Comparative Study of Ensemble Models of Deep Convolutional Neural Networks for Crop Pests Classification

zhongbin su
Northeast Agricultural University
jiaqi luo
Northeast Agricultural University
yue wang
Northeast Agricultural University
qingming kong (kkqqmmmm@126.com)
Northeast Agricultural University
baisheng dai
Northeast Agricultural University

Research Article

Keywords: Crop pest classification, Deep convolutional neural networks, Linear and Nonlinear, Ensemble model

Posted Date: December 10th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-1100814/v1

Comparative Study of Ensemble Models of Deep Convolutional Neural Networks for Crop Pests Classification

Zhongbin Su[a], Jiaqi Luo[a], Yue Wang[a], Qingming Kong[a*], Baisheng Dai[a*]

[a]Department of Electrical and Information engineering, Northeast Argicultural University

Articleinfo

Keywords:
Crop pest classification
Deep convolutional neural networks
Linear and Nonlinear
Ensemble model

ABSTRACT

Pest infestations on wheat, corn, soybean, and other crops can cause substantial losses to their yield. Early diagnosis and automatic classification of various insect pest categories are of considerable importance for accurate and intelligent pest control. However, given the wide variety of crop pests and the high degree of resemblance between certain pest species, the automatic classification of pests can be very challenging. To improve the classification accuracy on publicly available D0 dataset with 40 classes, this paper compares studies on the use of ensemble models for crop pests classification. First, six basic learning models as Xception, InceptionV3, Vgg16, Vgg19, Resnet50, MobileNetV2 are trained on D0 dataset. Then, three models with the best classification performance are selected. Finally, the ensemble models, i.e, linear ensemble named SAEnsemble and nonlinear ensemble SBPEnsemble, are designed to combine the basic learning models for crop pests classification. The accuracies of SAEnsemble and SBPEnsemble improved by 0.85% and 1.49% respectively compared to basic learning model with the highest accuracy. Comparison of the two proposed ensemble models show that they have different performance under different condition. In terms of performance metrics, SBPEnsemble giving accuracy of classification at 96.18%, is more competitive than SAEnsemble.

Introduction

According to the latest research, nearly half of global crop yields are affected by crop pests and diseases (Gandhi et al. 2018). Crop pests damage crops such as rice, wheat, and beans, and cause considerable crop production losses. Once pests develop in fields, only timely diagnosis by

farmers can enable effective treatment. Pest prevention methods depend on the species of pest infesting a field, but sorting pest species manually is cumbersome and inefficient because of the high similarity and complex structure among pests. Traditional pest classification methods include the use of the K-means clustering algorithm, which requires manual extraction of insect image features. This is time-consuming when the dataset is large (Faithpraise et al. 2013). As relevant information features are extracted from images manually, the learning system is not automated (Breitenreiter et al. 2015). In another study, a framework that classified leaf diseases of five diverse plants using image processing and machine learning algorithms achieved an accuracy of 83–94% (Al-Hiary et al. 2011). Traditional pest classification methods rely on manual operation and are generally used for small labels and datasets of small categories. In the face of multiple categories of pest classification, these methods appear increasingly inefficient and miscellaneous.

With the rapid development of deep convolutional networks in recent years, researchers have begun to use CNN to develop image classification systems for pest classification. A fine-tuning GoogLeNet model is used to classify ten common crop pests, and achieved an increase of 6.22% in accuracy compared to the most advanced method (Wang et al. 2020). Thenmozhi et al. (2019) built a CNN manually, compared this model with other advanced CNN pest classifiers, and analyzed the influence of various parameters on its performance. Nanni et al. (2020) used various methods to enhance the data to improve CNN performance and tested it on two pest datasets. The faster region-based CNN (Faster R-CNN) is trained to detect the location of lesions on leaves. Its classification accuracy of 7 types of tea tree insect pests was 89.4% (Lee et al. 2020). Chowdhury et al. (2020) used VGG16 and InceptionV3 to detect and identify rice pest. Concurrently, a two-stage training model derived from fine-tuning was proposed, which also achieved good results. Brahimi et al. (2017) used InceptionV3 and AlexNet models to detect tomato diseases, but they used the 1×1 filter for convolution, which filtered information along each layer and reduced the scale of the network. Fangyuan et al. (2020) proposed a cascade pest-classification method based on two stages framework. A context-aware attention network was constructed to classify the pest images, and then a pest classification model based on multi-projection was proposed. Arnal et al. (2018) conducted an important experiment in which models were trained using images obtained under experimental conditions as well as field conditions. The experiment highlighted a critical issue, no matter how accurate the model was, when applied to the actual production situation, the accuracy would be sharply reduced by nearly 50% and he proposed the suggestion of establishing a complete pest and disease bank. Wang et al. (2017) applied AlexNet and LeNet to a pest image dataset containing 82 classes and studied the effect of the convolutional layer on classification performance in CNN. In the case of large-scale multi-pest data, unbalanced classification substantially reduces the accuracy of classification. Recent studies on the classification of insect pests by CNN are summarized in Table 1.

Although the classification of insect pests has been studied for quite some time, the classification accuracy still needs to be improved, especially in the classification of multiple insect pests. Many algorithms in pest classification are based on the following models: Xception (Zhang et al. 2018), InceptionV3 (Guan et al. 2019), Vgg16 (Rangarajan et al. 2020), Vgg19 (Rajinikanth et al. 2020), MobileNetV2 (Meng et al. 2020), Resnet50 (Wen et al. 2020) and SqueezeNet. These models usually have different network structures and characteristics for learning different features and the generalization ability of model is also differs, which leads to the differing classification performance of different models for different class of crop pests. Many studies demonstrate that

the combination of multiple base learning models significantly improve accuracy than a single base learning model (Rokach et al. 2010). It is a good scheme to ensemble CNN models to create a system with high predictive capacity (Nanni et al. 2020). By ensembling models with different network structures and feature extraction characteristics, pest classification performance can be improved. Ayan et al. (2020) proposed an ensemble model based on a genetic algorithm to improve the classification accuracy of a basic model for multiple types of insect pests. However, the above work only considers the simple linear weighting of different CNN models, ignoring the complex nonlinear relations inside the outputs of basic models. Inspired by previous work, two kinds of ensemble models are proposed. The first creates a linear ensemble named SAEnsemble using simulated annealing algorithm, the second creates a nonlinear ensemble named SBPEnsemble using Back Propagation (BP) Neural Network. The influence of a foreground enhancement method on the performance of the CNN model for pest classification is also discussed.

The rest of this paper is organized as follows: Section 2 introduces the datasets and methods used in the current work. Section 3 presents the experimental results and advantages and disadvantages of the proposed ensemble models. Section 4 compare the two proposed models with other studies. Finally, Section 5 presents some conclusions and future prospects.

Table 1

Latest literatures of pest classification with CNN models

Literature	Model	Subject	Achievement
Wang et al. 2017	AlexNet, LeNet	82 classes pests	Apply CNN on multiple classifications
Brahimi et al. 2017	InceptionV3, AlexNet	Tomato diseases	Classify tomato diseases
Toenmozh et al.2019	Building CNN	Pests	Study the parameters of CNN
Lee et al. 2020	Faster R-CNN	Tea pests	Classify tea pests
Fangyuan et al. 2020	MDM	Pests	Develop multi-projection pest detection model
Yanfen et al. 2021	GoogLeNet	10 classes pests	Improve CNN accuracy
Ayan et al. 2021	InceptionV3, Xception ,MobileNetV2	40 classes pests	Combine three models

2. Materials and methods

2.1 Dataset

In this paper, the dataset is publicly available D0 dataset that consists of 40 kinds of crop pests. All 4508 RGB images having resolutions of 200×200 were proposed by Xie et al. (2018). The corresponding numbers and label of each class are listed in Table 2. D0 dataset was divided into three groups: training, verification, and testing, with 3151 images being used for training, 378 for verification, and 943 for testing. The D0 dataset contains class which are common such as the

Halyomorpha halys and Pieris rapae, class which have Obvious contours such as Sesamia inferens, class which have unique features such as Eurydema domulus, two classes are very similar and it is hard to distinguish such as Corythucha ciliat and Corythucha marmorata. Samples of 10 class are shown in Fig. 1.

