Deep Learning Based Recognition of Lepidoptera Insects

Chao He
School of Computer Science and Engineering
Xi'an Technological University
Xi’an, 710021, China
E-mail:1061410265@qq.com

Pingping Liu
School of Computer Science and Engineering
Xi'an Technological University
Xi’an, 710021, China
E-mail: 1341369601@qq.com

***Abstract*—The successful application of cutting-edge computer vision technology to automatic insect classification has long been a focus of research in insect taxonomy. The results of this research have a wide range of applications in areas such as environmental monitoring, pest diagnosis and epidemiology. However, there is still a gap between the current techniques used in automatic insect classification and the latest computer vision techniques. The research in this paper is conducted on Lepidoptera, a class of insects that are widely infested, including butterflies and moths. The study focuses on the application of deep learning algorithms in image processing of Lepidoptera insects. In order to improve the recognition rate for Lepidoptera insect recognition, this paper uses a detection model based on deep neural networks to realize the recognition of Lepidoptera insects in complex environments. Specifically, the yolov7 algorithm is adopted as the basic model for this experiment, and the reasons for using this model are explained in terms of the splicing of network modules, loss function, positive sample allocation strategy, and the merging of convolution and normalization, respectively. Through experiments, it is proved that the algorithm can effectively improve the gesture recognition rate, the recognition accuracy reaches 79.5%, and the recognition speed is as high as 33.08it/s.**

Keywords-Deep Learning; Deep Neural Networks; Yolov7; Recognition of Lepidopteran Insects

I. INTRODUCTION

Agriculture has always been a major development industry in China, and the upgrading and transformation of agricultural technology is one of the important elements in the modernization of science and technology. Especially in the fields of plantation and forestry, which are important components of agricultural production and sustainable ecological environment construction, there is an urgent need for various technological innovations. Lepidoptera is one of the main pests faced by plantation and forestry and one of the most numerous pest categories, including moths and butterfly insects.

Effective and accurate identification of Lepidoptera is crucial for pest control of agricultural and forestry crops. In the past, monitoring of lepidopteran insects was usually done by manual multi-point sampling and relying on naked eye observation and subjective experience for identification; however, there are some problems with this method of manual judgment. Image judgment relying on the human eye is prone to misjudgment. Moreover, the results of manual judgment are not only unstable in accuracy, but also cannot be carried out dynamically in real time, which also wastes a lot of human and financial resources. Therefore, it is necessary to introduce computer technology and deep learning and other techniques for lepidopteran insect recognition. Through techniques such as image processing, feature extraction and pattern recognition [1], automated and accurate insect recognition can be realized. This method not only improves the accuracy and stability of recognition [2], but also realizes real-time dynamic monitoring and saves human and financial resources to meet the needs of actual production.

In recent years, with the development of deep learning technology, deep learning methods, such as convolutional neural networks, have also been applied in China to achieve more accurate and automated insect recognition. These methods can automatically learn the features of insects from raw images through end-to-end training, and perform classification and recognition. However,

DOI: 10.2478/ijanmc-2023-0073

in practical application scenarios, the detection speed of the target is more important than the detection accuracy of the target. Since Lepidoptera insects belong to the target detection of small objects and they are in constant motion, it is necessary to constantly capture the position of the insect in order to detect it accordingly. Therefore, in this paper, YOLOv7, which has been performing better in the field of target detection recently, is used as the network structure to complete the research on the recognition of lepidopteran insects.

II. Related Works

The country has paid more and more attention to the development of agriculture. Target detection of insects is a hot research category in the field of computer vision combined with agriculture. This kind of research can be applied to the cultivation of agricultural products, environmental detection, agricultural insects and pest prevention and other disciplines. It can effectively save the time of agricultural researchers and better understand the benefits and hazards of insects on agricultural products at different times and environments, which is of great significance to promote agricultural development and pest control. In addition, although computer vision [3] based insect identification [4] is available today, most of its applications still lag behind the cutting-edge computer vision technology.

Traditional insect recognition algorithms mainly classify and identify insects by their color, stripe, shape and other features [5]. However, the actual effect of such algorithms is not good, because the variety of insects is very large, and some insects have very similar appearance characteristics, which makes it easy to recognize insects into the wrong category. Although the target detection algorithm based on deep learning has been developed for nearly a decade, most of its applications are used in traditional industry and manufacturing, based on the natural ecology and agriculture is still in a developmental stage, in order to better develop agriculture on a large scale, so the use of computer technology to assist in the development of agriculture, the environment and other fields is very necessary.

