\]

The shape of \( R_p \) is irregular, so to filter part of the background information and obtain the area where the pest is present, we further obtain a rectangular clipping box by computing the coordinates in \( R_p \). We determine the rectangle by calculating two points:

\[
x_{lt} = \min(x_{R_p}), y_{lt} = \min(y_{R_p})
\]

\[
x_{rb} = \max(x_{R_p}), y_{rb} = \max(y_{R_p})
\]

where \( (x_{lt}, y_{lt}) \) and \( (x_{rb}, y_{rb}) \) represent the top left and right bottom corners of the rectangle. \( x_{R_p} \) and \( y_{R_p} \) represent the coordinates of all coordinate points in the high response region.

The combination of features in different layers provides more comprehensive semantic information, thus improving the object localization accuracy. Therefore, we feed feature maps in the last layer and penultimate layer of TLM to localize the target. TLM will output two rectangles according to the above process. We crop the overlap region between the two rectangles as shown in Eq. 6:

\[
x_{lt}' = \max(x_{lt}^1, x_{lt}^2), y_{lt}' = \max(y_{lt}^1, y_{lt}^2)
\]

\[
x_{rb}' = \max(x_{rb}^1, x_{rb}^2), y_{rb}' = \max(y_{rb}^1, y_{rb}^2)
\]

where \( (x_{lt}^1, y_{lt}^1) \) and \( (x_{lt}^2, y_{lt}^2) \) represent the coordinates of the top left and right bottom corners of the overlap region. The coordinate data with superscript 1 and 2 represent the clipping coordinates obtained from the feature maps of the penultimate layer and the last layer in the feature extraction network, respectively.

After determining the final cutting coordinates, we obtain the pest target patch \( X_o \) from the raw image \( X_r \) as the input of the second branch.
2) Attention Detection Module

TLM filters the backgrounds and crops the patch for fine-grained feature extraction. Discriminative regions in the patch are more useful in improving the fine-grained feature capability. We designed ADM, which utilizes anchors to detect these discriminative regions. Figure 4 illustrates the structure of ADM.

The object patch $X_o$ is fed to the feature extraction network to generate $F_o$ and further obtain object response map $R_o$. The calculation formulas of $F_o$ and $R_o$ are similar to Eq. 1 and Eq. 2. Then, we set $n$ groups of multiscale anchors represented as $m_1, m_2, \ldots, m_n$ to detect the discriminative regions. By sliding these anchors on the pest patch, we obtain $n$ groups of response values according to Eq. 7:

$$R_{Att}^i = g_i(R_o), i = 1, 2, \ldots, n$$

where $R_{Att}^i$ represents the i-th group response values. $g_i(\cdot)$ denotes the anchors with ratio and scale $m_i$.

To cope with the redundant areas that result from overlapping, we adopt a non-maximum suppression (NMS) method to select a fixed number of bounding boxes $(x_i, y_i)$:

$$(x_i^0, y_i^0, x_i^1, y_i^1) = NMS(R_{Att}, IOU, s), i = 1, 2, \ldots, s$$

where $s$ represents the number of windows for final output. $R_{Att}$ represents response values. $IOU$ indicates the threshold used in NMS. $(x_i^0, y_i^0)$ and $(x_i^1, y_i^1)$ represent the top left and bottom right coordinates of the i-th window, respectively.

To extract more stable fine-grained features, we further clip and magnify these high response regions to obtain the object parts $X_p$ and take them as input to the third branch, where $X_{ip}$ denotes the i-th object part corresponding to the i-th window coordinates, $i = 1, 2, \ldots, s$. Since all these parts are cropped from high response regions, they contain more foreground information compared with those from random cropping.

3) Attention Removal Module

Pest images captured from fields invariably have occlusion problems. In this case, the trained model may be dominated by a single strong feature which reduces the robustness of the model to occlusion. Random erasing [37] is the most widely used method to tackle occlusions. It enhances the stability of the model by feeding randomly deleted images into training. However, because it is purposeless, it cannot effectively guide the model to learn multiple feature regions.