Fig. 1. Samples of 10 defferent insects from D0 dataset

Table 2

Information of insect species, numerical lables of insect species in D0 dataset

Label	Insect Species	Total	Label	Insect Species	Total
0	Iscadia inexacta	79	**20**	Bemisia tabaci	147
1	Chilo supperssalis	93	**21**	Cletus punctiger	169
2	Porthesia taiwana Shiraki	141	**22**	Nezara viridula	175
3	Leptocorisa acuta	133	**23**	Callitettix ersicolor	156
4	Sesamia inferens	126	**24**	Dicladispa armigera	150
5	Cicadella viridis	138	**25**	Scotinophara lurida	117
6	Callitettix versicolor	156	**26**	Diostrombus politus Uhler	238
7	Plutella xylostella	68	**27**	Halyomorpha halys	101
8	Dolerus tritici Chu	88	**28**	Corythucha marmorata	98
9	Spilosoma obliqua	66	**29**	Pieris rapae	71
10	Stollia ventralis	72	**30**	Eurydema dominulus	150
11	Corythucha ciliata	90	**31**	Luperomorpha suturalis Chen	101
12	Lycorma delicatula	92	**32**	Nilaparvata lugens	61
13	Dolycoris baccarum	87	**33**	Ceroplastes ceriferus	100
14	Spodoptera litura	130	**34**	Chauliops fallax Scott	68
15	Dryocosmus KuriphilusYasumatsu	50	**35**	Maruca testulalis Gryer	73
16	Strongyllodes variegatus	135	**36**	Chromatomyia horticola	114
17	Ceutorhynchus asper Roelofs	146	**37**	Graphosoma rubrolineata	116
18	Laodelphax striatellus	61	**38**	Phyllotreta striolata	187
19	Riptortus pedestris	110	**39**	Aulacophora indica	78

2.2 Methods

To improve accuracy compared to basic learning model and to select suitable ensemble model under different situation. This paper contains the following two main processes. First, six basic learning models are pre-trained by using the transfer learning on the D0 dataset, three models having the best performance are selected. Then, two ensemble models are designed to investigate in crop pest classification and compare to three traditional ensemble models. Fig. 2 shows the framework of the comparative study.

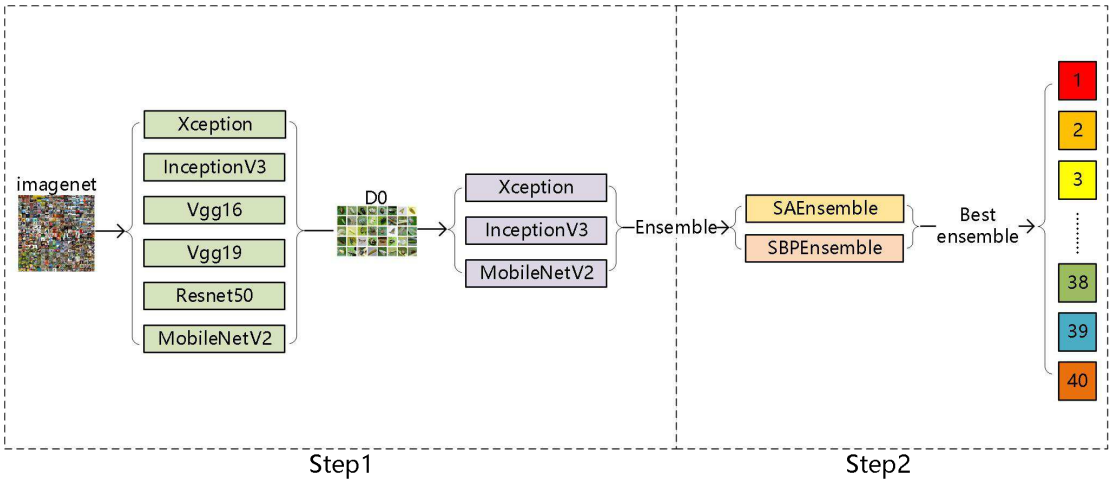


Fig. 2. General framework of the comparative study

2.2.1 Transfer learning

Transfer learning is an emerging family of machine learning techniques and has been actively studied in machine learning and AI communities in recent years (Pan et al. 2010). It is to solve new learning tasks using fewer examples by using information gained from solving related tasks (Mahmud et al. 2009). There are Two transfer learning methods. First, all the layers of the pre-training model are frozen, except the layers added by searchers. Second, the part of layers of the pre-training model are frozen, usually the multi-layer convolutional layer close to the input, for the large amount of low-level information retained. In this comparative study, Vgg16, Vgg19, and Resnet50 use the first method. InceptionV3, Xception, and MobileNetV2 use the second method.

2.2.2 Xception

Xception uses depth-wise separable convolution instead of traditional convolution operation. This helps maintain the classification accuracy while reducing the number of parameters and the amount of computation (Chao et al. 2021). This paper freeze the first 120 layers of Xception and add three fully connected layers containing 512 neurons. The activation function is ReLU (Hanin et al. 2019), it is used to avoid linearity, Softmax activation function is used in the final output layer to classify the pests. Fig. 3(a) shows the model structure.

2.2.3 InceptionV3

InceptionV3 is an implementation of GoogLeNet, its ability to deconstruct these features into smaller convolution sections (Zhao et al. 2020). Its network can be efficiently decomposed into

small convolution kernels, which greatly reduces the number of parameters of the model and the chance of overfitting (Liu et al. 2020). This paper freeze the first 270 layers of InceptionV3 and add three fully connected layers containing 512 neurons. The corresponding activation function is ReLU. Softmax activation functions are used in the final output layer to classify the pests. The specific structure is shown in Fig. 3(b).

2.2.4 MobileNetV2

MobileNetV2, the fast execution speed makes experimenting and parameter tuning much easier, while the low memory consumption is a desirable quality in the context of an ensemble of networks (Buiu et al. 2020). This paper freeze all layers of MobileNetV2 and add two fully connected layers containing 512 neurons. The model structure is given in Fig. 3(c).