In recent years, artificial intelligence has developed and made significant breakthroughs in the field of target detection [6], and a variety of network structures have appeared, which have shown good performance in both large-volume and microbial target detection. Most of the network structures are based on deep learning for deep convolution of images [7], extracting large features from large images[8], and then downsampling on the basis of large images to extract features based on small images. Finally the features are fused and the positive sample with minimum loss function is selected as the final output value.

III. Introduction to Algorithms

A. Introduction to Target Detection Algorithms

Target detection algorithms are mainly categorized into two types, one is traditional target detection algorithms and the other is deep learning based target detection algorithms [9]. Deep learning based target detection algorithms are further divided into two stages, One-Stage and Two-Stage. Two-Stage has better accuracy, such as the RCNN algorithm [10], whose main method of target detection is to find out a bunch of similar targets by selective search first, and then classify the similar targets. And One-Stage has faster speed, such as YOLO algorithm [11], which mainly solves the regression problem, unlike Two-Stage which solves the classification problem. The main purpose of this paper is to detect the target of Lepidoptera in a complex environment, which requires high real-time monitoring, so the YOLOv7 algorithm, which has a faster detection speed, is used in this paper.

B. Network structure of YOLOv7

YOLOv7 is proposed in 2022 based on YOLOv4 and YOLOv5 [12], which is the current target detection algorithm with high detection speed and detection accuracy in the field of target detection. Its model framework mainly consists of four modules: input, backbone network, feature fusion, and prediction head, as shown in Fig.1, this model can better realize real-time target detection.YOLOv7 model, as a One-Stage target detection algorithm, can be processed by direct regression once to obtain the target area, location information, and category of the corresponding object. Compared with the Two-Stage target detection algorithm, it can locate the target area more quickly and its accuracy is relatively balanced, which lays a good foundation for the recognition of lepidopteran insects in this paper.