To encourage the model to learn multiple identifying parts and flexibly deal with the occlusions, we design an ARM. ARM removes highly identifiable object parts acquired from ADM, which is more goal-oriented and effective than random erasing.

This module randomly selects one of the high response regions’ output using ADM and generates an attention removal mask $M$. We then conduct Hadamard product on the mask $M$ and the object patch $X_o$ to obtain the attention-deletion image $X_d$, which drops a high response region. Training the network with high response removal images improves the feature extraction and noise-reduction abilities of the model.

ADM enhances the robustness and generalization ability of the whole network. The structure of ARM is shown in Fig. 5.

4) Multiscale Association Learning

**Training Process.** We designed three modules and obtained output images on three different scales. Therefore, the network takes four inputs, which are the raw image $X_r$, the object patch $X_o$, the object part-attention patches $X_p$, and the attention removal patch $X_d$. These images represent pest information at different scales. To comprehensively learn information on every scale and recognize pests, we construct an attention-guided coarse-to-fine CNN named MS-ALN.

The branches in MS-ALN take the above-mentioned four different scale images as input and correspondingly output four groups of feature maps $F_i$, $F_o$, $[F_p^1, F_p^2, \ldots, F_p^n]$ and $F_d$. The feature maps are further passed into full connection layers for prediction. MS-ALN...
finally outputs prediction probabilities \( P(X_r), P(X_o), P(X_p), \ldots, P(X_d) \) and \( P(X_d) \) at every scale. Taking the first scale as an example, the prediction process of the full connected layer is as follows:

\[
P(X_r) = f_c(F_r) \tag{9}
\]

where \( f_c(\cdot) \) stands for the function that maps feature maps to probabilities.

As in the case of most classification approaches, we choose cross-entropy loss as the loss function for all branches, which is formulated as follows:

\[
\begin{align*}
\text{Loss}_r &= \text{Loss}_{cls}(P(X_r), Y^*) \\
\text{Loss}_o &= \text{Loss}_{cls}(P(X_o), Y^*) \\
\text{Loss}_p &= \sum_{i=1}^{s} \text{Loss}_{cls}(P(X_p^i), Y^*) \\
\text{Loss}_d &= \text{Loss}_{cls}(P(X_d), Y^*)
\end{align*}
\tag{10}
\]

where \( \text{Loss}_{cls} \) is the entropy loss function and \( Y^* \) is the ground truth.

In this study, the four losses in Eq. 10 are summed up as the final loss. The \( \text{Loss}_p \) is multiplied by a weight \( \alpha \) to prevent it from dominating the optimization process. The weight \( \alpha \) varies with the preset window number \( s \). The overall loss calculation process is shown in Eq. 11.

\[
\text{Loss} = \text{Loss}_r + \alpha \times \text{Loss}_p + \text{Loss}_d \tag{11}
\]

**Testing Process.** Notably, ADM and ARM belong to two functional modules that provide different scale information for the same image. In training, the model improves the ability of fine-grained feature extraction by learning the attention regions obtained from ADM. It also improves the robustness to occlusions by learning the attention removal patch obtained from ARM. In testing, we unload these two modules to improve the speed without losing the test accuracy. We feed the raw image \( X_r \) into CNN backbone and TLM to filter out the background information and obtain the target patch \( X_o \). We then put \( X_o \) into the same CNN backbone again and obtain the object feature maps \( F_o \). Finally, we feed \( F_o \) into the classifier and obtain prediction classification.

**B. DECOUPLED LEARNING**

Directly training on the long-tailed dataset seriously affects the performance because the unevenly distributed data restrict feature extraction. Specifically, few data samples in tail classes easily lead to over-fitting and the numerous samples in the head classes tend to dominate the classifier training. The unbalanced distribution often leads to better performance on the head classes and poor performance on the tail classes. Because of the geographical distribution, growth environment, and quantitative characteristics of pests, pest data are often characterized by a long-tailed distribution. Fig. 6 shows the number distribution of different classes in dataset IP102 [6]. The bars in Fig. 6 represent the number of images in each class. The head classes on the left contain many images while the tail classes on the right contain few images. The entire distribution of the dataset is long-tailed.

