Insect Recognition Under Natural Scenes Using R-FCN With Anchor Boxes Estimation

Hong-Wei Pang[1,2], Peipei Yang[1,2], Xiaolin Chen[3], Yong Wang[3], and Cheng-Lin Liu[1,2]

[1] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2] University of Chinese Academy of Sciences, Beijing, P.R. China
[3] Key Laboratory of Zoological Systematics and Evolution, Institute of Zoology, Chinese Academy of Sciences, Beijing, China
panghongwei17@mails.ucas.ac.cn, ppyang@nlpr.ia.ac.cn, xlchen@ioz.ac.cn, 530468931@qq.com, liucl@nlpr.ia.ac.cn

Abstract. Insect species recognition is an important application of computer vision in zoology and agriculture. Most of existing methods resort to hand-crafted features and traditional classifiers, which usually give poor accuracy and apply only to elaborately taken full-size pictures. In this paper, we focus on a more challenging case where the images are taken in the wild with complex backgrounds, and propose to use a deep learning based detection model to deal with it. It exploits multi-class object detection to eliminate interferences from complex backgrounds, while taking advantages of deep learning to significantly improve the performance of recognition. After evaluating several popular detection methods, R-FCN is selected as the base model. To further improve its performance, we introduce a clustering algorithm for estimation of the anchor boxes instead of using predefined ones. The experimental results on a dataset of insect images collected in the wild prove the effectiveness of our proposed method in improving both accuracy and speed.

Keywords: multi-class object detection, r-fcn, deep learning, insect recognition, natural scenes

1 Introduction

Many species of insects are damaging to vegetables, fruits or other crops, which might have negative impact on agriculture and related international trade [1]. This requires rapid and accurate species identification where expertise is often lacking. Automatic or semi-automatic identification of insects are greatly needed for diagnosing causes of damage and quarantine protocols for the economically relevant species. Some computer-aided systems have been implemented in recent decades for identifying harmful species [2–6], and the intelligent recognition approaches for insects living in natural scenes have also made some progresses recently [7–9].

For example, Favret et al [4] used a sparse processing technique and support vector machine to successfully recognize the specimens. Wang et al [5] proposed to use Gabor surface features in automated identification to further improve the recognition accuracy. Deng et al [7] proposed to recognize the insect living in natural scene using natural statistics (SUN) model. In spite of the obvious progresses in insect recognition, existing approaches suffer from the following two limitations.

First, these methods apply to almost only elaborately taken pictures. Earlier approaches extract hand-crafted features and then predict the label of an image with a traditional classifier. They are effective for the images with clean background such as specimens, but liable to fail on the images taken in the wild due to their complex backgrounds. They can neither deal with the case where there are more than one species in an image. Recently, there appear a few works applying detection techniques to insect recognition [7, 9, 10]. However, in these works, the images are always taken elaborately, where the insects occupy most of the space in the image with a simple background as shown in Fig. 1. Up to now, we have not seen a method that recognizes insects freely taken in real natural scenes with complex backgrounds, which seriously limits its applicability.

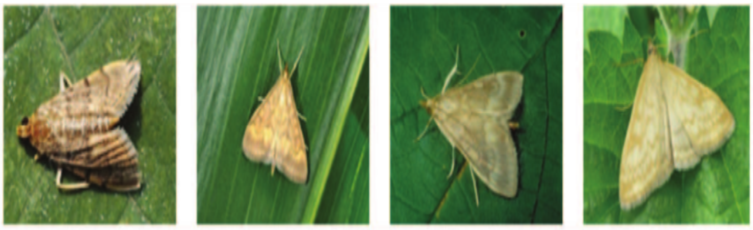


Fig. 1. The images are almost taken elaborately, where the insects occupy most of the space in the image with a simple background.