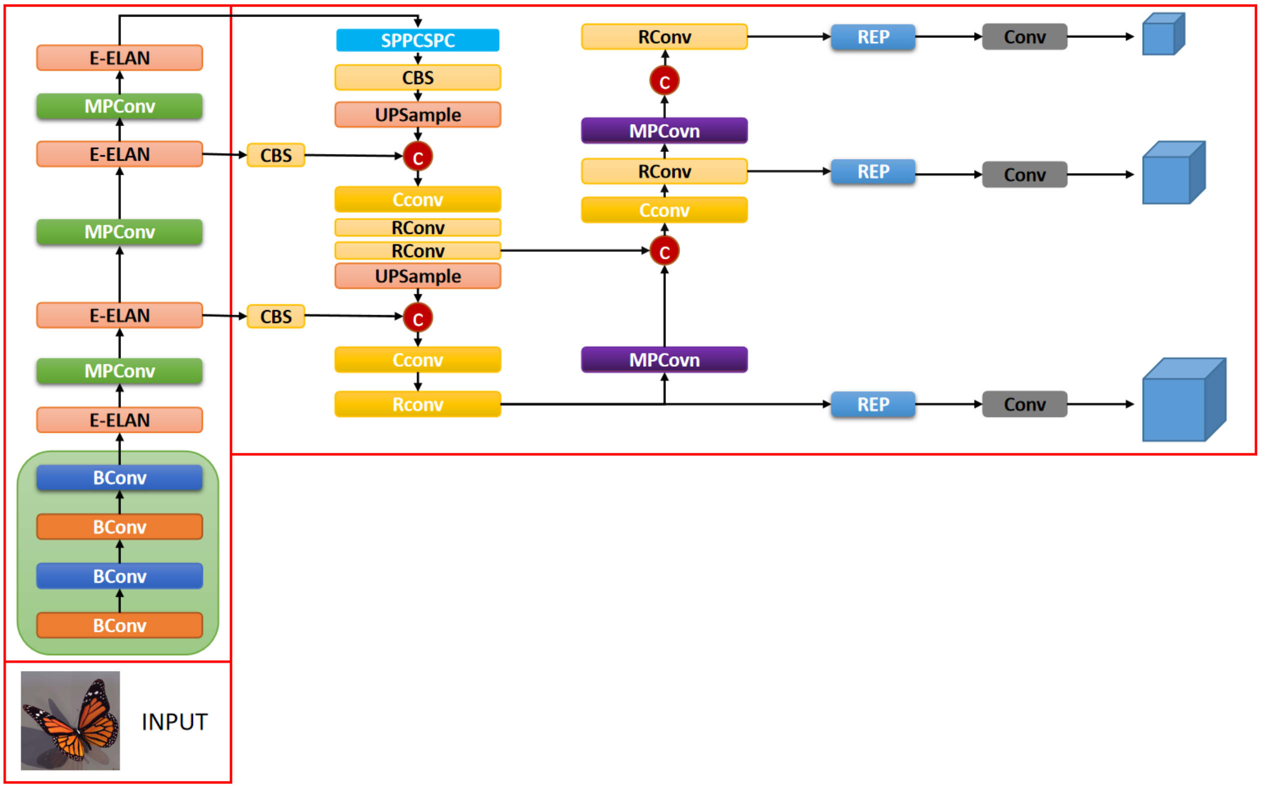


Figure 1. Network structure of YOLOv7.

C. Feature Splicing

The YOLOv7 network structure follows the YOLOv5's feature map splicing, which is performed in both the input network module and the output network module. As shown in Fig. 1, a large number of E-ELAN and MPConv layers are embedded in the network structure after image input, and the main operation of these layers is to perform feature map splicing. The advantage of this is that more comprehensive feature information can be obtained in the training phase, making the trained feature information richer. Since there are many splicing operations in this network structure, this paper mainly introduces the E-ELAN layer and MPConv layer.

In ordinary convolutional structures, basically the results of the previous layer convolution are used as the input value of the next layer convolution, as shown in Fig. 2, while in the E-ELAN layer of YOLOv7, the results of the convolution of the first 1, 3, 5, and 6 layers are merged, and the value is used as the input value of the next layer convolution. The purpose of this operation is that in the training process, it is not certain that the results of the upper layer of convolution must be better than the results of the previous convolution, so a few results of the upper layer of convolution are merged and used as the input value of the lower layer, which makes the feature maps more comprehensive and avoids the loss of important features in the downward training.

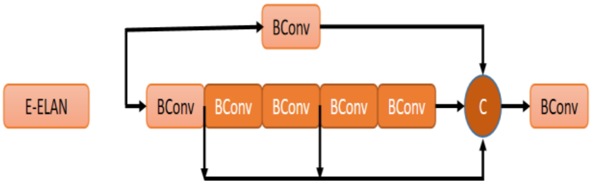


Figure 2. Network structure of the E-ELAN layer.

In the pooling layer, there is also a splicing operation similar to E-ELAN. For ordinary network structures, basically only MaxPooling is used in the pooling layer to compress their feature maps. However, in the YOLOv7 network model, the same feature map is compressed twice. As shown in Fig. 3, the first time the feature map is convolved after MaxPooling. Since convolution of larger feature maps with smaller convolution kernels can also result in feature map compression, 2 convolution kernels are used in the second compression to compress the original feature map. After that, the feature maps compressed by the above two methods are merged. This operation can make the compressed feature map more balanced than the traditional pooling operation without knowing which method is better for compressing the feature map.

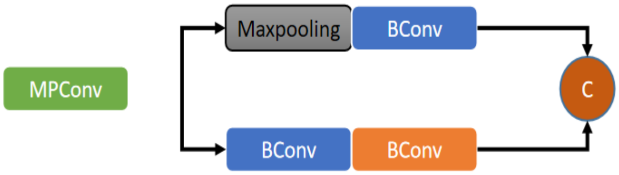


Figure 3. Network structure of the MPConv layer.

D. Loss function

This studyThe loss function used in this experiment is the loss function of YOLOv7 model, which mainly includes three parts. The first part is the target category lossclass loss, this part of the loss function is used to calculate the prediction accuracy of the model on the target category, generally use the cross-entropy loss function to calculate the prediction error of the target category, the formula is shown in equation 1. The second part is the bounding box position loss box loss, this part of the loss function is used to calculate the model's prediction accuracy of the target bounding box position, usually using the Smooth L1 loss function Smooth L1 Loss to calculate the positional difference between the predicted bounding box and the real bounding box, and its formula is shown in equation 2. The third part of the target confidence loss object loss, this part of the loss function is used to calculate the prediction accuracy of the model on the target confidence, usually using the two-classified cross-entropy loss function to categorize the target and the background, and its formula is shown in equation 3.