In this study, we adopt decoupled learning [34] to alleviate the influence of uneven data distribution. The feature extractor and classifier are decoupled and trained with different sampling strategies. According to [38], this strategy promotes classifier training without damaging the ability to learn deep features caused by distorting original distributions.

**Sample Equalization.** Re-sampling and re-weighting are effective in dealing with long-tailed distribution problems, but they inevitably waste header data or over-fit tail data. To prevent the feature learning from being affected by distortions caused by the sampling strategy, instance-balanced sampling is used for feature learning and the parameters are fixed after 80 epochs. This sampling method allows each image to be sampled equally, regardless of the number of samples in a class. The probability \( p_i \) of sampling an image from class \( i \) is given by:

\[
p_i = \frac{n_i}{N} \tag{12}
\]

where \( n_i \) represents the image quantity of category \( i \) and \( N \) represents the number of training samples.

**Classifier Re-training.** In the classifier re-training process, we fix feature extraction network parameters, and update the classifier parameters using class-balanced sampling.
Thus, the probability $p_i$ of sampling from class $i$ is:

$$p_i = \frac{1}{C}$$

where $C$ represents the number of categories.

### IV. EXPERIMENTS

The proposed method was applied on the IP-102 dataset [6]. This section first introduces the public dataset and implementation details. Then the experimental results, which demonstrate the effectiveness of the proposed method, are presented.

#### A. DATASETS

IP102 [6] is one of the most extensive pest datasets, comprising of more than 75,000 pictures of 102 common agricultural pests. It is divided into 45,095 training images, 7,508 validation images, and 22,619 testing images. This dataset is derived from several professional agriculture and insect science websites and it contains common categories of pests. Compared with other pest datasets, IP102 better reflects the main problems of pest recognition tasks which are complex backgrounds, occlusions, and uneven data distribution.

#### B. IMPLEMENTATION DETAILS

The proposed MS-ALN used ResNet50 pre-trained on ImageNet [39] as the backbone for feature extraction and the last layer of Conv4 was selected as feature maps $F$. The size of input images was adjusted to $448 \times 448$. We first enlarged the images to $500 \times 500$ and then cropped them to $448 \times 448$. We performed random cropping during training and center cropping in the testing. The size of object patch $X_o$, and attention removal patch $X_d$ were all scaled to $448 \times 448$ pixels, and the object parts patches $X_p$ were adjusted to $224 \times 224$ pixels.

The proposed model only used image-level labels and did not require any other form of annotations. Loss $L_r$ in the first scale was used to optimize the network during the first 3 epochs. In subsequent training, the sum of loss in every branch was used to update the parameters in the network. The optimizer was SGD with a momentum of 0.9 and a weight decay of 0.0001. The batch size was set as 8 and the parameters were updated 5,636 times in one epoch on a V100 GPU. The initial learning rate was set to 0.001, with exponential decay of 0.5 after 15 epochs. Notably, the parameters of the feature extractor were frozen after 80 epochs and the classifier was trained with a constant learning rate of 0.0001 in the subsequent epochs.

#### C. EXPERIMENTAL RESULTS

1) Ablation Study

In this section, we report on the ablation study performed on the IP-102 dataset to demonstrate the effectiveness of the proposed TLM, ADM, ARM, and decoupled learning. The recognition results are illustrated in Table 1.