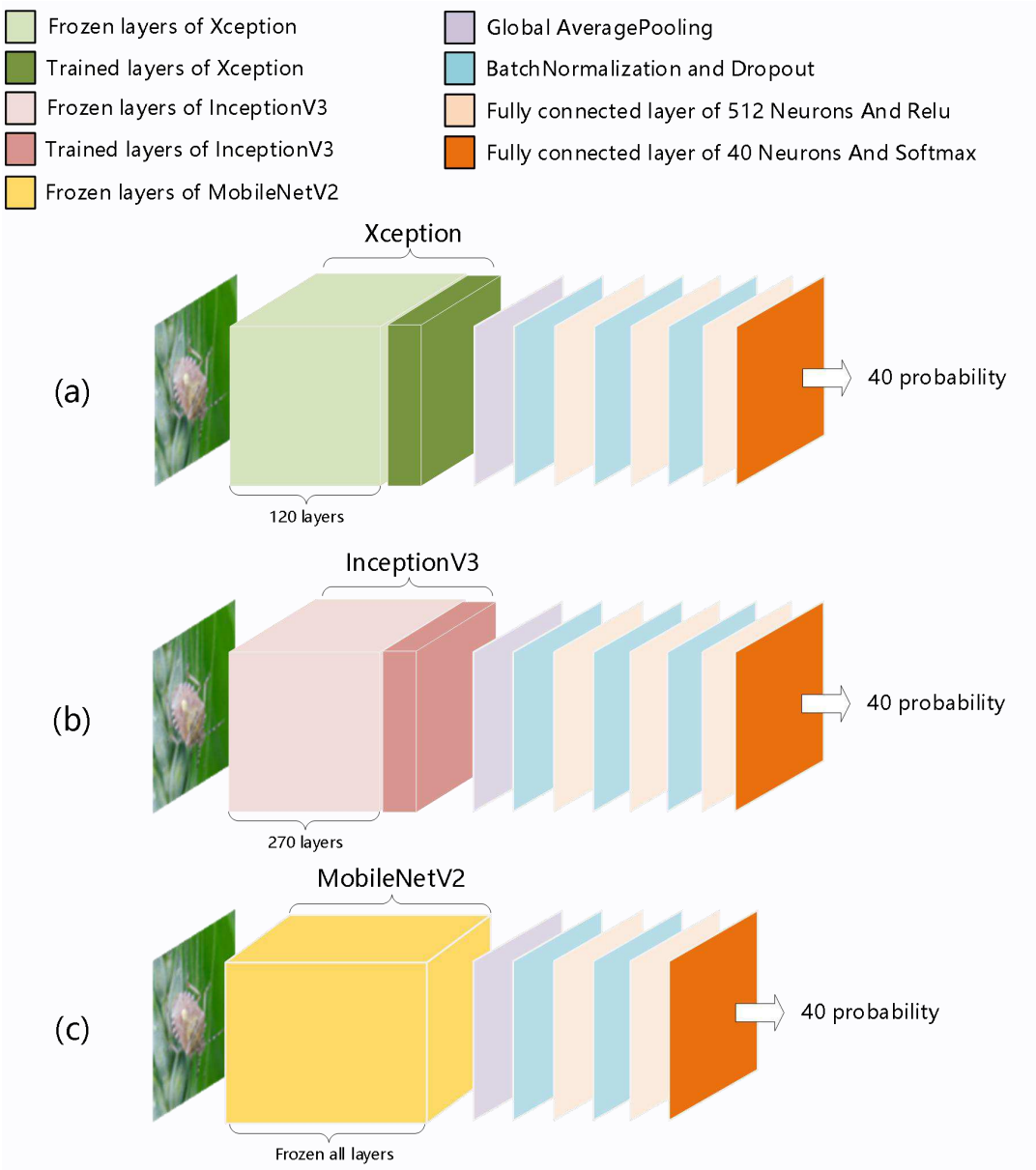


Fig. 3. Three architectures of CNN models after transfer learning (a) Xception, (b) InceptionV3, (c) MobileNetV2

2.3 Ensemble models

In this paper, ensemble models are based on voting method (Herrnson et al. 2019), it is divided into two kinds. One is hard voting that includes one ensemble model, in which each basic model votes on the class with the highest prediction probability, and each model has a vote. When the voting results of the three models are different, the model with the highest probability on the verification set is chosen to provide the result, it is shown in Fig. 4(a).

Second is soft voting that includes three ensemble models, the first model takes the average of all models' predicted probability in a certain class as the standard and the class with the highest probability as the result. it is shown in Fig. 4(b). It has disadvantages that the accuracy of the model is not considered, so the model with low accuracy may have a high impact on the result. To solve the problem, the second ensemble model of soft voting is proposed. This ensemble model takes the accuracy of the three models in the verification set as the weight, linearly multiplies the weights with the predicted results. In this way, models with high accuracy will be assigned higher weight, it is shown in Fig. 4(c). But the accuracy of model is not the best weight. To obtain the best weight, the linear ensemble model based on simulated annealing, SAEnsemble is proposed. It is the third model of soft voting.

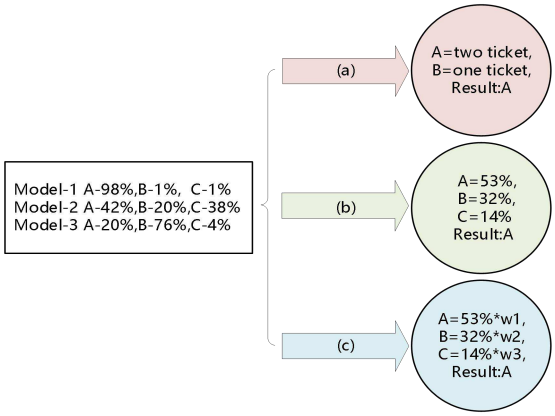


Fig. 4. Three traditional ensemble models

2.3.1 Linear ensemble

Simulated annealing (Xiang et al. 2000) is a random optimization algorithm based on the mountain climbing algorithm. Its starting point is based on the similarity between the annealing process of solid material in physics and the general combinatorial optimization problem. It will start from a relatively high initial temperature. With the continuous decline of temperature parameters, combined with the probability jump characteristics to randomly find the global optimal solution of the objective function in the solution space, the local optimal solution can jump out probabilistically and eventually approach the global optimal solution. This algorithm is usually used to optimize tasks.

In this paper, the initial solution space consists of three random solutions between 0 and 1. The corresponding initialization objective function value y_max is obtained. It is determined whether T is higher than Tmin. If it is large, it enters the loop. if it is small, it jumps out of the loop. After entering the loop, each new solution will float up and down the range of 0.05×0.025×T on the basis of the old one. After obtaining new solutions, it is checked whether they are all

between 0 and 1. If they are all consistent, the objective function value yNew is calculated. if yNew is greater than y, it is updated to the new solution space and then compared with the previous maximum objective function value y_max. if it is greater, the maximum objective function value and the corresponding solution space are updated. If yNew is smaller than the previous y, then the equation (1) determines whether the solution space is updated or not. This process is repeated until the optimal objective function value and the optimal solution space are attained. The steps described above are shown in Fig. 5, which is the processes of obtaining best weight by SAEnsemble.

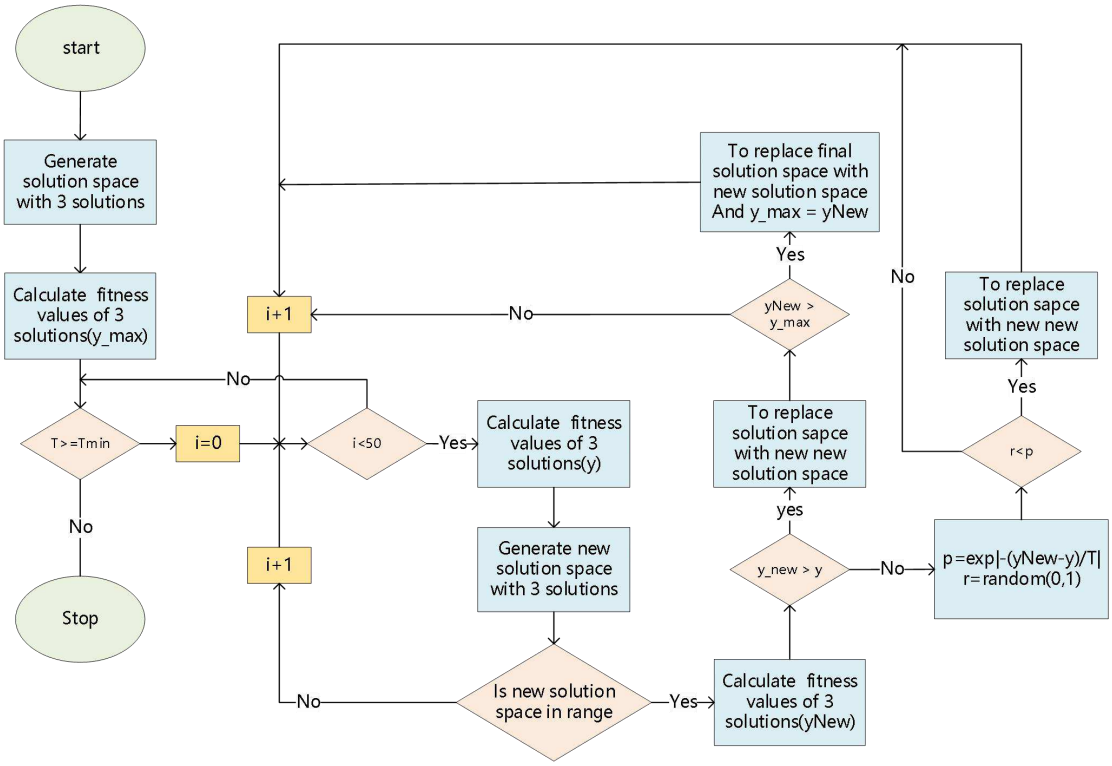


Fig. 5. Process of obtaining the best weights for the models through simulated annealing

$$p = \begin{cases} 1 & , E(X_{new}) < E(X_{old}) \\ \exp\left(-\frac{E(X_{new})-E(X_{old})}{T}\right) & , E(X_{new}) \geq E(X_{old}) \end{cases} \quad (1)$$

2.3.2 Nonlinear ensemble

Linear ensemble assign the weight to basic model directly, which will ignore the important nonlinear relations inside the outputs of basic model. Nonlinear ensemble takes the output of basic model as the input of another new network for training. In this comparative study, each basic model outputs the prediction probabilities of 40 classes at one time, so the three models output 120 prediction probabilities in total, which are taken as the inputs of the Back Propagation(BP) neural network, This network will pass the loss function to measure the probability distribution of neural network prediction and the distance between the real-worth probability distribution. The sample loss is reduced by increasing the output probability of the corresponding position with the

target value. It iterate repeatedly to obtain the optimal solution that the 40 best predictions for an image. This ensemble model is called SBPEnsemble.