Second, with the breakthrough of deep learning technology, computer vision made considerable progress and has been successfully applied in many fields. The deep learning has been the state-of-the-art technique for both image classification and object detection. Meanwhile, most of existing insect recognition approaches still resort to traditional classifiers with hand-crafted features. Thus, they cannot take full advantages of deep learning in its high accuracy and robustness.

In this paper, we propose an insect recognition for images taken freely under natural scenes by exploiting the technique of deep learning based multi-class object detection. Our approach exceeds previous works in the following aspects. (1) We focus on direct recognition of the images taken freely under natural scenes, which are obviously different from the ones appearing in previous works as shown in Fig. 1 and Fig. 2. The backgrounds are much more complex and the insects occupy only small space in the images. This makes the recognition much more difficult. (2) We exploit multi-class object detection with deep learning for insect recognition. On one hand, the multi-class object detection model discovers the area of insect to be recognized directly from the image, rather

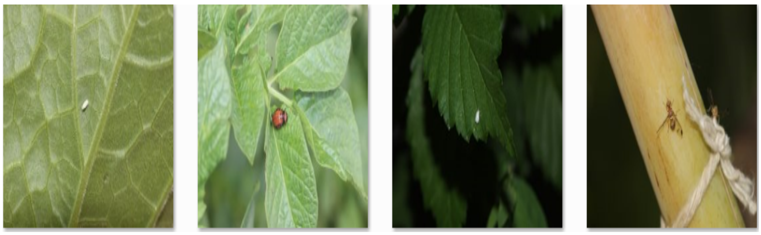


Fig. 2. The images in our dataset are taken in the wild more freely. Without preprocessing, insects occupy only small space of the whole images with complex backgrounds.

than recognizing the whole image, and thus can effectively eliminate the interferences from complex natural backgrounds. It also easily solve the problem of multiple objects in one image. On the other hand, it exploits deep learning that is the state-of-the-art method in computer vision to significantly improve the recognition accuracy. Furthermore, we evaluate several currently popular deep learning detection methods on our dataset and select R-FCN that gives the best performance as our base model. (3) We introduce the technique of estimating the anchor boxes by a clustering algorithm into R-FCN. It supplies more appropriate anchor boxes to the detection algorithm, which improves the recognition accuracy and reduces the training time.

The rest of this paper is organized as follows. Section 2 introduces the dataset for insect recognition in the wild and the selection of the base detection method. Section 3 introduces our insect recognition based on R-FCN with estimation of anchor boxes. Section 4 details the experimental setting and results, and section 5 concludes this paper.

2 Dataset construction and selection of basic model

In this section, we will give a introduction to the dataset for insect recognition under natural scenes. Then we evaluate three object detection methods on this dataset, and select the one giving the best performance to be our base model.

2.1 Dataset construction

As stated in Section 1, we aim at recognizing the insects from images taken under natural scenes. To obtain the insect recognition model, we first build the dataset. There are 19 species of insects to be recognized, and the images are collected by researchers of Key Laboratory of Zoological Systematics and Evolution, Institute of Zoology, Chinese Academy of Sciences. They travel to Yunnan Province, Xinjiang Province, Hebei Province, and suburb of Bejing City in China for data acquisition. The images are taken freely in the wild and there are often complex backgrounds. Some examples of the data are shown in Fig. 2.

We then build our dataset using the collected images. There are a total of 4,538 samples in 19 classes in our dataset. Each sample is an image of insect whose resolution is 5000×4000. Every class corresponds to an insect species to be recognized, and the names of the species are listed in Table 1. For each image, we annotate the objects with bounding boxes, and assign their labels to them. For convenience of evaluation, we divide the data into a training set and a test set with a ratio of 3 : 1. The number of samples in each class and the division of training/test set are also listed in Table 1.

Table 1. The names, numbers of samples, and the division of training/test set for all species in our dataset.

