Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 5s (2025)

Insect Identification System using Faster RCNN with ADAM optimizer Segmentation model

R.Padmavathi[1], Dr. K.Kavitha[2]
Research scholar, Department of Computer Science. Mother Teresa Women's University, Kodaikanal
Assistant professor,Agurchad Manmull jain college,Meenambakkam.
padmabaviya11@gmail.com, padmavathi.r@amjaincollege.edu.in.
Assistant Professor, Department of Computer Science, Mother Teresa Women's University, Kodaikanal.
kavitha.urc@gmail.com.

Article History:

Received: *10-10-2024*

Revised: *27-11-2024*

Accepted: *08-12-2024*

Abstract:

Researchers have been more interested in automated insect recognition in recent years, and many different approaches have been taken to studying the practical implications of this field. When it comes to designing pest management tactics and safeguarding beneficial insects, accurate identification of the insects at play is crucial. Insect target detection has always relied heavily on artificial identification methods; however, deep learning can automatically extract characteristics for detection, solving the issue of poor detection accuracy due to subjective considerations. Here, we present our work on an improved method of insect identification using segmentation-based visual cues. The bug images were segmented using Faster RCNN with the ADAM optimizer, and the features were extracted using InceptionV3. The findings show that our suggested model outperformed competing methods in terms of accuracy and performance.

Keywords: Image Segmentation, Preprocessing. Faster RCNN, ADAM optimizer, InceptionV3

1. Introduction

Approximately 75% of all animal species may be found in the insect kingdom, making them a massive and pivotal part of the natural food chain. Since most insects eat plants, pest infestation is a key issue restricting agricultural productivity because of the negative impact it has on crop quality and yield. Because of this, it's crucial to investigate insect image recognition and discover possible patterns for the development of pest management methods [1]. Insects have traditionally been identified by specialists or professionals with specialized understanding of insects. However, the current pool of classification specialists and technologists is woefully inadequate to satisfy the demands of the ever-expanding number of real-world use cases. However, the varied texture, tiny characteristics, and changeable habitat of insects make insect image identification a challenging image recognition task. The already challenging challenge of picture identification is made much more so by the interspecies resemblance across insect groups & the variances induced by varied gestures and motions. There are so many different kinds of insects that it's simple for humans to overlook important details while building screening features, leading to misclassification [2].

There are so many different kinds of insects that it may be difficult to tell them apart, which in turn threatens the foundations of biodiversity, conservation, and related studies. Traditional insect identification techniques are complex, and there are fewer insect taxonomists available to work on this problem. Taxonomists have been on the lookout for effective strategies to fulfill practical needs in insect identification. In the past two decades, several other computer-based insect identification aids, including as the automatic bee identification system (ABIS), the digital automated identification system (DAISY), BugVisux, and But2fly, have been created and tested. Built in 1995, ABIS is able to recognize bees in the wild by examining digital photos of their wing veins. DAISY is a prototype system used for recognizing and analyzing museum collections; it is based on fingerprint identification technology & is used for identifying insects using digital imagery. There are now 40 bug species that can be identified with BugVisux using morphologic traits, and 43 butterfly species that can be identified with But2fly using color attributes of wings [3].

Target detection approaches for insects have historically relied heavily on visual inspection of the insects themselves, with human eyes comparing observed details to well documented specimen models. This technology, which relies on fabricated insect detection, is plagued by a number of drawbacks, including a high labor cost, inconsistent target detection criteria, poor efficiency, and so on. As machine vision, image processing technology, and deep learning have advanced rapidly in recent years, the deep learning-based target detection technology has emerged as the dominant algorithm in this space, finding applications in areas as diverse as pedestrian detection, vehicle detection, face detection, and driverless cars. This technique may also be used to locate insects as a target. Some of these methods for spotting the corpses of insects, however, are laborious, expensive, and inefficient. One of the hottest subjects in recent years is application design, and one popular way is based on image processing since it is nondestructive and easy to use. Insects may be counted using image processing in a few different ways, the most common of which are by counting the insects' morphological traits, the binary sketch of the grayscale picture, and the insects' pixels. Unfortunately, these techniques suffer from a number of drawbacks, including a lack of precision when counting tiny grain pests, a high price tag for the necessary gear for picture collecting and monitoring, and a computational overhead that prevents real-time detection. Since picture-based insect target detection may address the aforementioned problems with artificial detection, the widespread adoption of advanced computer image recognition technology in this area promises to significantly enhance detection accuracy [4] [5].