The first layer of SBPEnsemble is an input layer of 512 neurons, with 120-size one-dimensional data input. It randomly throws away half of the data. The second layer contains 512 neurons, regularized by L2, and the activation function is ReLU. It also randomly throws away half of the data. The third layer, i.e., the output layer, contains 40 neurons and uses a linear activation function. Fig. 6 shows the specific structure.

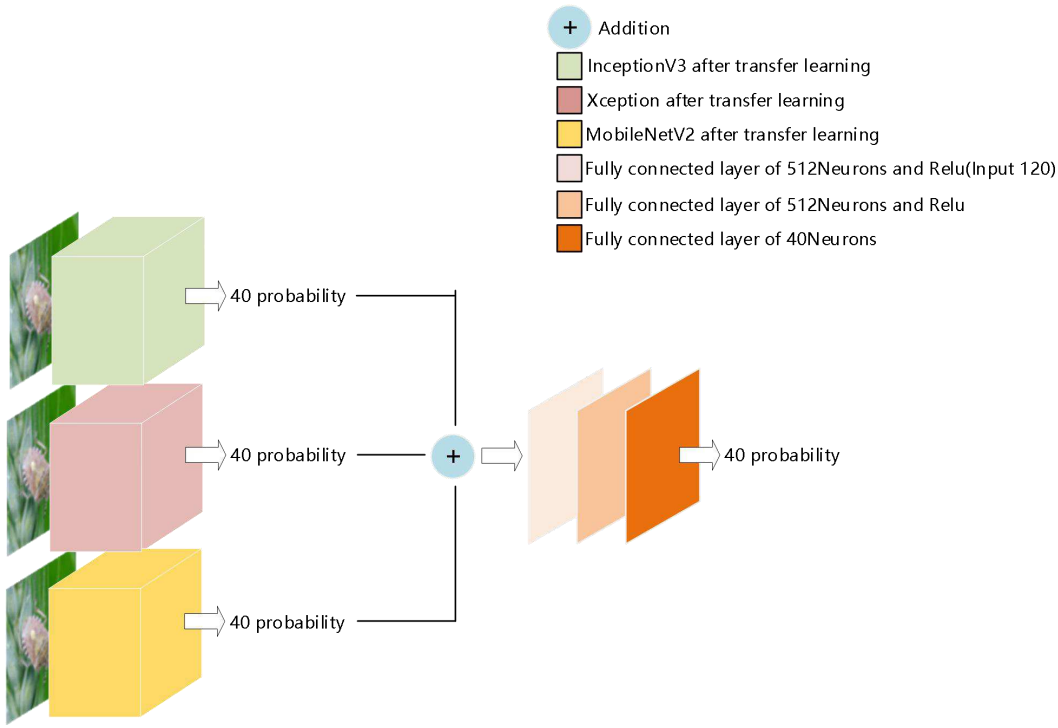


Fig. 6. Process of obtaining the best probability through BP neural network

3. Results

3.1 Experimental setup and training processes

In this comparative study, three basic learning models were used for transfer learning, two ensemble models were used for training. Each model was trained through 20 epochs. Before the image is input into the model, it is also subjected to data enhancement using image processing methods such as rotation, scaling, and mirroring. The experimental environment is 3080TI of the Ubuntu operating system, using Python, Keras, and OpenCV learning framework. InceptionV3 has the most parameters and it takes the longest time to train. The training time of a basic model is listed in Table 3. The loss curve of InceptionV3 and Xception fluctuates greatly and MobileNetV2 is smooth in validation, loss curve is shown in Fig. 7. The accuracy of three models tend to be stable within 20 cycles in validation, accuracy curve is shown in Fig. 8. In order to test the performance of each model, accuracy, precision, and Recall are used as evaluation indices. It is shown in Table 4.

Table 3

Time required for training models and models parameters

Model	Time	Parameters(millions)
SAEnsemble	5.54 s	-
SBPEnsemble	10.2 s	-
Xception	395 s	21.93
InceptionV3	1177 s	23,14
MobileNetV2	1086 s	3.47

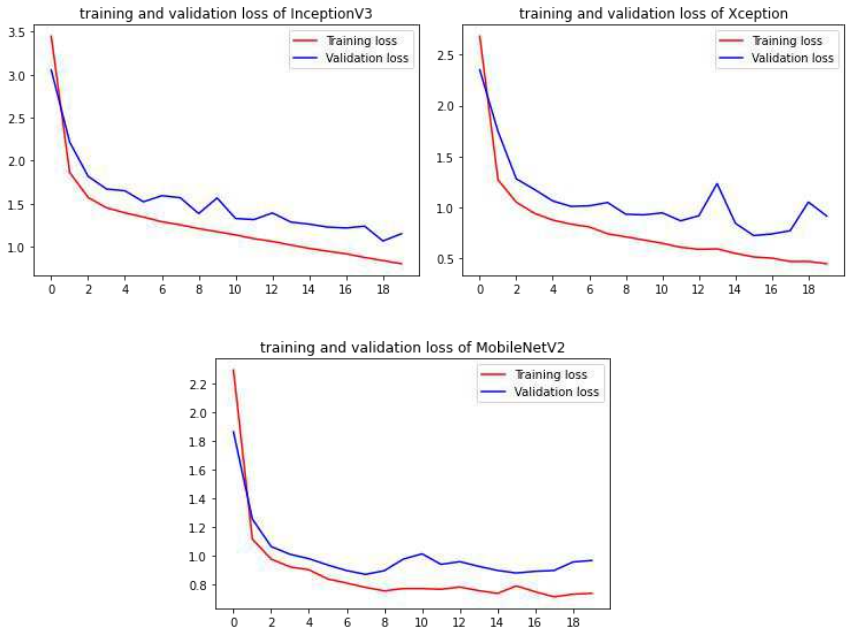


Fig. 7. Loss curve of three basic models

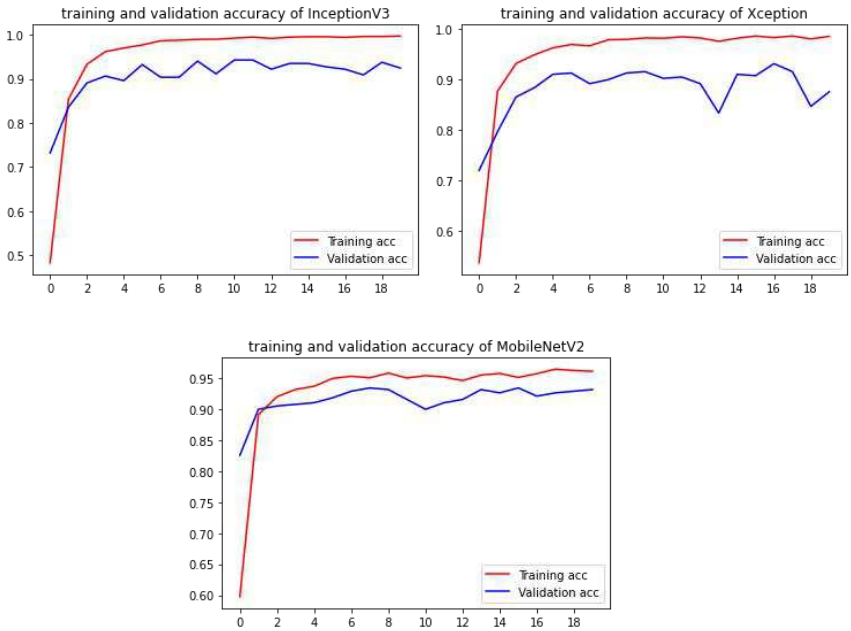


Fig. 8. Accuracy curve of three basic models

Table 4

Experimental classification results (%) on the test set

Model	Accuracy	Precision	Recall	Improvement on MobileNetV2
InceptionV3	94.69	94.89	93.48	3.71
Xception	91.94	92.60	90.97	0.96
MobileNetV2	90.98	95.62	94.64	-
SAEnsemble	95.54	96.03	95.74	4.56
SBPEnsemble	96.18	96.45	95.37	5.20