No.	**Species**	**images**	**train**	**test**
1	*Bemisia tabaci*	120	90	30
2	*Bactrocera dorsalis*	295	221	74
3	*Phenacoccus solenopsis*	27	20	7
4	*Vinsonia stellifera*	180	135	45
5	*Rhyncophorus ferrugineuss*	338	253	85
6	*Lissorbqptrus oryzqphilus*	404	303	101
7	*Leptinotarsa decemlineata*	412	309	103
8	*Bactrocera correcta*	195	146	49
9	*Anoplophora sp*	333	250	83
10	*Agrilus mali*	307	230	77
11	*Aleurodes proletella*	189	142	47
12	*Bactrocera cucurbitae*	121	91	30
13	*Batracomorphus pandaru*	224	168	56
14	*Carpomya vesuviana*	499	374	125
15	*Ceroplastes rubens*	152	114	38
16	*Cydia pomonella*	273	205	68
17	*Eulecanium rugulosum*	92	69	23
18	*Hyphantria cunea*	213	159	54
19	*Stilprotia salicis*	164	123	41

2.2 Selection of the base model

The most straightforward way to implement the insect recognition is to use a classifier to predict its class from an image of the insect [11–17]. However, since our images are collected freely in the wild, the insects to be recognized usually occupy only a small proportion in the images and there exist complex backgrounds. This obviously affects the recognition accuracy, because the classifier is incapable to well distinguish the insect to recognize and the background. Considering that our insect recognition may be used in various applications, where the backgrounds of images are various and different from the samples in dataset, the recognition accuracy will further deteriorate in practical usage.

One way to eliminate the impact of the background is to cut the region of the insect from the image and then apply the classification. It is evidently avoid the interferences from background. However, users have to cut the region of insect before recognition, which makes it inconvenient to use. In this paper, we propose to use the multi-class object detection to recognize the insects under complex backgrounds. It is capable to detect the object to recognize directly from the image, giving both the class of the object and its location. It can effectively facilitate the insect recognition while obtaining high accuracy.

To select a proper detection model for our insect recognition, we evaluate three currently popular detection methods on our dataset: Faster R-CNN [18], YOLO [19], and R-FCN [20]. Among them, Faster R-CNN and R-FCN are two-stage methods which include a process of Region Proposal Network (RPN), while YOLO belongs to one-stage methods that directly predict class and location of object on the feature map.

We evaluate the three methods above on our insect recognition dataset and the results are given in Fig. 4. In our experiments, the R-FCN with ResNet-101 outperforms other methods. On one side, as a two-stage detection method, R-FCN benefits from the RPN that can improve the accuracy of detection especially for small objects such as some insects in our dataset. On the other side, although there is also a region proposal process in Faster R-CNN, the direct full connection of feature map after the ROI pooling layer is not an optimal choice. Full convolution networks such as GoogLeNet [21] and ResNet [22] have proved better performance and adaptation to different scales of the image. Besides, the number of channels in Faster R-CNN ROI pooling layer is very large, resulting in a large amount of calculations for the fully connected layer. In contrast, R-FCN without the fully connected layer significantly reduces the calculation. In summary, the R-FCN is appropriate for our insect recognition and thus we select it as the basic model.

3 Insect recognition using R-FCN With Anchor Boxes Estimation

In this section, we will propose our insect recognition in detail. Firstly, a brief introduction to the base model R-FCN is given. Then we propose to estimate the anchor boxes by clustering of bounding boxes instead of predefined ones in RPN of R-FCN. It effectively improve its performance on small objects while speeding up the detection process. Finally, the algorithm of anchor boxes generation is presented in detail.

3.1 A brief introduction to R-FCN

R-FCN is a region-based, fully convolutional networks for object detection [20]. It proposes the position-sensitive score maps to address a dilemma between translation-invariance in image classification and translation-variance in object detection, which makes it possible to adopt fully convolutional image classifier backbones in object detection. R-FCN usually costs less calculations while keeping high accuracies.