Segmentation is the process of dividing a picture into individual, distinct sections that have a common property, such color or texture, but do not overlap. Color, texture, and edges are the three main categories of image attributes. Edge features are simple to extract and are particularly adept at isolating signal from noise. All fourteen characteristics utilized in this study are edges [6]. Automating the identification and counting of insects using data extracted from digital images is possible with the help of modern machine learning algorithms, computer vision techniques, and image processing (IP) [7].

Using a more in-depth network structure, CNN has proven effective in recognizing images in computer vision. Having additional training data is essential because to the deep network topology. Many researchers choose for the ImageNet data set as a starting point for their models since it is widely accepted as the gold standard for image recognition [8]. Because of the availability of machine learning frameworks like TensorFlow and models like Inception and GoogleNet, deep convolutional neural networks have made great strides in recent years. The number of taxa studied, as well as the accuracy and effectiveness of image classifiers used for species identification, have both improved considerably in recent years [9].

While there have been significant advancements in the field of automated insect identification using images, most of these studies have only considered images with ideal conditions, such as bright lighting, a fixed insect location, and a top-down perspective. Good results may be achieved from insect identification systems when using global features—those that provide a direct and broad description of an image or insect object—under conditions of high picture quality. These systems are limited by their susceptibility to noise and background clutter, their insistence on a particular insect position, and the difficulty of their picture acquisition procedure [10]. In this study, we provide a novel method for identifying insects using picture segmentation. For insect image segmentation, we employed Faster RCNN with the ADAM optimizer.

2. Related Works

Using computing resources, recent research has automated the process of identifying and counting insects, worms, and cells in digital photographs. The tests confirm that manual identification is a time-consuming procedure that is prone to mistakes and lacks accuracy, which makes it unsuitable for widespread use. Here, we looked at many approaches for developing insect identification models.

A detection instrument for early detection and identification of agricultural diseases and pests was developed by Lidia Cleetus et al. [11]. To achieve this goal, many deep learning architectures were tested to determine which would best aid in the development of a reliable and productive detection model. In their research, they used Convolutional Neural Network, VGG16, InceptionV3, & Xception deep learning architectures. Pre-trained models built on CNN architecture may be broken down into three distinct groups: VGG16, InceptionV3, and Xception. The Xception model has been shown to be the most effective in detecting and identifying illnesses and pests on tomato leaves after being compared to other models using test accuracy scores. The model achieves an accuracy of 82.89 percent on the tomato disease dataset, and 77.5 percent on the pest dataset.

According to Maxime Martineau et al. [12], Entomology has been used as an indicator of biodiversity and has implications in many other areas of biology. Automated entomology has been available for decades, and it has been developed in response to the increasing biological demand and the reducing worker quantity. Their problem has been worked on by both computer scientists and biologists. This review looks at forty-four research on the subject, attempting to provide a comprehensive picture of the scientific evidence and how the issue was treated. Opinions are taken on the image-taking process, feature extraction, classification strategies, and datasets used for testing.

The technique and effectiveness of catching tomato whitefly and its predatory bugs using yellow sticky traps was given by Ard Nieuwenhuizen et al. [13]. These devices are photographed using a digital single-lens reflex camera and a smartphone camera in both controlled and natural lighting settings. The procedure includes the following actions. The first step is to manually annotate and subset the images. Step two involves teaching a convolutional neural network to use deep learning. The third stage involves sorting the photos into categories. The last stage is a check against manual counts of insects. When applied to the detection of insects, deep learning achieved an average accuracy of 87.4 percent. Insects in smartphone photographs were tallied by both humans and deep learning systems, with a correlation of above 0.95. The techniques used demonstrate that the offered data may be transferred from controlled settings required for training data utilized in smartphone imaging.

Conventional detection and eye observation approaches, as noted by Vivek Tiwari et al. [14], are ineffective for big crops. They presented mask RCNN, a theoretically versatile and broad framework for identifying, localizing, and masking crop-damaging insects. It combines the processes of object detection with instance segmentation, which entails assigning each detected object's pixel to one of many predetermined classes. The main goal of mask RCNN is to generate a mask that may be used to identify the features and locations inside an image. Insect detection and mask generation are performed with great efficacy. This shortens the time it takes to find insects and requires less human involvement. Proposed model is now trained with four insects, but may be readily expanded to categorize additional insects, assisting farmers in accurate insect identification and the use of appropriate pesticides (by research into insect kind and population). This will make crop protection easier for them. Moreover, by incorporating this model into a mobile-based application, it may be made readily available to them.