3.2 Performance of the CNN models in pest classification

Look at classes with the highest accuracy of each basic learning model, it is found that Xception can outperform the other models on some beetles classes, such as class 12,17, 23 ,24, 25, see second line of Fig. 9. To input one image into the three basic learning models, the Fusion Feature Map extracted from the last convolution layer of Xception is precisely concentrated in the target region, while MobileNetV2 is also concentrate in the target region but the region is more divergent and the learning performance is not as good as Xception, see Fig. 10. However, MobileNetV2 show better performance on some moths classes, such as class 1,9,20, 26,35, which is shown in third line of Fig. 9. To input one image into the three basic learning models, only MobileNetV2 learned the correct area, see Fig. 11. InceptionV3 has the highest accuracy on test sets. Different basic learning model has different classification performance for pest class, this is one of the reasons to combine the three basic learning models. The process of Fusion Feature Map is shown in Fig. 12.

Fig. 9. The five classes with the highest accuracy of three basic learning models

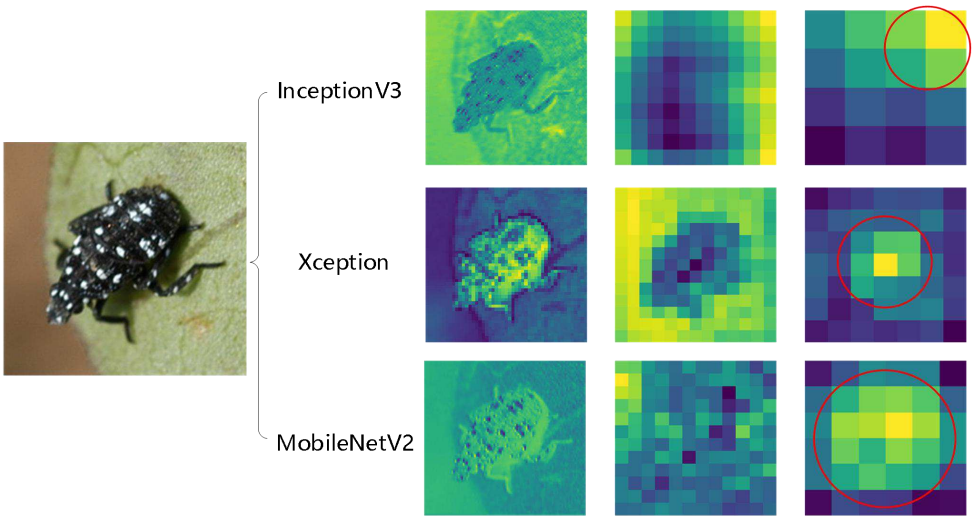


Fig. 10. The Fusion Feature Map of first, middle and last convolution in three basic learning model on class 12

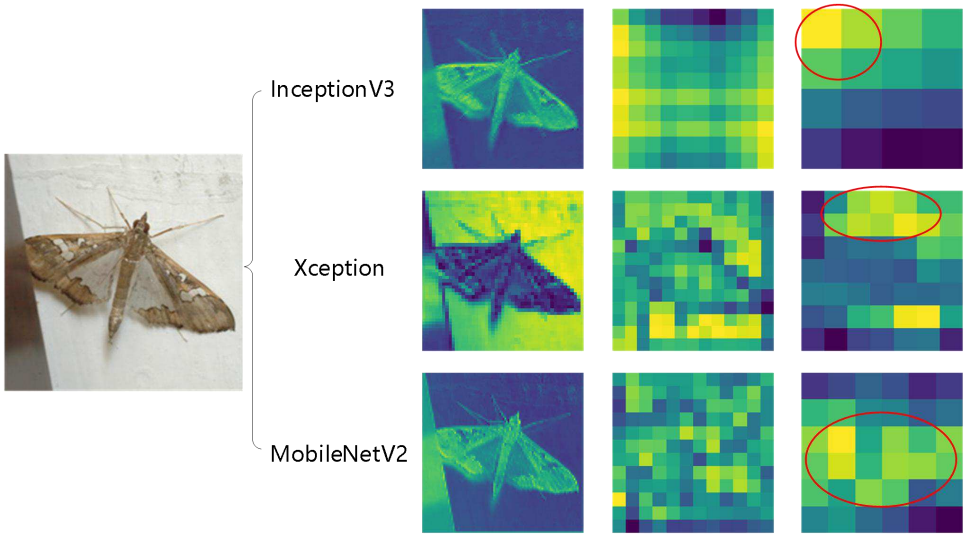


Fig. 11. The Fusion Feature Map of first, middle and last convolution in three basic learning model on class 15

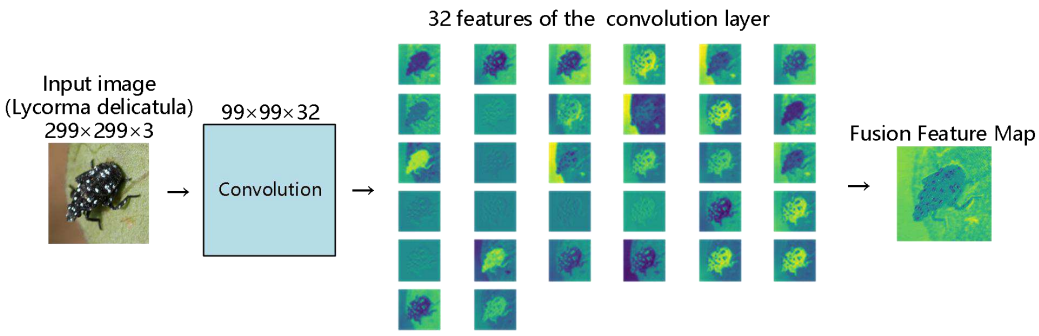


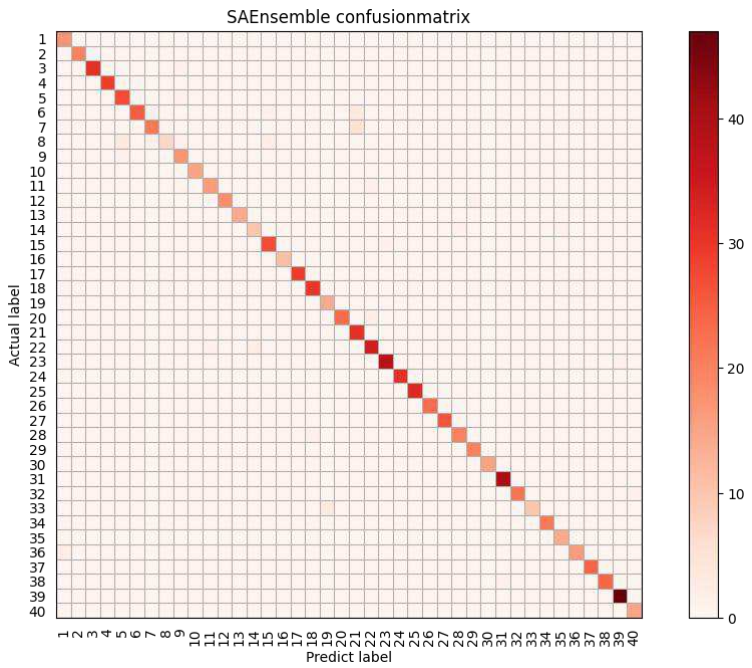
Fig. 12. The process of Fusion Feature Map of first convolution for InceptionV3

3.3 Comparison of SAEnsemble and SBPEnsemble

Fig. 13 shows the confusion matrix of prediction results of two proposed models, Table 5 presents the accuracy of SBPEnsemble on the test set. The model can fully classify 23 insect pests and average classification rate of 96.55%. Table 6 presents the accuracy of SAEnsemble on the test set. The model could fully identify 22 insect pests and average classification rate of 96.03%. The distribution of the two matrices is roughly the same because they are combined with the same three basic learning model, but there are some difference between the two ensemble models in performance.