The framework of R-FCN is shown as Fig. 3. The first part is a convolutional network (ResNet-101 is used here) that is used to extract the convolutional features from the images. After feature extraction, a region proposal network (RPN) is exploited to generate region of interest (ROI) on the feature map. At the same time, a k^2-dimensional position-sensitive score map is obtained from the feature map for classification, while a $4k^2$-dimensional position-sensitive score map is for regression. Finally, the class and position information can be obtained by a position-sensitive ROI pooling.

3.2 Improve the detection by anchor boxes estimation

In the original version of R-FCN, the RPN generates ROIs from feature map using 9 predefined anchor boxes of different scales and shapes. The predefined anchor boxes work well in most practical cases. However, in the problem of insect detection, some objects are so small that the predefined anchor boxes cannot cover their shapes. This makes these insects apt to be missed by the detector.

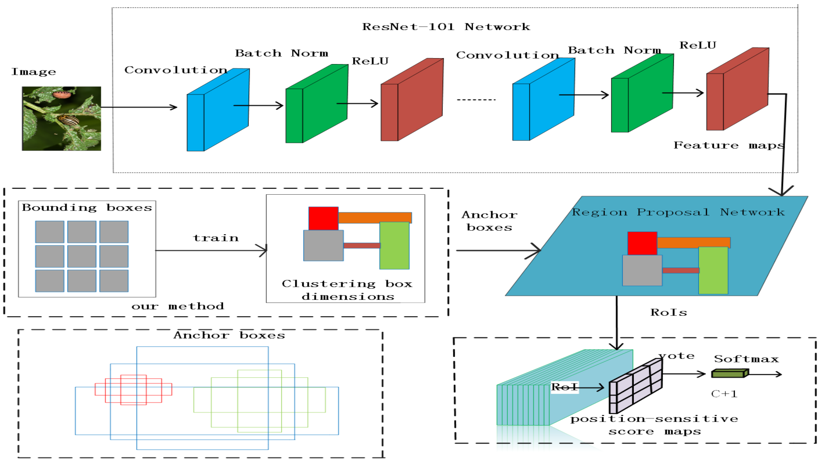


Fig. 3. Proposed framework comprised of R-FCN and the clustering bounding boxes in the region proposal network.

To solve this problem, we can use a set of more appropriate anchor boxes instead of the predefined ones. In Yolov2 [19], the authors propose to set the scale and shape of the anchor boxes according to data. To be specific, the bounding boxes are collected from the training data, and then clustered into several clusters. The scales and shapes of the bounding boxes in cluster centers are used to set the anchor boxes. We also introduce this technique into R-FCN in our insect recognition. Using the anchor boxes estimated in this way, the region proposal network can generate ROIs that adapt better to the data, which improves the performances of detection, especially on small objects. Besides, the algorithm can obtain comparable performances with less anchor boxes, and this efficiently reduces the calculation.

3.3 The anchor boxes estimation algorithm

The detailed anchor boxes estimation algorithm based on clustering of bounding boxes is given in Algorithm 1. In this algorithm, each bounding box in training sample is represented as a point in 2-dimensional space, where the two axes correspond to the width and height of the box. Then these points are clustered using the k-means clustering algorithm [23]. After initialization, the update of the cluster of each point and the center are updated alternately. The criteria to determine the closest point is not the Euclidean distance as traditional k-means algorithm because it always generates greater errors for large boxes. Instead, the distance is measured according to the IOU score as [19]

$$d(box, centroid) = 1 - IOU(box, centroid)$$

Using this algorithm, a set of appropriate anchor boxes are selected, which improves the detection performances and accelerates the calculation.