2) Performance Comparison

To verify the effectiveness of MS-ALN in pest classification, we compared it with typical pest classification algorithms proposed in recent years. The comparison results are shown in Table 2.

| Method                        | Acc  |
|-------------------------------|------|
| ResNet50                      | 66.08|
| ResNet50+TLM                  | 69.13|
| ResNet50+TLM+ADM              | 72.43|
| ResNet50+TLM+ADM+ARM          | 73.56|
| ResNet50+TLM+ADM+ARM+DL       | 74.61|

We use the implementation details in section IV-B in all experiments, and these details (e.g. image size, learning rate) are different from [6], which achieved accuracy of 49.4% using ResNet50 only. Under the details of our implementation, the recognition of the basic network ResNet50 was 66.08%. TLM is designed to filter the background information, improving the accuracy by 3.05%. ADM is designed to detect high response regions and guide the network to learn better fine-grained features. The performance was improved with an increase of 3.3% in accuracy. To further improve the overall generalization ability and robustness of the model, we added another recognition branch by ARM. After adding ARM, the test accuracy increased to 73.56%, with an improvement of 1.13%. When training the network with a decoupling strategy, the accuracy finally improved to 74.61%. This improvement proves that the problem caused by the data long-tailed distribution can be effectively alleviated by decoupled learning. The experimental results in Table 1 demonstrate that both the network and the training strategy used in this study contribute greatly to the recognition performance.

### TABLE 1. Ablation Study

| Method                        | Acc  |
|-------------------------------|------|
| ResNet50                      | 66.08|
| ResNet50+TLM                  | 69.13|
| ResNet50+TLM+ADM              | 72.43|
| ResNet50+TLM+ADM+ARM          | 73.56|
| ResNet50+TLM+ADM+ARM+DL       | 74.61|
CNN. Therefore, traditional machine learning methods are not considered.

We also performed comparative experiments to prove the effectiveness of ADM and ARM compared with random cropping and random erasing [37]. The results are shown in Table 3. Baseline is the method of combining ResNet50 with TLM. Although random cropping and random erasing are traditional data augmentation methods, their aimlessness limits further improvements of accuracy. ADM and ARM are more goal-oriented because they are built with model-based attention.

TABLE 3. Comparative experimental results of ADM and ARM

| Method              | Acc     |
|---------------------|---------|
| Baseline            | 69.13   |
| Random Cropping     | 71.98   |
| ADM                 | 72.43   |
| ADM+Random Erasing  | 73.00   |
| ADM+ARM             | 73.56   |

To verify the effectiveness of decoupled learning (DL) strategy in solving the problem of pest long-tailed distribution, we compared it with re-sampling (RS) and re-weighting (RW) strategies that are most widely used in the field of long-tailed data distribution. We designed a comparative experiment using class-balanced sampling [34] and class-balanced Softmax cross-entropy loss [29] as RS and RW strategies, respectively. To better contrast performances in classes with different numbers of images, we divided all classes into two groups and calculated the average F1-score. We followed Liu et al. [31] and combined the actual distribution of IP102 dataset and split the classes into two groups: Head class group (more than 100 images per class) and Tail class group (less than 100 images per class). As Figure 7, although all strategies improve the overall effect to some extent, both traditional methods (RW and RS) performed poorly in the head classes. These experimental results prove the effectiveness of decoupled learning in handling the long-tailed distribution of pest data.

8
VOLUME 4, 2016

FIGURE 7. Average F1-score in different groups using different long-tailed learning strategies.

3) Decoupled learning analysis

For a more detailed report on the impact of decoupled learning on classes with different numbers of images, we also plot the F1-score of per class in Fig. 6. Decoupled learning increases F1-score in most classes, especially in tail classes. It proves that decoupled learning is effective in alleviating the long-tailed distribution of pest data.
4) Visual Analysis

**Visual Comparison**: To compare the performance of the algorithm in practical application, we screened some samples that were misclassified by ResNet50 [13] while correctly classified by our method and visualized the activation maps in the network. As shown in Fig. 8, because the background in some of the images is very similar to pests, it is difficult for ResNet50 to focus on pests; ResNet50 is easily disturbed by the environment, which further leads to misclassification. Compared with ResNet50, our algorithm can locate the pests well even when the similarity between the pests and the background is very high.

![Visual Comparison](image)

**FIGURE 8.** Visual Comparison. The first row represents the original image. The images in the second and third rows represent the corresponding visualization of activation maps generated by ResNet50 [13] and our method, respectively.