SAEnsemble can comprehensively consider the features extracted from each model to get the weight and then carry out linear ensemble, when the insect pest and the background were very similar, the features extracted by a basic learning model were not obvious, leading to classification errors in SAEnsemble, such as Fig. 14(1)(2). When the characteristics of the two insect pests are very similar, the features extracted by three basic learning model are also similar. Which lead to classification errors in SAEnsemble. The feature images extracted from two classes by three basic learning models are shown in Fig. 15.

The nonlinear ensemble model SBPEnsemble is a BP neural notwork, which learns all the outputs of the three models. From a mathematical point of view, it avoids the issue that SAEnsemble faces, the extraction of unobvious features and similar features to some extent. But, it is slightly weaker than that of SAEnsemble under the condition of obvious pest characteristics, simple contour, and huge difference between pest and background. In such a case, SAEnsemble could accurately extract the characteristics of insect pests and classify them correctly, while SBPEnsemble may made wrong predictions, see Fig. 16. Therefore, in a real-world production situation, when the pest morphology is complex and the pest have a high degree of similarity with background, the SBPEnsemble can be used to combine the model. When the pest morphology is relatively simple and distinguishable from the background, SAEnsemble may achieve better results. In summary, SBPEnsemble has better performance, Fig. 17 shows Accuracy Line of SAEnsemble and SBPEnsemble.



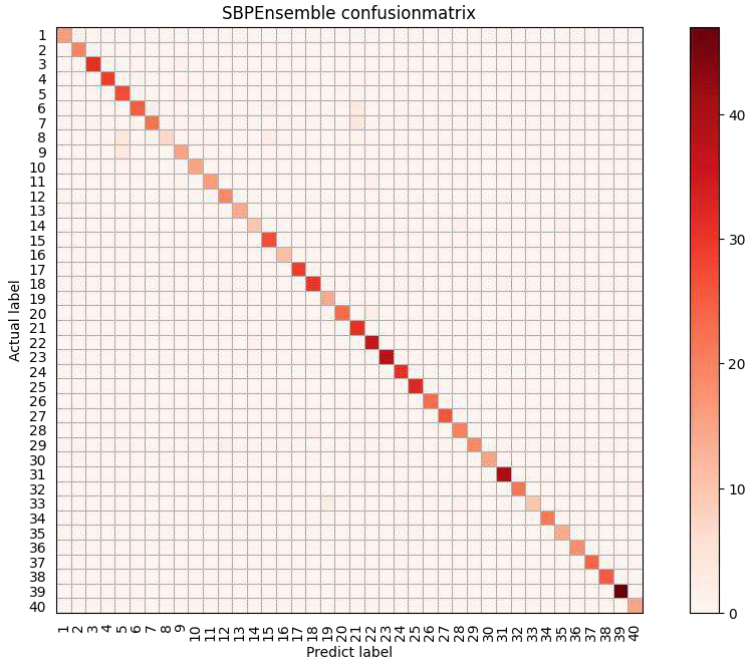


Fig. 13. Accuracy confusion matrices of SAEnsemble and SBPEnsemble

Table 5

Individual class accuracy of SBPEnsemble(words in bold are the five least accurate)

Label	Accuracy	Nums	Label	Accuracy	Nums
0	1.00	17	**20**	1.00	31
1	1.00	20	**21**	0.95	37
2	0.97	32	**22**	0.97	39
3	1.00	29	**23**	1.00	31
4	0.96	28	**24**	1.00	32
5	0.83	30	**25**	1.00	23
6	0.88	26	**26**	1.00	26
7	0.50	14	**27**	0.95	21
8	0.83	18	**28**	0.95	20
9	1.00	15	**29**	1.00	15
10	0.94	17	**30**	1.00	40
11	1.00	19	**31**	0.96	23
12	1.00	14	**32**	0.77	13
13	0.83	12	**33**	1.00	21
14	0.96	28	**34**	1.00	14
15	1.00	11	**35**	1.00	18
16	1.00	29	**36**	1.00	24
17	1.00	30	**37**	0.96	25
18	1.00	14	**38**	1.00	47
19	0.92	25	**39**	1.00	15

Table 6

Individual class accuracy of SAEnsemble(words in bold are the five least accurate)

Label	Accuracy	Nums	Label	Accuracy	Nums
0	1.00	17	**20**	1.00	31
1	1.00	20	**21**	0.92	37
2	0.97	32	**22**	0.97	39
3	1.00	29	**23**	1.00	31
4	0.96	28	**24**	1.00	32
5	0.83	30	**25**	1.00	23
6	0.81	26	**26**	1.00	26
7	0.50	14	**27**	0.90	21
8	0.89	18	**28**	1.00	20
9	1.00	15	**29**	1.00	15
10	0.94	17	**30**	1.00	40
11	0.95	19	**31**	0.96	23
12	1.00	14	**32**	0.77	13
13	0.75	12	**33**	1.00	21
14	0.96	28	**34**	1.00	14
15	1.00	11	**35**	0.89	18
16	1.00	29	**36**	1.00	24
17	1.00	30	**37**	0.96	25
18	1.00	14	**38**	1.00	47
19	0.92	25	**39**	1.00	15