The Faster R-CNN-based object identification method was used by Yufeng Shen et al. [15] to identify stored-grain insects in field conditions with contaminants. The technique could identify the insects despite their weak adherence. In addition to developing deep convolutional neural networks, an enhanced inception network was created to increase the precision with which microscopic insects could be detected. With an increased mAP of 87.99, the enhanced inception network outperformed both the suggested inception network (81.39 points) and VGG16 (82.66 points). We also reduced the size of the model from 261M to 62M using the SVD method, and the resulting mAP improvement was 3.34.

The compact deep network CPAFNet was developed by Jin Wang et al. [16]. The model achieved a recognition accuracy of 92.63 percent after a 6000-step iterative training experiment, surpassing the performance of the standard deep learning models VggA, Vgg16, Inception V3, and ResNet50. With a training duration that is just 1 hour and sixteen minutes, the CPAFNet model outperforms even the most conventional deep learning models.

In order to classify photos of soybean pests, Everton Castelao Tetila et al. [17] compared the efficacy of the Inception-v3, Resnet-50, VGG-16, VGG-19, and Xception deep learning architectures. The SLIC superpixels technique was investigated for use in an image segmentation phase to separate the insects in the field-collected photos. The performance of deep learning architectures for various fine-tuning and transfer learning techniques was compared to that of more conventional feature extraction and learning methods during the classification challenge. The Resnet-50 architecture, which was trained via fine-tuning computed weights, achieved an accuracy of up to 93.82% in experiments, which was much greater than that of previous machine learning approaches. As shown by the findings, the analyzed designs generalize well in soybean pests datasets and may aid professionals and farmers in keeping pest populations under control.

A fresh open field-image dataset of important (principal and secondary) insect pests of maize plantations, together with a deep learning model (a modified InceptionV3*) for identification of those insects, were presented by Witenberg S. R. Souza et al. [18]. The collection contains 4320 pictures of 6 maize-related insect pests. The suggested model (Inception-V3*) outperformed both the baseline (regular) Inception-V3 (94.8%) and the state-of-the-art (96.3%) AlexNet in terms of average (cross-validated) accuracy. Our long-term goal is to expand our work with this updated version of Inception-V3* to include other insect classes and crop types, despite its similar accuracy to the original model during training. To reduce the use of chemical pesticides and to promote more sustainable agricultural ecosystems, accurate identification of insect pests in field pictures is a pressing need.

Using enhanced VGG19, Denan Xia et al. [19] introduced a target identification approach for fast and accurate insect detection in pictures. The authors of the present study used the pre-trained VGG19 model from the Caffe library to train the optimum model for this research since its design strikes a good compromise between feature extraction and model training. The testing results on the current dataset "MPest" demonstrated the superior speed and accuracy of this approach compared to the state-of-the-art.

Butterfly dataset selection, butterfly data set processing, butterfly identification and classification using Faster R-CNN, and a final average classification accuracy of 70.4% are all addressed in a paper by Ruoyan Zhao et al. [20]. Justify your findings by summarizing the whole experiment. The first step is to guarantee an adequate sample size is used during training. To guarantee efficient training, they may flip the dataset upside down, rotate it 90 degrees to the left or right, provide a cooling effect, inject some noise, and so on; Secondly, they need guarantee the unpredictability of samples while assessing the accuracy. It need frequent parameter tuning and replacement in order to maintain high accuracy in classification and identification. The study of butterfly identification is vital not only to the development of image recognition technology but also to the preservation of butterfly horticulture.

Summary:

Taxonomists often need to identify a specimen to the order or family level before they can assign it a species designation, and identification to the order level is more useful to the general public and younger taxonomists. Therefore, identifying an insect at the order level is a crucial part of the bigger picture of naming insects.

We can employ UAVs with higher-resolution cameras to measure how well a method works with data from a variety of altitudes. Insects in UAV-captured photos may be automatically counted, allowing us to gauge the efficacy of current pest control measures.

We need to think more about the parameter redundancy issue in the network and how to extract the crucial information in the feature map if we want to boost a model's recognition rate.

Automatic feature extraction is a major benefit of deep learning's recent development, giving it an edge over artificial extraction.