$$L=-\sum i=1n\left[yi*ln\ (pi)+\ (1-yi)*ln\ (1-pi)\right] \quad (1)$$

$$SmoolthL_1=\begin{matrix}|x|-0.5,|x|>1\\0.5x^2,|x|<1\end{matrix} \quad (2)$$

$$Loss=-\ i=\sum_1^n\left(yilog\,(yi)+\ (1-yi)log\,(1-\ yi)\right) \quad (3)$$

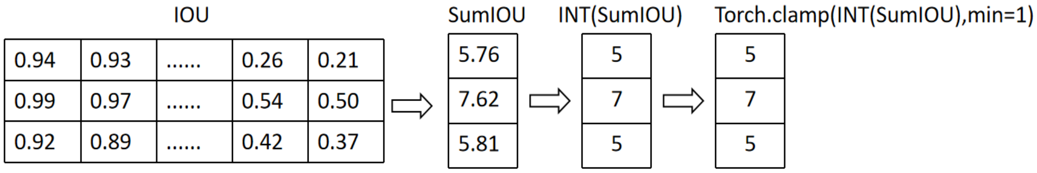
When calculating the total loss function, the loss values of the three parts need to be weighted and summed, where the weights of the target category loss and the bounding box position loss can be adjusted according to the actual task requirements. By using these three parts of the loss function, YOLOv7 can optimize the target category prediction, bounding box position prediction and target confidence prediction at the same time, thus improving the accuracy of target detection. Compared to the YOLOv5 model loss function module, YOLOv7 may improve the comprehensive performance by making improvements to the loss function. It may optimize the weights of the loss function, the way it is combined, or the calculation method based on the experience and feedback from previous versions to further improve the detection and localization capabilities of the model.

E. Positive sample allocation strategy

The more the number of positive samples for target detection, the better it is for model training. However, in most networks for target detection, the number of negative samples is much larger than the number of positive samples due to the different allocation strategies [13]. For example, in YOLOv3, YOLOv4 versions, after the ground truth is IOU'd with each anchor, the anchor with the largest IOU value is selected as a positive sample candidate. The number of candidate frames in an anchor is fixed, so it can only select the appropriate positive samples from the candidate frames in this anchor, which leads to the number of positive samples is too small, and the results of model training may not be stable enough. In YOLOv7, when selecting the positive samples, firstly, it continues the characteristics of YOLOv3 and YOLOv4 to select the anchor with the largest IOU value, and secondly, it will offset the ground truth by 0.5 in each of the four directions, and then it will carry out the IOU operation, and select the two anchors with the largest IOU value, which results in three anchors, and the positive samples will be selected from this anchor. The number of positive samples is expanded by selecting the 2 anchors with the largest IOU values.

After expanding the number of positive samples, not every positive sample can be the final output, so this gives rise to the positive sample allocation strategy. The allocation strategy of positive samples in YOLOv7 is mainly divided into three parts, firstly, the comparison of the aspect ratio of ground truth and anchor is screened, secondly, it is screened by IOU [14], and finally, it is screened by calculating the category prediction loss.

For the screening of aspect ratio and the screening of category prediction loss is relatively simple. So this section focuses on the computation of loss of IOU. The screening of IOU in YOLOv7 is done by adding and rounding the IOU values of all candidate boxes of an anchor, and then selecting the first integer IOU values as the values for the next screening. For example, the need to predict the number of positive samples for the 3, an anchor of the candidate box for 10, the IOU calculation will produce 10 IOU values, by adding the 10 IOU values and rounding, you can get this anchor can be the next screening of the number of candidates, as shown in Figure 4.



IOU

0.94	0.93	......	0.26	0.21
0.99	0.97	......	0.54	0.50
0.92	0.89	......	0.42	0.37

SumIOU	INT(SumIOU)	Torch.clamp(INT(SumIOU),min=1)
5.76	5	5
7.62	7	7
5.81	5	5

Figure 4. IOU loss calculation process.

F. Merging Convolution and Normalization

Since the algorithm used in this paper is One-Stage and the application scenarios are more complex, the speed of target detection is an important factor. Compared to the previous versions of YOLO, in the testing stage, it is generally after the feature convolution of the target object[15], and then each channel of the convolution of the normalization operation, such a process will need to spend nearly twice the time. However, if the feature convolution and normalization are combined into a new convolution then only one convolution operation is needed to complete the two operations, which greatly saves the testing time and thus improves the speed.