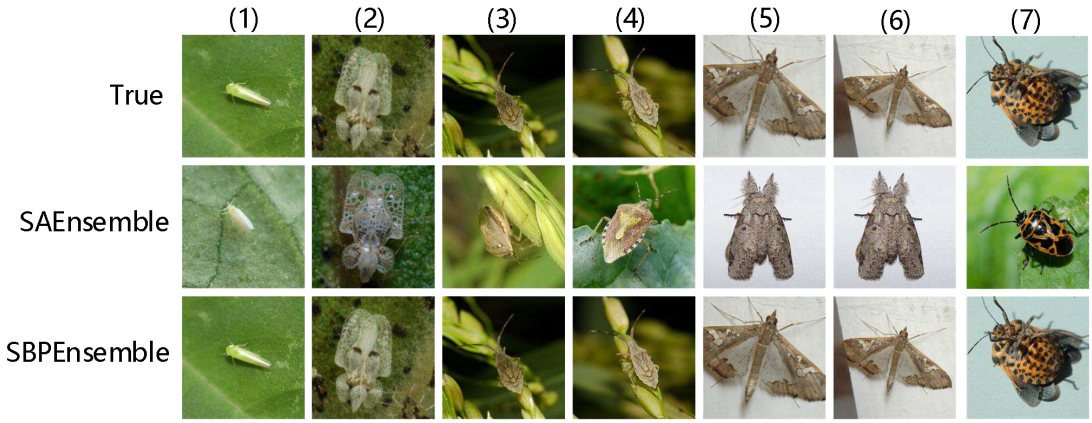


Fig. 14. Misclassified images of SAEnsemble

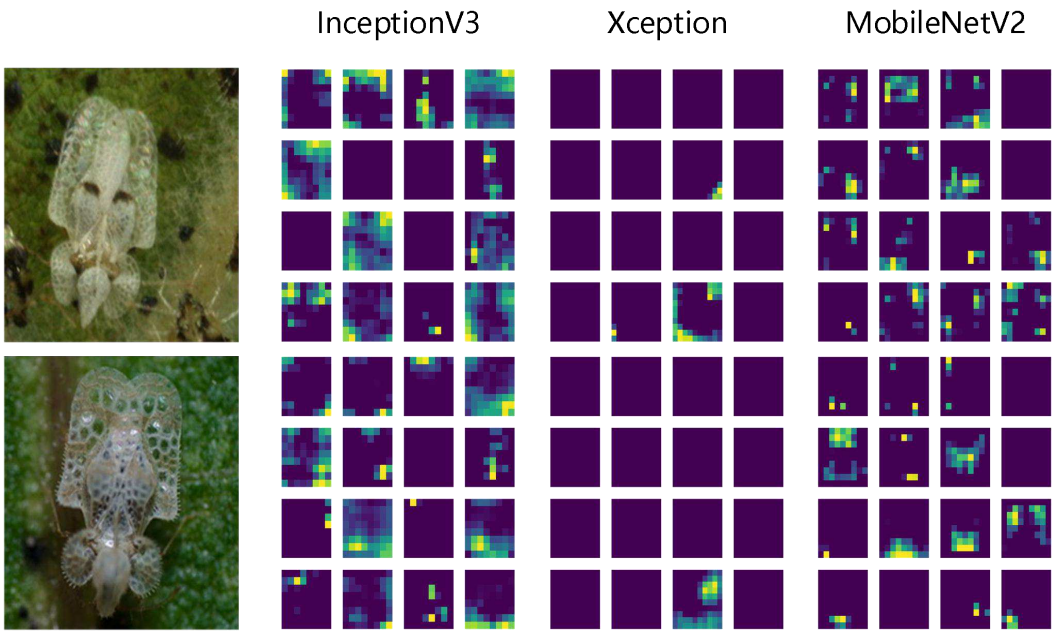


Fig. 15. First 16 features of last layers for three models of two images

Fig. 16. Misclassified images of SBPEnsemble

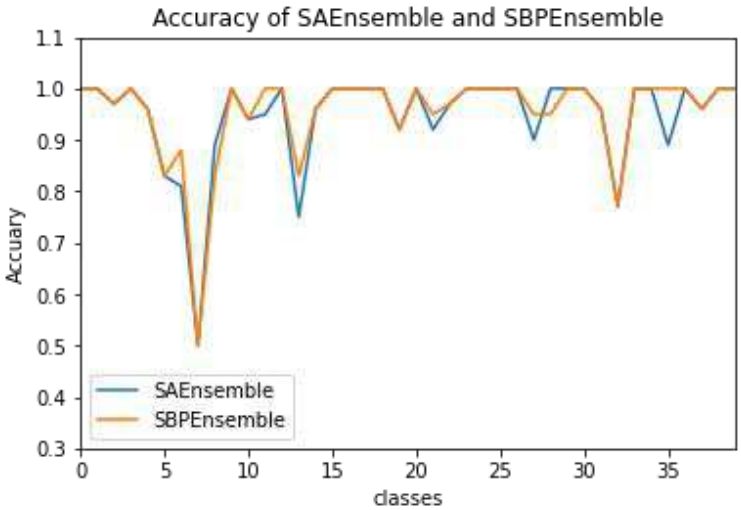


Fig. 17. Accuracy Line chart of SAEnsemble and SBPEnsemble

4. Discussion

This section contains result illustration of two proposed models and compare the two proposed models with other studies on D0 dataset. It also contains comparison of model training with/without foreground-based enhancement.

4.1 Result illustration of two ensemble models

When checking the prediction results of the five models, including three basic models and two proposed ensemble models. It is found that when the predicted values of the three basic models were different, SAEnsemble and SBPEnsemble can obtain the same (correct) results. The results obtained by ensemble models is not only based on the basic model with the highest accuracy, InceptionV3, but also considered two other models with lower accuracy, Xception and MobileNetV2, as shown in the second row in Fig. 18. It is proved that although the ensemble model is likely to rely on the model with the highest accuracy, it also learn the other two models with lower accuracy, so as to obtain better results. This indicates that the ensemble model is effective.

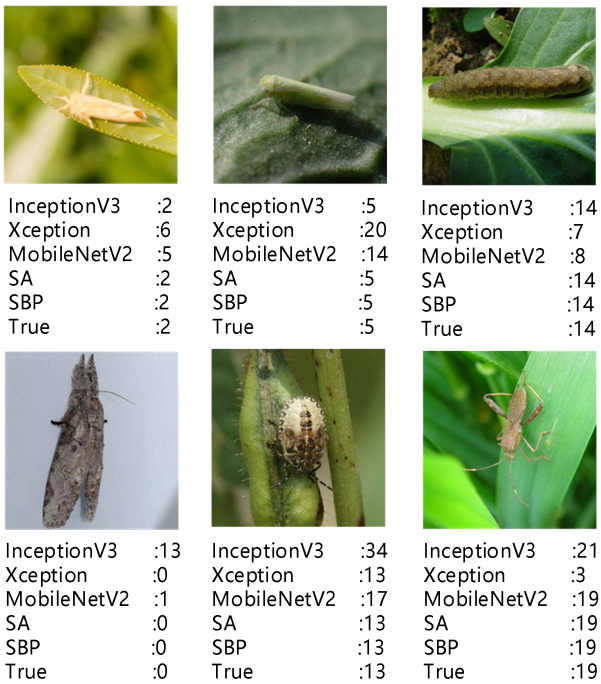


Fig. 18. Some classification results of five models(InceptionV3, Xception, MobileNetV2, SAEnsemble, SBPEnsemble)

4.2 Comparison of the proposed model with existing methods

On D0 dataset, Thenomzhi and Reddy (2019) proposed a CNN model that have twelve layers, this model achieved a classification 95.97%. Xie et al. (2018) created an automated system for crop pest classification and achieved a classification 89.3%. SBPEnsemble with accuracy of

96.18% outperforms the other studies. The accuracy of proposed models and other methods on D0 is presented in Table 7.

Ayan et al. (2020) proposed a linear ensemble model for crop pest classification in D0, named GAEnsemble. InceptionV3, Xception, MobileNet serve as its basic learning models, the accuracy reached 97.06%, 97.93%, and 97.39% respectively. They combine the models with a genetic algorithm (GAEnsemble) to achieve 98.81% accuracy, it increased by 0.82% compared to the most accurate basic model (Xception). This paper user InceptionV3, Xception, MobileNetV2, the accuracy are 94.69%, 91.94%, 90.98% respectively. Two proposed ensemble models, SAEnsemble and SBPEnsemble, increased by 0.85% and 1.49% compared to most accurate basic model (InceptionV3). Considering that the higher the accuracy of basic model, the lower the improvement of ensemble model. When the accuracy of a basic model is low, the ensemble model will perform better. The training and adjustment of the basic model in this paper still need further study in future.

Table 7

The accuracy of the proposed model and other method for D0 dataset

Work	Accuracy(%)	Ensemble(yes/no)
(Xie et al. 2018)	89.30	no
(Themozhi and Reddy 2019)	95.97	no
SAEnsemble	95.54	yes
SBPEnsemble	96.18	yes

4.3 Comparison of model training with/without foreground-based enhancement

In the experiment (Thenmonzhi et al. 2019), image processing is used to extract the main body of insect pests, which resulted in a rise in the accuracy. In this comparative study, we try to process the image based on foreground enhancement (Deepak et al. 2016). First, the image is converted to a grayscale image (Gunes et al. 2016). Second, the image is extracted by using the Sobel operator (Chunlan et al. 2009) for image gradient calculation (Ye et al. 2014). Third, a Gaussian filter (Kotecha et al 2003) is used to eliminate the Gaussian noise (Majee et al. 2005) in the image. The insect contour is outlined by the (Hai-xia et al. 2009) of image expansion and then corrosion. Finally, the coordinates of the four edges of the contour are obtained, and images are captured with the insect profile and features intact. This process is illustrated in Fig. 19. It is found that images with simple backgrounds can be extracted satisfactorily, while those with complex backgrounds tend to be unsatisfactorily extracted. As shown in Fig. 20. The three basic models were trained on dataset D0(D0_original) and dataset D0(D0_processed) after foreground-based enhancement. The result is showed in Table 8. The accuracy of one model was improved on D0_processed, while the accuracy of other models were decreased. Therefore, in the actual production situation, it is recommended to carry out foreground-based enhancement for model training if the images have simple background.