3. Proposed Method

Recently, autonomous feature selection has allowed deep learning algorithms to make significant advancements in picture identification. Manually designed features are utilized in conventional image recognition algorithms. These include features from the scale-invariant feature transform and accelerated robust features. Once features have been collected, they are fed into machine learning algorithms to carry out picture identification. The performance of the standard method for picture recognition relies greatly on the integrity of the features that are extracted. Feature extraction, however, is difficult and time-consuming. Redesigning the features is often required when applying them to various issues or uses. We use the term "feature engineering" to describe this process. Expertise and a large number of algorithms are needed for feature engineering, which in this instance is recognizing pests and illnesses. For different pests or crop photos, conventional machine learning methods and image characteristics must be rethought and reselected, respectively. The advent of deep learning in recent years has made it possible to employ CNN for automated feature extraction without the need for hand-crafted features.

Insect images were formerly analyzed only by human brains, but now that computing power has advanced, computers can do the job just as well, if not better. Mathematical morphology-based and threshold-based segmentation approaches remain the gold standard for insect image segmentation at the current time [1]. The primary process in target identification is the accumulation and analysis of data. Faster-RCNN target detection model identification outcomes are somewhat dependent on the quality and size of the data set used to train the model. Therefore, it is important to amass a big number of digital photographs of insects from their natural habitats. For each insect species, researchers require a large sample size from which to draw conclusions. The input size of a picture including insects is irrelevant to the Faster-RCNN model. Figure 1 shows the results of our Faster-RCNN and ADAM optimizer-based picture segmentation for insect identification.

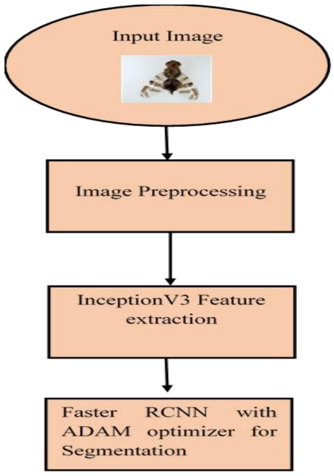

Figure1. Proposed Model Block Diagram

Dataset

Thanks to "Feng Wu and Yueying Li" for their dataset, which consists mostly of Chilo suppressalis,Cicadellidae, and Coleoptera, and which was used to inform the proposed study. We also included information from a Kaggle dataset called "Dangerous farm insect dataset" that includes the Africanized Honey Bees (Killer Bees), Armyworms, Brown Marmorated Stink Bugs, and Colorado Potato Beetles that we caught and released. We only utilized 187 images total, however the dataset had more than 90 images for each category. The information has been divided into a 70% training data and 30% test data set.

Data Preprocessing

Data collected in part is labeled one, and the remaining data must be labeled using the Matlab toolbox for images.

In this stage, data is preprocessed by picking appropriate photos for training and testing and resizing them to a uniform 200x200. Providing data variations and making sure each class has an equal number of examples is essential for training a deep learning model that can generalize.

Train and Test split: Train and test sets are created from the collected data [13]. It is important to keep the training set the same so as to not favor one bug over another in the final product. Split ratio training: test: 70 : 30.

InceptionV3 Feature extraction

Feature extraction is crucial to the final identification outcome. Features extracted from photos should be meaningful in terms of taxonomy and should be easily attainable.

The Inception challenges the conventional wisdom of Convolutional neural networks, which held that adding more Convolutional layers to the network's depth would lead to better results. Learning to approximate the best local sparse nodes with dense components is key to the Inception architecture. When compared to other Convolutional neural networks, Inception stands out due to the fact that the network designer is not responsible for choosing the Convolutional layer's filter type, as well as the layer's placement and formation. The network model can decide whether the Convolutional layer is necessary, which filters to apply, and what values to add. Join all the results together, and the network will figure out how to combine model filters and what learning parameters to use. The InceptionV3 network model has three distinct inception modules, each of which utilizes a unique set of convolution kernels and receptive field sizes to extract picture characteristics at various scales, before combining all of the recovered features into a single one. Simultaneously, a 1 x 1 convolution kernel is implemented to accomplish the dimensionality reduction effect. By applying 1 x 1 convolutions to the big input layer, we may create a more manageable bottleneck layer. Without compromising on network speed, it may drastically cut down on the size of the presentation layer, which in turn reduces computational expenses [7].