**Visual Presentation**: To further verify the effectiveness of TLM, ADM, and ARM, we present the intermediate results of each module, as shown in Figs. 9-11.

Images from the top to the bottom of Fig. 9 are raw images, marked images, and object patches. The yellow rectangles in the second row represent target boxes obtained by the two groups of feature maps respectively, and the intersecting part is highlighted by red rectangles. If the target range is determined only by one box, it may cause the crop box to be wrong or too large. When the crop box is determined by two groups of adjacent feature maps, the intersection of the range covers the complete object and filters more backgrounds.

![TLM Visualization](image)

**FIGURE 9.** TLM Visualization

In Fig. 10, rectangles with red, purple, blue, and yellow colors are used to represent different ranges of high-response regions detected in ADM. ADM effectively detects the recognizable areas. By magnifying these areas, the small differences can be amplified and the fine-grained features can be further learned by the network.

![ADM Visualization](image)

**FIGURE 10.** ADM Visualization

The multiple attention regions output by ADM further guide the ARM to randomly remove the key region. As shown in Fig. 11, this module takes over the output of the ADM and randomly selects the attention region to delete, prompting the network to learn multiple attention regions. It is more effective in improving the stability and generalization of the network.

![ARM Visualization](image)

**FIGURE 11.** ARM Visualization

V. CONCLUSION

In the study, we proposed a pest recognition algorithm named MS-ALN. In view of the specific problems in pest recognition, we designed three modules named TLM, ADM, and ARM. Through experiments, we proved that the network we proposed can well cope with the problems of complex backgrounds and occlusion. The long-tailed distribution was effectively alleviated by introducing decoupled learning strategy into the training stage. On the large-scale pest data set IP102 [6], our model obtained state-of-the-art performance. By analyzing the experimental results, we found that the model easily misclassified small pests, and the identification accuracy of the tail data still needs to be improved. Therefore, we have two main future research directions: First, how to make the model capture the tiny features in the image of pests, to improve the identification ability of small pests. Second, how to further deal with the problems caused by the long-tailed distribution of pest data and further improve the overall performance of the model.
ACKNOWLEDGMENT
This work was supported in part by the Natural Science Foundation of Shandong Province under Grant ZR2020MF097, and in part by the National Natural Science Foundation of China under Grants 62106195, 61973185 and in part by the Development Plan of Young Innovation Team in Colleges and Universities of Shandong Province under Grant 2019KJN011.