Algorithm 1 Anchor Boxes Estimation Algorithm By Clustering

```
Input: The set of samples $D = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$, where $\mathbf{x}_i = [x_i^w, x_i^h]^T$, $k$.
    $m$ is the total number of samples, $k$ is the number of clusters, and $x_i^w, x_i^h$ represents
    the width and height of the bounding box of the $i$-th sample.
Output: $C = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_k\}$
Randomly select $k$ samples from $D$ to be the initial values of $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k\}$;
repeat
    $C_i = \emptyset \quad (1 \leq i \leq k)$
    for $j = 1, 2, ...m$ do
        Compute the overlap between edges of the $j$-th box and the $i$-th center:
        $L^w = \min(x_j^w, \mu_i^w)$, $L^h = \min(x_j^h, \mu_i^h)$
        Compute their intersection: $S_I = L^w \times L^h$
        Compute their union: $S_U = x_j^w \times x_j^h + \mu_i^w \times \mu_i^h - S_I$
        Compute IOU: $S_{ji} = S_I / S_U$
        Compute the distance between $\mathbf{x}_j$ and $\boldsymbol{\mu}_i$: $d_{ji} = (1 - S_{ji})$
        Assign $\mathbf{x}_j$ to the cluster with the closest mean:
        $\lambda_j = \arg\min_{i \in \{1,2,\ldots,k\}} d_{ji}$, $C_{\lambda_j} = C_{\lambda_j} \cup \{\mathbf{x}_j\}$
    end for
    for $i = 1, 2, ...k$ do
        Calculate new means: $\boldsymbol{\mu}_i' = \frac{1}{|C_i|} \sum_{\mathbf{x} \in C_i} \mathbf{x}$
        if $\boldsymbol{\mu}_i' \neq \boldsymbol{\mu}_i$ then
            update $\boldsymbol{\mu}_i = \boldsymbol{\mu}_i'$
        end if
    end for
until all $\boldsymbol{\mu}_i$'s keep unchanged
```

4 Experiments

4.1 Data preparation and experimental settings

In this paper, we evaluate our proposed method on the data of insect recognition taken in the wild. The dataset has been proposed in Section 2. Since the original images are of unnecessarily high resolution, we downsample the images to be of resolution 600 × 600 before training or recognition.

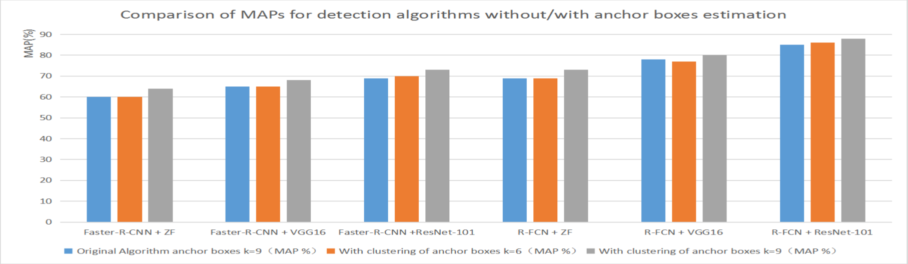


Fig. 4. Comparison of MAPs for original detection algorithms and the ones with anchor boxes estimation, where both 6 and 9 clusters are used in the estimation.

Our experiments are performed a NVIDIA Titan X GPU. In the experiment, the learning rate for ResNet-101 is set to 0.001, the maximum number of iteration is set to $110,000$, the momentum is set to 0.9, and the weight decay is set to 0.0005. The mean average precision (MAP) [24] is used to evaluate the performance of algorithms, which reflects both the recall and precision of the detection algorithm. In our experiment, if the ratio of the intersection of a ground-truth bounding box and a predicted bounding box to the union of them is greater than 0.5, it is regarded as a good prediction.