The operation to normalize the data of a batch in the RepConv layer in YOLOv7 is equation 4.

$$\hat{x}_i = \gamma \frac{x_i - \mu}{\sqrt{\sigma^2 + \varepsilon}} + \beta = \gamma \frac{x_i}{\sqrt{\sigma^2 + \varepsilon}} + \beta - \frac{\gamma\mu}{\sqrt{\sigma^2 + \varepsilon}} \quad (4)$$

The normalization operation of the RepConv layer is disassembled to obtain equation 5.

The $\hat{F}_{C,i,j}$ in equation 5 represents the result after normalization for each channel, $F_{C,i,j}$ represents each channel, $\frac{\gamma 1}{\sqrt{\hat{\sigma}_C^2 + \varepsilon}}$ represents the weights by which the normalization scales the features, and $\beta_C - \gamma_C \frac{\hat{\mu}_C}{\sqrt{\hat{\sigma}_C^2 + \varepsilon}}$ represents the offsets. After observing equation 5, its form similar to the convolution form of WX+B.

$$\begin{pmatrix} \hat{F}_{1,i,j} \\ \hat{F}_{2,i,j} \\ \vdots \\ \hat{F}_{C-1,i,j} \\ \hat{F}_{C,i,j} \end{pmatrix} = \begin{pmatrix} \frac{\gamma 1}{\sqrt{\hat{\sigma}_1^2+\varepsilon}} & 0 & \cdots & \cdots & 0 & \\ 0 & \frac{\gamma 1}{\sqrt{\hat{\sigma}_2^2+\varepsilon}} & 0 \cdots & \cdots & 0 & \\ \vdots & \cdots & \ddots & \cdots & \vdots & \\ 0 & \cdots & \cdots \cdots & \frac{\gamma 1}{\sqrt{\hat{\sigma}_{C-1}^2+\varepsilon}} & 0 & \\ 0 & \cdots & \cdots & \cdots & \frac{\gamma 1}{\sqrt{\hat{\sigma}_C^2+\varepsilon}} & 0 \end{pmatrix} \bullet \begin{pmatrix} F_{1,i,j} \\ F_{2,i,j} \\ \vdots \\ F_{C-1,i,j} \\ F_{C,i,j} \end{pmatrix} + \begin{pmatrix} \beta_1 - \gamma_1 \frac{\hat{\mu}_1}{\sqrt{\hat{\sigma}_1^2+\varepsilon}} \\ \beta_2 - \gamma_2 \frac{\hat{\mu}_2}{\sqrt{\hat{\sigma}_2^2+\varepsilon}} \\ \vdots \\ \beta_{C-1} - \gamma_{C-1} \frac{\hat{\mu}_{C-1}}{\sqrt{\hat{\sigma}_{C-1}^2+\varepsilon}} \\ \beta_C - \gamma_C \frac{\hat{\mu}_C}{\sqrt{\hat{\sigma}_C^2+\varepsilon}} \end{pmatrix} \quad (5)$$

Since the merging of convolution and normalization is performed in the testing phase after the training is completed, not only the convolution kernel for feature convolution and each input channel can be obtained in this phase. The convolution kernel for the normalization operation and the channels that have undergone feature convolution can also be obtained. The expression for fusing feature convolution and normalization can be written as equation 6. Where $W_{conv} \bullet f_{i,j} + b_{conv}$ generation is the feature result after convolution of the input channel. Then multiplying its feature result by the normalized weight W_{BN} and adding the offset b_{BN} gives the result of normalizing the feature convolution for that channel $\widehat{f}_{i,j}$.

$$\widehat{f}_{i,j} = W_{BN} \bullet W_{conv} \bullet f_{i,j} + (W_{BN} \bullet b_{conv} + b_{BN}) \quad (6)$$

By looking at the expansion of Eq. 3 one can merge the normalized convolution kernel with the feature convolution kernel to obtain a new convolution kernel. The input channel is always unchanged. The new bias is obtained by the normalization operation. Thus the merging of convolution and normalization is done and can be written as equation 7.

$$\widehat{f}_{i,j} = W_{BN,Conv} \bullet X_{f_{i,j}} + b_{BN,Conv} \quad (7)$$

IV. EXPERIMENTAL SETUP AND ANALYSIS

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper.