Table 8

Accuracy (%) of three basic learning models on different datasets

Image dataset	InceptionV3	Xception	MobileNetV2
D0_original	94.68	90.08	89.92
D0_processed	91.68	91.62	85.69

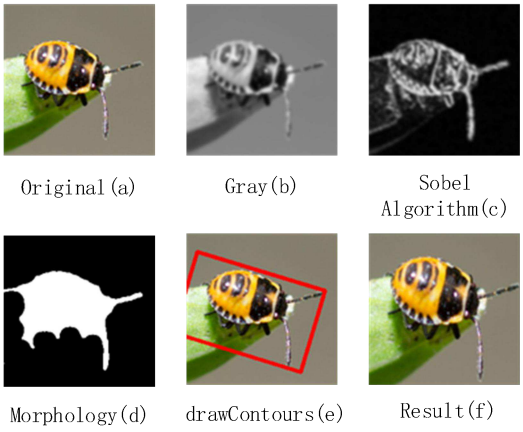


Fig. 19. Input image and main process of foreground-based enhancement

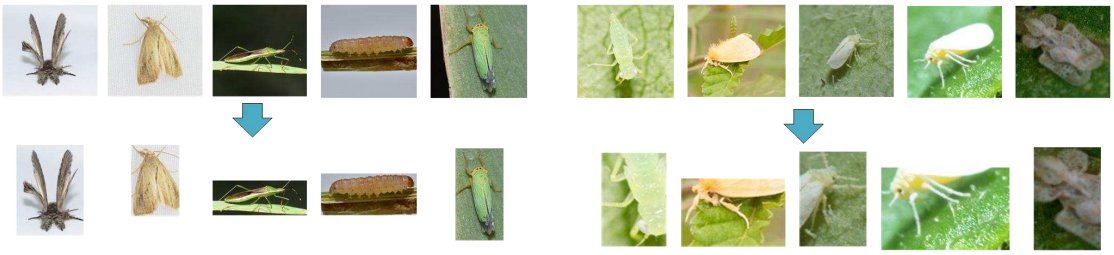


Fig. 20. different results of foreground-based enhancement

5. Conclusion

This paper proposed and compared two ensemble models, SAEnsemble and SBPEnsemble, for corp pest classification. The experimental results on D0 dataset of 4508 crop pest images show that SAEnsemble and SBPEnsemble achieved high accuracy rates of 95.54% and 96.18% respectively, which are 0.85% and 1.49% higher than the basic learning model. In addition, different basic learning model has different classification performance in pest class, Xception is suitable for pests of the beetle class and MobileNetV2 is suitable for pests of moth family class. Different ensemble models can be selected as per different actual agricultural production conditions, the linear ensemble is suitable for case when pest profile is obvious and distinguishable from the background, the nonlinear ensemble is suitable for case when pest profile is complex and similar to the background. But, The performance of the nonlinear ensemble is better than linear ensemble. Additionally, image processing with foreground-based enhancement is suitable for basic model training on datasets with the images having simple background. In future, this can serve as a guide to aid decision-making by farmers and help in preventing pest transmission.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (No. 31902210), the Heilongjiang Province “Hundred Million” Engineering Science and Technology Major Special Project (No. 2019ZX14A04), the Central Government to Support the Reform and Development Fund of Heilongjiang Local Universities (No. 2020GSP15).

References

Gandhi, R., Nimbalkar, S., Yelamanchili, N., Ponkshe, S. (2018). Plant disease detection using CNNs and GANs as an augmentative approach. In: 2018 IEEE International Conference on Innovative Research and Development (ICIRD). IEEE, pp. 1–5.

Faithpraise, F., Birch, P., Young, R., Obu, J., Faithpraise, B., Chatwin, C., 2013. Automatic plant pest detection and recognition using k-means clustering algorithm and corre-spondence filters. Int. J. Adv. Biotechnol. Res. 4, 189–199.

Breitenreiter, A., H. Poppinga, T. U. Berlin, and F. N. Technik. 2015. Deep learning. Nature 521:2015.

Al-Hiary, H., Bani-Ahmad, S., Reyalat, M., Braik, M., ALRahamneh, Z., 2011. Fast and accurate detection and classification of plant diseases. Int. J. Comput. Appl. 17,31–38.

Nanni, L., et al. (2020). "Insect pest image detection and recognition based on bio-inspired methods." Ecological Informatics 57.

Ayan, E., et al. (2020). "Crop pest classification with a genetic algorithm-based weighted ensemble of deep convolutional neural networks." Computers and Electronics in Agriculture 179.

Balagourouchetty, L., et al. (2020). "GoogLeNet-Based Ensemble FCNet Classifier for Focal Liver Lesion Diagnosis." Ieee Journal of Biomedical and Health Informatics 24(6): 1686-1694.

Buiu, C., et al. (2020). "MobileNetV2 Ensemble for Cervical Precancerous Lesions Classification." Processes 8(5).

Chao, X., et al. (2021). "Construction of Apple Leaf Diseases Identification Networks Based on Xception Fused by SE Module." Applied Sciences-Basel 11(10).

Chunlan, Y., et al. (2009). "Study of Infrared Image Edge Detection Based on Sobel Operator." Laser and Infrared 39(1): 85-87.

Deepak, K. T. and S. R. M. Prasanna (2016). "Foreground Speech Segmentation and Enhancement Using Glottal Closure Instants and Mel Cepstral Coefficients." Ieee-Acm Transactions on Audio Speech and Language Processing 24(7): 1204-1218.

Guan, Q., et al. (2019). "Deep convolutional neural network Inception-v3 model for differential diagnosing of lymph node in cytological images: a pilot study." Annals of Translational Medicine 7(14).

Gunes, A., et al. (2016). "Optimizing the color-to-grayscale conversion for image classification." Signal Image and Video Processing 10(5): 853-860.

Hai-xia, Y. A. N., et al. (2009). "Edge detection method based on Percentile Morphological transform." Microcomputer Information 25(3): 288-289.

Hanin, B. (2019). "Universal Function Approximation by Deep Neural Nets with Bounded Width and ReLU Activations." Mathematics 7(10).

Herrnson, P. S., et al. (2019). "Mobilization Around New Convenience Voting Methods: A Field Experiment to Encourage Voting by Mail with a Downloadable Ballot and Early Voting." Political Behavior 41(4): 871-895.

Kotecha, J. H. and P. M. Djuric (2003). "Gaussian sum particle filtering." Ieee Transactions on Signal Processing 51(10): 2602-2612.

Lee, S.-H., et al. (2020). "Identification of tea foliar diseases and pest damage under practical field conditions using a convolutional neural network." Plant Pathology 69(9): 1731-1739.

Liu, K., et al. (2020). "An Improved InceptionV3 Network for Obscured Ship Classification in Remote Sensing Images." Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing 13: 4738-4747.

Liu, X., et al. (2018). "Ensemble Transfer Learning Algorithm." Ieee Access 6: 2389-2396.

Majee, P., et al. (2005). "Colored non-Gaussian noise induced resonant activation." Chemical Physics Letters 416(4-6): 256-260.

Meng, L., et al. (2020). "Semantic Segmentation Algorithm Based on Improved MobileNetV2." Acta Electronica Sinica 48(9): 1769-1776.

Rajinikanth, V., et al. (2020). "A Customized VGG19 Network with Concatenation of Deep and Handcrafted Features for Brain Tumor Detection." Applied Sciences-Basel 10(10).

Rangarajan, A. K. and R. Purushothaman (2020). "Disease Classification in Eggplant Using Pre-trained VGG16 and MSVM." Scientific Reports 10(1).

Shen, Z. and Y. Tang (2015). "A modified ensemble Kalman particle filter for non-Gaussian systems with nonlinear measurement functions." Journal of Advances in Modeling Earth Systems 7(1): 50-66.

Thenmozhi, K. and U. S. Reddy (2019). "Crop pest classification based on deep convolutional neural network and transfer learning." Computers and Electronics in Agriculture 164.

Wang, F., et al. (2020). "Fusing multi-scale context-aware information representation for automatic in-field pest detection and recognition." Computers and Electronics in Agriculture 169.

Wang, Z., et al. (2005). "An Optimized Neural Network Linear Ensemble for Classification." Journal of Software 16(11): 1902-1908.

Wen, L., et al. (2020). "A transfer convolutional neural network for fault diagnosis based on ResNet-50." Neural Computing & Applications 32(10): 6111-6124.

Xiang and Gong (2000). "Efficiency of generalized simulated annealing." Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics 62(3 Pt B): 4473-4476.

Ye, X., et al. (2014). "Gradient Analysis for Image Local Reconstruction." Computer Engineering 40(12): 235-241.

Zhang, Q., et al. (2018). "Fine-grained image classification based on Xception." Journal of Chongqing University. Natural Science Edition 41(5): 85-91.

Zhao, Y., et al. (2020). "Intelligent Recognition of Fatigue and Sleepiness Based on InceptionV3-LSTM via Multi-Feature Fusion." Ieee Access 8: 144205-144217.