REFERENCES

[1] X. Yang, Y. Luo, M. Li, Z. Yang, C. Sun, and W. Li, “Recognizing pests in field-based images by combining spatial and channel attention mechanism,” IEEE Access, vol. 9, pp. 162448–162458, 2021.
[2] R. Li, R. Wang, J. Zhang, C. Xie, L. Liu, F. Wang, H. Chen, T. Chen, H. Hu, X. Jia et al., “An effective data augmentation strategy for cnn-based pest localization and recognition in the field,” IEEE Access, vol. 7, pp. 160274–160283, 2019.
[3] Z. Luyu, H. Jin, T. Zhen, F. Sun, and H. Xu, “Small object recognition algorithm of grain pests based on ssd feature fusion,” IEEE Access, vol. 9, pp. 43202–43213, 2021.
[4] Y. Ai, C. Sun, J. Tie, and X. Cai, “Research on recognition model of crop diseases and insect pests based on deep learning in harsh environments,” IEEE Access, vol. 8, pp. 17166–17169, 2020.
[5] L. Nanni, A. Manfè, G. Maguolo, A. Luminì, and S. Brahnam, “High performing ensemble of convolutional neural networks for insect pest image detection,” Ecological Informatics, vol. 67, p. 101515, 2022.
[6] X. Wu, C. Zhan, Y.-K. Lai, M.-M. Cheng, and J. Yang, “Lp102: A large-scale benchmark dataset for insect pest recognition,” 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8779–8788, 2019.
[7] M. A. Ebrahimi, M. H. Khoshgahzaz, S. Minaei, and B. Jamshidi, “Vision-based pest detection based on svm classification method,” Comput. Electron. Agric., vol. 137, pp. 52–58, 2017.
[8] F. O. Faithpraise, P. Birch, R. C. D. Young, J. Obu, B. Faithpraise, and C. R. Chitum, “Automatic plant pest detection and recognition using k-means clustering algorithm and correspondence filters,” 2013.
[9] N. Larios, H. Deng, W. Zhang, M. Sarpola, J. Yuan, R. K. Paasch, A. Moldenke, D. A. Lytle, S. Ruiz-Correa, E. N. Mortensen, L. G. Shapiro, and T. G. Dietterich, “Automated insect identification through concatenated histograms of local appearance features: feature vector generation and region detection for deformable objects,” Machine Vision and Applications, vol. 19, pp. 105–113, 2007.
[10] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, “Visual categorization with bags of keypoints,” workshop on statistical learning in computer vision eccv, 2004.
[11] R. Samanta and I. Ghosh, “Tea insect pests classification based on artificial neural networks,” International Journal of Computer Engineering Science (IJCES), vol. 2, no. 6, pp. 1–13, 2012.
[12] A. A. Alfaris, Q. Chen, and M. Guo, “Deep learning based classification for pest diseases recognition,” Proceedings of 2018 International Conference on Mathematics and Artificial Intelligence, 2018.
[13] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.
[14] F. Ren, W. Liu, and G. Wu, “Feature reuse residual networks for insect pest recognition,” IEEE Access, vol. 7, pp. 122758–122768, 2019.
[15] E. Ayan, H. Ertöz, and F. Varçın, “Crop pest classification with a genetic algorithm-based weighted ensemble of deep convolutional neural networks,” Comput. Electron. Agric., vol. 179, p. 105809, 2020.
[16] H. T. Ung, H. Q. Ung, and B. T. Nguyen, “An efficient insect pest classification using multiple convolutional neural network based models,” arXiv preprint arXiv:2107.12189, 2021.
[17] Z. Liu, J. Gao, G. lai Yang, H. Zhang, and Y. He, “Localization and classification of paddy field pests using a saliency map and deep convolutional neural network,” Scientific Reports, vol. 6, 2016.
[18] L. Liu, R. Wang, C. Xie, P. Yang, F. Wang, S. Sudirman, and W. Liu, “Pestnet: An end-to-end deep learning approach for large-scale multi-class pest detection and classification,” IEEE Access, vol. 7, pp. 45301–45312, 2019.
[19] Y. Li, H. Wang, L. M. Dang, A. Sadeghi-Niaraki, and H. Moon, “Crop pest recognition in natural scenes using convolutional neural networks,” IEEE Access, vol. 169, p. 105174, 2020.
[20] M. T. Reza, N. Mehdizadeh, N. Tafazolli, and M. A. Alam, “Identification of crop consuming insect pest from visual imagery using transfer learning and data augmentation on deep neural network,” 2019 22nd International Conference on Computer and Information Technology (ICICT), pp. 1–6, 2019.
[21] L. Nanni, G. Maguolo, and F. Pancino, “Research on insect pest image detection and recognition based on bio-inspired methods,” Ecol. Informatics, vol. 57, p. 101089, 2020.
[22] P. Chen, W. Li, S. Yao, C. Ma, J. Zhang, B. Wang, C. Zheng, C. Xie, and D. Liang, “Recognition and counting of wheat mites in wheat fields by a three-step deep learning method,” Neurocomputing, vol. 437, pp. 21–30, 2021.
[23] Z. Wang, W. Gong, and W. Li, “A dynamic feature weighting method for mangrove pests image classification with heavy-tailed distributions,” Proceedings of the 2020 4th International Conference on Vision, Image and Signal Processing, 2020.