A. Data set construction

Since most of the lepidopteran insects belong to the pest type at present, the IDADP Agricultural Pests and Diseases Research Atlas was used in this paper. However, since it does not have a separate database for Lepidoptera, this paper adopts its butterfly and moth related datasets, and collects some related images of other Lepidoptera by web crawling. The images are labeled with LabelImg, the export type is YOLO type, and the categories are Islepidoptera and Nolepidoptera, which finally form a lepidopteran dataset with 5000 pictures.

B. Experimental platforms

The experimental server for this experiment is CPU for AMD Ryzen 9 7940H w, GPU for NVIDIA GeForce RTX 4050/PCle, operating system is windows 11, memory is 16GB, training environment is pytorch. Data is randomly divided according to 8:1:1, put into the grid for training iteration 300 epochs, batch_size16, training image size set to 640×640. The data was randomly divided according to 8:1:1, and put into the grid for 300 epochs, batch_size16, and the training image size was set to 640×640.

C. Evaluation indicators

In this experiment, in order to evaluate the performance of the model, the average accuracy mAP, recall, precision, and specific mathematical formulas are used as shown in the following equation. Where c is the number of categories, APi is the precision rate of the first category, TP is the positive sample of correctly predicted samples, and FP is the negative sample of incorrectly judged positive samples.

$$mAP = \frac{\sum_{i=1}^{c} APi}{c} \tag{8}$$

$$\mathrm{Re}\,call = \frac{TP}{TP + FN} \tag{9}$$

$$\mathrm{Pr}\,ecision = \frac{TP}{TP + FP} \tag{10}$$

The performance of this paper's YOLOv7-based algorithm is tested using the test set divided in the lepidopteran insect dataset, and comparative experiments with YOLOv5m6 and YOLOv5s6 algorithms, respectively, so as to validate the performance of this paper's YOLOv7-based algorithm in the detection and identification of lepidopteran insects, which is mainly the comparative experiments based on the test set. In this experiment, in order to evaluate the performance of the model, the average accuracy mAP, recall, precision, and specific mathematical formulas are used as shown in the following equation. Where c is the number of categories, APi is the precision rate of the first category, TP is the positive sample of correctly predicted samples, and FP is the negative sample of incorrectly judged positive samples.

After training the algorithm on YOLOv5m6 and YOLOv5s6 with the same dataset, validation set, and epoch respectively, the above algorithmic models are tested with the same test set and the average accuracy value as shown in equation 8 and the number of iterations per second of the model to process the data as shown in equation 11 are used as the evaluation metrics. Where iterationNum represents the number of this iteration and Time represents the time spent on this iteration.

$$Speed = \frac{iterationNum}{Time} \tag{11}$$

The results after testing are shown in Table 1.

TABLE I. YOLOv5m6, YOLOv5s6, YOLOv7 Performance Comparison

Arithmetics	mAP/%	Speed
YOLOv5m6	78.6%	22.36it/s
YOLOv5s6	75.8%	24.69it/s
YOLOv7	79.5%	33.08it/s

A comparison of the YOLOv5m6 and YOLOv5s6 algorithmic models shows that YOLOv7 has improved its average accuracy. In YOLOv5, the convolution and normalization operations were performed separately to generate nearly twice the time for these two operations, and YOLOv7 achieved the merger of convolution and normalization in the test phase, so its iteration speed has been greatly improved, so that it can be effectively used in complex real-world scenarios.

D. Loss and accuracy analysis

Because the background of lepidopteran insect recognition in the actual environment is generally more complex, and there may be multiple insects interfering with the recognition in the same background. By observing the graphs of precision and recall, it can be found that the algorithm model has a good performance in both the training set and the validation set, which proves that the algorithm model has a high practicability in real scenarios.