4.2 Experimental results

The detection results are shown in Fig. 4 and Fig. 5 in MAP. Comparing the accuracies and the running time of the detection methods to the ones with clustering of anchor boxes estimation, we obtain the following observations. (1) When we use 6 anchor boxes obtained by clustering in the detection, the algorithms obtain almost the same accuracies as the ones using 9 predefined anchor boxes. With less anchor boxes, the test time is obviously reduced. Thus clustering of anchor boxes is helpful to reduction of the calculations while keeping comparable accuracies. (2) When we use the same number of anchor boxes obtained by clustering as the original algorithms, the performances of detection is improved. This shows that a set of more appropriate anchor boxes are obtained by clustering. In the following, we will illustrate some detection results directly in the images. For brevity of illustration, we use R-FCN-Cl to represent R-FCN with anchor boxes estimation using the clustering algorithm. Fig. 6 illustrates the detection results of *Bemisia tabaci* using Faster R-CNN, R-FCN, and R-FCN-Cl. The objects of *Bemisia tabaci* are relatively small in the images, which are therefore apt to be missed by the detector. Fig. 7 compares the detection results of *Leptinotarsa decemlineata* given by R-CNN and R-CNN-Cl. There are less false retrievals in the result of R-CNN-Cl.

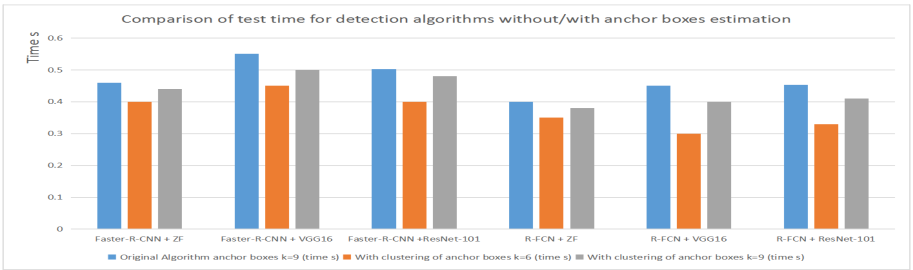
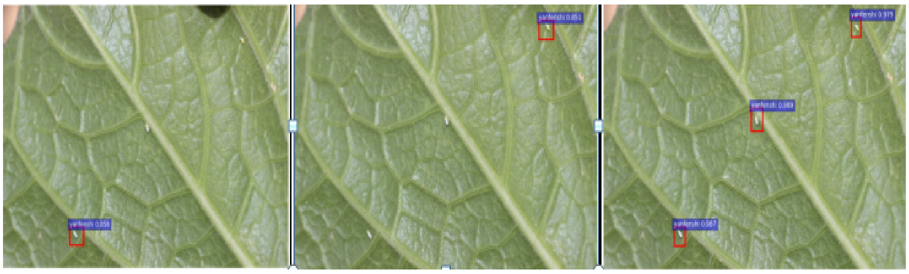


Fig. 5. Comparison of test time for original detection algorithms and the ones with anchor boxes estimation, where both 6 and 9 clusters are used in the estimation.



(1) Faster-R-CNN (2) R-FCN (3) R-FCN-Cl

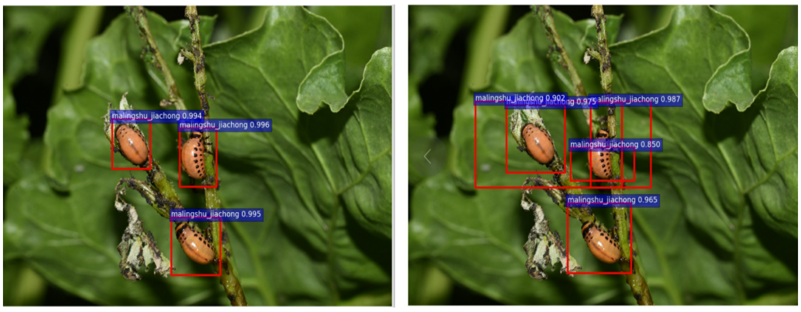
Fig. 6. The detection results of *Bemisia tabaci* using Faster R-CNN, R-FCN, and R-FCN-Cl. All the small objects have been successfully detected using R-FCN-Cl method. However, the Faster-R-CNN and the R-FCN easy missing detect the small objects.