[24] G. Yang, G. Chen, C. Li, J. Fu, Y. Guo, and H. Liang, “Convolutional rebalancing network for the classification of large imbalanced rice pest and disease datasets in the field,” Frontiers in Plant Science, vol. 12, 2021.
[25] ——, “Convolutional rebalancing network for the classification of large imbalanced rice pest and disease datasets in the field,” Frontiers in Plant Science, vol. 12, p. 1150, 2021.
[26] M. Buda, A. Maki, and M. A. Mazurowski, “A systematic study of the class imbalance problem in convolutional neural networks,” Neural networks : the official journal of the International Neural Network Society, vol. 106, pp. 249–259, 2018.
[27] H. He and E. A. Garcia, “Learning from imbalanced data,” IEEE Transactions on Knowledge and Data Engineering, vol. 21, pp. 1263–1284, 2009.
[28] K. Cao, C. Wei, A. Gaidon, N. Areshigha, and T. Ma, “Learning imbalanced datasets with label-distribution-aware margin loss,” Advances in neural information processing systems, vol. 32, 2019.
[29] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. J. Belongie, “Class-balanced loss based on effective number of samples,” 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9260–9269, 2019.
[30] W. Ouyang, X. Wang, C. Zhang, and X. Yang, “Factors in finetuning deep model for object detection with long-tail distribution,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 864–873, 2016.
[31] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, and S. X. Yu, “Large-scale long-tailed recognition in an open world,” 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2532–2541, 2019.
[32] X. Yin, X. Yu, K. Sohn, X. Liu, and M. Chandraker, “Feature transfer learning for face recognition with under-represented data,” 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5697–5706, 2019.
[33] Q. Chen, Q. Liu, and E. Lin, “A knowledge-guide hierarchical learning method for long-tailed image classification,” Neurocomputing, vol. 459, pp. 408–418, 2021.
[34] B. Kang, S. Xie, M. Rohrbach, Z. Yan, A. Gordo, J. Feng, and Y. Kalantidis, “Decoupling representation and classifier for long-tailed recognition,” in Eighth International Conference on Learning Representations (ICLR), 2020.
[35] X. S. Wei, J.-H. Luo, J. Wu, and Z.-H. Zhou, “Selective convolutional descriptor aggregation for fine-grained image retrieval,” IEEE Transactions on Image Processing, vol. 26, no. 6, pp. 2868–2881, 2017.
[36] F. Zhang, M. Li, G. Zhai, and Y. Liu, “Multi-brand and multi-scale attention learning for fine-grained visual categorization,” in MMM, 2021.
[37] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, “Random erasing data augmentation,” in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 07, 2020, pp. 13001–13008.
[38] B. Zhou, Q. Cui, X.-S. Wei, and Z.-M. Chen, “Bbn: Bilateral-brand network with cumulative learning for long-tailed visual recognition,” 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9716–9725, 2020.
[39] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in 2009 IEEE conference on computer vision and pattern recognition, Ieee, 2009, pp. 248–255.
FUXIANG FENG is now pursing his B.E. degree of intelligent science and technology in School of Mathematics and Statistics, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China. His research interests include neural networks and image classification.

HANLIN DONG is now pursing his B.E. degree of intelligent science and technology in School of Mathematics and Statistics, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China. He is interested in neural networks and their practical applications.

YOUMEI ZHANG received the Ph.D. degree in pattern recognition and intelligent system from Shandong University, China, in 2019. She is currently working in school of Mathematics and Statistics, Qilu University of Technology (Shandong Academy of Sciences) as a lecturer, China. Her research interests include crowd counting, facial expression recognition and abnormal event detection.

YU ZHANG received the B.E. degree in control science and engineering from Shandong University, Jinan, China, in 2014, and Ph.D. degree in Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong, in 2019. She is currently a lecturer in Qilu University of Technology, Jinan, China. Her research interests include hyperspectral image segmentation and classification.

BIN LI received the bachelor’s, master’s, and Ph.D. degrees in control science, operational research and cybernetics, and pattern recognition and intelligent system from Shandong University, China, in 2002, 2005, and 2012, respectively.

***