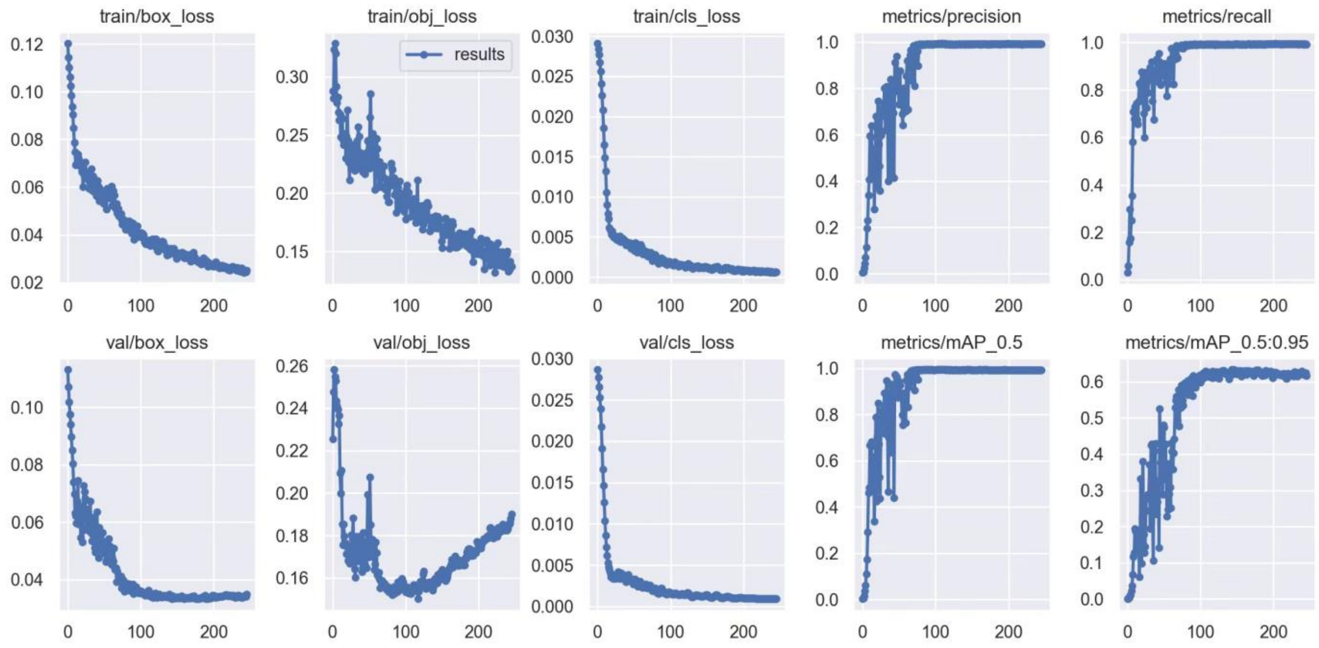
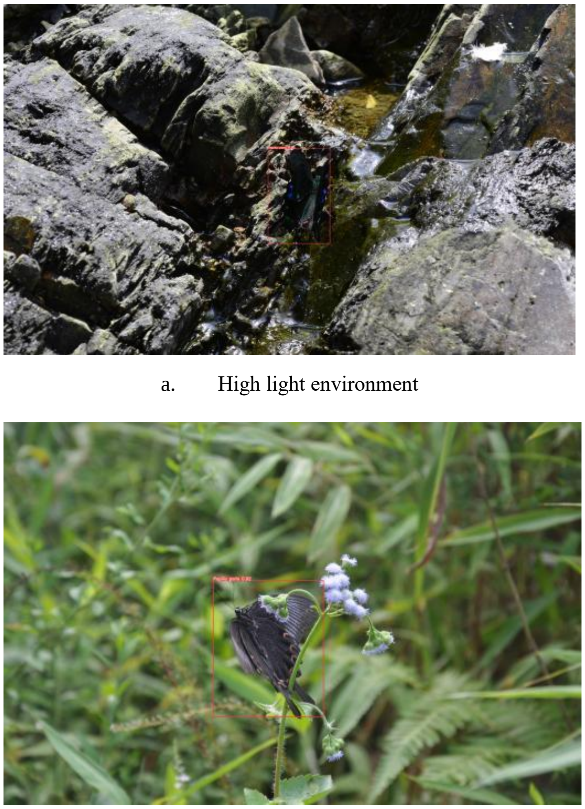


Figure 5. Loss, Precision, Recall, mPA0.5 and mAP0.5-0.95 curves

E. Comparison of test results

In order to more intuitively verify the inspection effect of YOLOv7 on Lepidoptera, in this paper, The respectively tested Lepidoptera in several groups of complex situations, including detection in bright light, detection in ordinary environment and detection in dark light situation, as well as detection in the case of smaller targets and farther distance.