Fig. 8 compares the detected bounding boxes of *Anoplophora sp* given by Faster R-CNN, R-FCN, and R-FCN-Cl, which are shown in red, blue, and green borders. From the image, we see that R-FCN-Cl gives the most appropriate box for the object. Besides, we also show some detection results of other insect species in Fig. 9. In these images, the objects are all correctly detected and recognized even under complex backgrounds. It proves the effectivity of our proposed method.

We also evaluate the performance of our method as an insect recognizer. To be specific, we construct a multi-label classification problem: the labels of an image are defined to be all the insect species that appear in it. Then, we apply the detection algorithm to the image and assign all detected insect species to it as its labels. A sample is correctly classified if and only if all its labels are exactly matched to the ground truth. In our experiments, we obtain a recognition accuracy of 98.5%.

5 Conclusion

In this paper, we propose to use multi-class object detection based on deep learning to solve the problem of insect recognition under natural scenes. On one



(1) R-FCN-Cl (2) R-FCN

Fig. 7. Comparison of detection results on *Leptinotarsa decemlineata* using R-FCN-Cl and R-FCN. There are several obviously false detection given by R-FCN, while R-FCN-Cl gives better results.

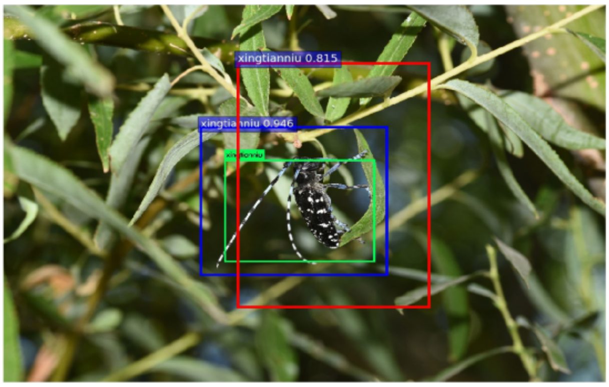


Fig. 8. The detection results of *Anoplophora sp* using Faster R-CNN, R-FCN, and R-FCN-Cl. The red, blue, and green lines represent bounding boxes given by Faster R-CNN, R-FCN, and R-FCN-Cl, respectively.

side, using detection technique, it is capable to accurately recognize the insects in the images even taken freely under complex backgrounds, and easily deal with the case where there are multiple species of insects to be recognized in one image. The positions of the objects in the image are also given by the algorithm. On the other hand, we take advantages of deep learning on object detection to significantly improve the performance of insect recognition, and simplify the recognition process.

We select the R-FCN method for multi-object detection in our insect recognition system after evaluating some popular methods. To further improve the detection accuracy and speed, we propose to design anchor boxes adaptively by clustering. We build a dataset for insect recognition in the wild, where the images are collected under natural scenes. The experimental results show that our method work well on the real data, and anchor boxes estimation by clustering is effective to improve both detection accuracy and speed.

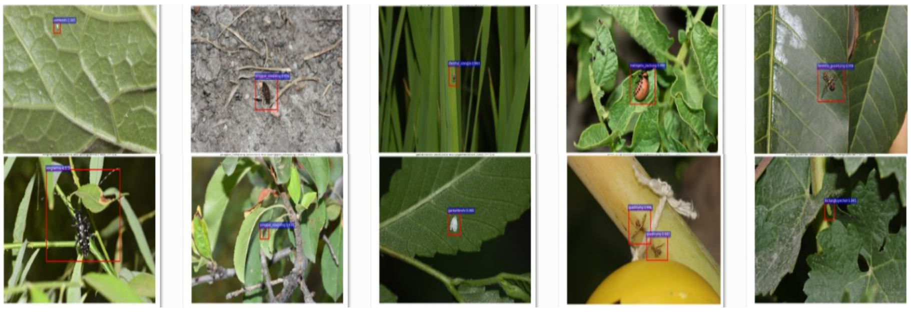


Fig. 9. Illustration of the detection results on some other insect species using R-FCN-Cl.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (NSFC) Grants 61721004, 61836014, the Beijing Municipal Science and Technology Project grant Z181100008918010, and National Key R&D Program of China grant 2017YFC1200602.

References

1. White, I.M., Elson-Harris, M.M., et al.: Fruit flies of economic significance: their identification and bionomics. CAB International (1992)
2. Hassan, S.N.A., Rahman, N., Zaw, Z.: Vision based entomology: a survey. International Journal of Computer Science & Engineering Survey **5** (2014) 19–31
3. MacLeod, N.: Automated taxon identification in systematics: theory, approaches and applications. CRC Press (2007)
4. Favret, C.R., Sieracki, J.M.: Machine vision automated species identification scaled towards production levels. Systematic Entomology **41** (2016) 133–143
5. Wang, J.N., Chen, X.L., Hou, X.W., Zhou, L.B., Zhu, C.D., Ji, L.Q.: Construction, implementation and testing of an image identification system using computer vision methods for fruit flies with economic importance (diptera: Tephritidae). Pest management science **73**(7) (2017) 1511–1528
6. Wang, L., Huang, L., Yang, H., Gao, L., et al.: Developing and testing of image identification system for bactrocera spp. Plant Quarantine (Shanghai) **27**(5) (2013) 29–36
7. Deng, L., Wang, Y., Han, Z., Yu, R.: Research on insect pest image detection and recognition based on bio-inspired methods. Biosystems Engineering **169** (2018) 139 – 148
8. Ebrahimi, M., Khoshtaghaza, M., Minaei, S., Jamshidi, B.: Vision-based pest detection based on svm classification method. Computers and Electronics in Agriculture **137** (2017) 52 – 58
9. Hu, Z., Liu, B., Zhao, Y.: Agricultural robot for intelligent detection of pyralidae insects. In: Agricultural Robots-Fundamentals and Applications. IntechOpen (2018)
10. Zhong, Y., Gao, J., Lei, Q., Zhou, Y.: A vision-based counting and recognition system for flying insects in intelligent agriculture. Sensors **18**(5) (2018) 1489

11. Gassoumi, H., Prasad, N.R., Ellington, J.J.: Neural network-based approach for insect classification in cotton ecosystems. In: International Conference on Intelligent Technologies. (2000) 13–15
12. Asefpour Vakilian, K., Massah, J.: Performance evaluation of a machine vision system for insect pests identification of field crops using artificial neural networks. Archives of Phytopathology and Plant Protection **46** (2013) 1262–1269
13. Wang, J., Lin, C., Ji, L., Liang, A.: A new automatic identification system of insect images at the order level. Knowledge-Based Systems **33** (2012) 102–110
14. Ding, W., Taylor, G.: Automatic moth detection from trap images for pest management. Computers and Electronics in Agriculture **123** (2016) 17–28
15. Ding, W., Taylor, G.: Automatic moth detection from trap images for pest management. Computers and Electronics in Agriculture **123** (2016) 17 – 28
16. Wang, J., Ji, L., Liang, A., Yuan, D.: The identification of butterfly families using content-based image retrieval. Biosystems Engineering **111**(1) (2012) 24 – 32
17. Sun, Y., Cheng, H., Cheng, Q., Zhou, H., Li, M., Fan, Y., Shan, G., Damerow, L., Lammers, P.S., Jones, S.B.: A smart-vision algorithm for counting whiteflies and thrips on sticky traps using two-dimensional fourier transform spectrum. Biosystems engineering **153** (2017) 82–88
18. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. (2015) 91–99
19. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 7263–7271
20. Dai, J., Li, Y., He, K., Sun, J.: R-fcn: Object detection via region-based fully convolutional networks. In: Advances in neural information processing systems. (2016) 379–387
21. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2015) 1–9
22. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 770–778
23. Hartigan, J.A., Wong, M.A.: Algorithm as 136: A k-means clustering algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics) **28**(1) (1979) 100–108
24. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision **88**(2) (2010) 303–338