As shown in Figure 6, the model trained by YOLOv7 has better test results no matter in complex situations such as strong light or dark light, and its targets almost always fall in the correct candidate box, with few false checks and fewer misses, and the detection confidence is also relatively high. It proves that the detection ability of the model as well as the feature extraction ability of the YOLOv7 model in such complex situations presents a good performance.



a. High light environment

b. High light environment

Figure 6. Lepidoptera Insect Detection Effect

V. Conclusions

In this paper, a deep learning algorithm is used to accomplish the task of detecting lepidopterous insects. In the preliminary experiments, it is considered that this kind of target detection is mainly used in complex and high real-time environment, so the YOLOv7 model with faster detection speed is used, and according to the analysis of the results of the comparative experiments, it can be seen that the algorithm model is indeed much faster than the other models in the detection speed, which can reach 33.08it/s, but the improvement of the accuracy is not obvious. Misdetection of Lepidoptera is easy to occur in some occasions where insects are gathered, but occasional misdetection in the actual agricultural production scenarios will not affect its overall use, and the speed can be used to detect insects several times, thus enhancing the correct detection of the target.

Based on the analysis of the experiments, its main problem lies in two aspects, lack of sufficient samples and background interference. For the Lepidoptera detection task, a limited number of Lepidoptera species and samples in the training dataset may result in the model performing poorly on unseen Lepidoptera classes or variants. The solution is to collect more Lepidoptera image data and ensure that the dataset contains diverse Lepidoptera species and postures. Secondly, Lepidoptera may appear in a variety of complex backgrounds, such as flowers and leaves. For target detection algorithms like YOLOv7, if the background is similar to the color and texture of lepidopteran insects, it may lead to false or missed detection. One improvement is to use semantic segmentation networks or other image processing techniques to extract regions of butterflies and use them as inputs for target detection to minimize background interference. For the above problems, it is advisable to continue to review the information and references later on to improve the settings of the network structure to enhance the performance of the detection of lepidopterous insects.

References

[1] YANG Hongwei, ZHANG Yun. Progress in the application of computer vision technology in insect identification [J]. Bioinformatics, 2005, (03):133-136.

[2] Yu M C, Yu T, Wang S C, et al. Big data small footprint: the design of a low-power classifier for detecting transportation modes [J]. Proceedings of the Vldb Endowment, 2014, 7(13):1429- 1440.

[3] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2016: 779－788.

[4] Zhu Leqing, Zhang Daxing, Zhang Zhen. Image recognition of lepidopteran insects based on color names and OppcmmtSIFT features [J]. Journal of Entomology, 2015, 58(12).

[5] ZHANG Lei, CHEN Xiaolin, HOU Xinwen et al. Construction and testing of an automatic digital image recognition system for insects of the genus Drosophila in the family Psittacidae [J]. Journal of Entomology, 2011, 54(2):184-196.

[6] Redmon, 3oseph, and Ali Farhadi. YOL09000: Better; Faster, Stronger. Conference on Computer Vision and Pattern Recognition, 6517-6525, 2017.[4].

[7] Girshick R. Fast R-CNN [C]. Proceedings of ICCV 2015, 2015.

[8] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2013:580-587.

[9] He, Kaiming et al. Deep Residual Learning for Image Recognition [C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016): 770-778.

[10] W. Zhang, S. Wang, S. Thachan, J. Chen and Y. Qian. Deconv R-CNN for Small Object Detection on Remote Sensing Images [J]. IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium, 2018:2483-2486.

[11] K. S. Htet and M. M. Sein. Event Analysis for Vehicle Classification using Fast RCNN [C]. 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE), 2020:403-404.

[12] Xing Shanshan, Zhao Wenlong. A review on UAV target detection in complex scenes based on YOLO series algorithms [J]. Computer Application Research, 2020, 37(S2):28-30.

[13] Szegedy C, Wei L, Jia Y, et al. Going deeper with convolutions [C]. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2015.

[14] Matthew D. Zeiler and Rob FergusD. Fleet et al. Visualizing and Understanding Convolutional Networks [C]. ECCV 2014, Part I, LNCS 8689, pp. 818‑833, 2014.

[15] Chen, Yaqian. Research on Quantization and Deployment of Convolutional Neural Networks Based on Nor Flash [D]. University of Science and Technology of China, 2021.