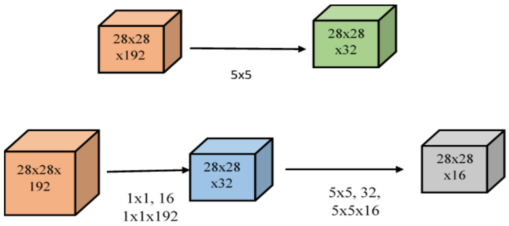

Figure2. InceptionV3

When comparing the computational efficiency of VGGNet and other Inception Networks, it has been shown that Inception Networks like GoogleNet and Inception v1 perform better. It is crucial that computational benefits of an Inception network be preserved. The vulnerability of the resulting network manifests itself whenever an Inception network is repurposed for a new application. For the purpose of loosening restrictions and making model adaption easier, many methods were suggested for use in the Inception v3 model's optimization of the network [10].

Widening the neural network model (several kernel sizes were used to extract the same feature maps) and increasing its depth are both recommended for producing a high-quality model. The model's Convolutional layers were expanded using the inception structure. The inception structure's primary function was to choose a suitable dense component to substitute for the ideal local sparsity structure and then to replicate this structure in other locations. Using this method, the units with the highest relative correlation would be grouped together to create a new layer, which would then be connected to the one above it. Figure 2 depicts the V3 structure at conception. The 1 x 1 convolution kernels were used before the 3 x 3 and 5 x 5 convolution kernels [14] because they required fewer parameters and ran more quickly.

Faster RCNN with ADAM optimizer for Segmentation

Insect detection requires pinpointing the exact position of insects inside an image and labeling them according to kind. The R-CNN detection algorithm takes a more conventional approach by first pinpointing a region that could contain an object, then transforming its size into the convolution network's input format, and finally determining whether or not the region contains an object and what kind of object it is. Finally, the region containing an object is subjected to additional regression and micro-adjustment to improve its framing accuracy. A unique layer, the ROI layer, is proposed by Fast R-CNN. Since R-CNN and Fast R-CNN have been thrown out, a new method called Faster R-CNN has been presented. Its primary contributions are twofold: first, it presents a regional recommendation network (RPN) to rapidly create candidate areas; second, it facilitates parameter sharing between the RPN and the Fast R-CNN network by means of alternating training. Faster R-CNN's structure incorporates feature extraction, regression, and classification into a single network, which boosts performance across the board, particularly in terms of how quickly classifications can be made. Insect segmentation using Faster-RCNN. Figure 2 below outlines Faster R-CNN's fundamental structure.

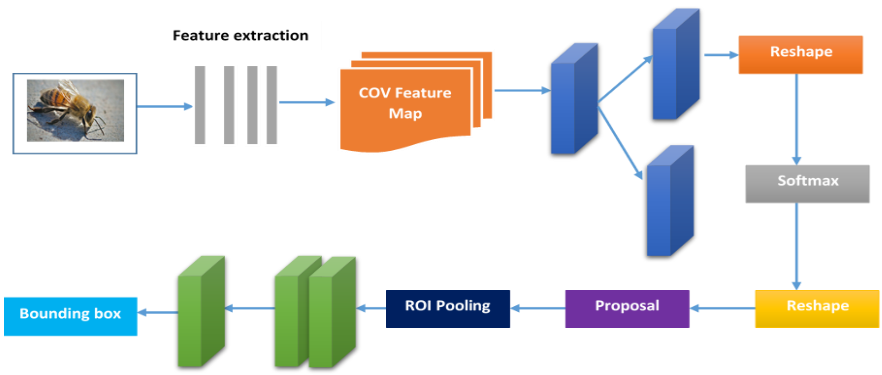

Figure3. Faster R-CNN

Faster RCNN contains two parts: Specifically, Fast RCNN and the Regional Proposal Network (RPN). RPN's goal is to identify potential target locations in an input picture. Fast RCNN is used to categorize the candidate areas created, and to refine the candidate region borders. Features collected by the Convolutional neural network are shared across these two networks and then sent into the RPN for further processing. To construct a feature map of size w h, RPN employs a sliding window technique to gather k starting areas at each pixel point. There are a total of 4k outputs from the regression layer, each of which encodes the 4k coordinates of the k frame, and a total of 2k scores from the classification layer, each of which predicts the chance that a given area is either a target or a background. We first utilize a radial basis function (RPF) network to locate potential pest anchors, and then we use a boundary regression approach to precisely identify suitable locations. To classify these candidate areas, we then send them to the Fast RCNN [23].

Since training in deep learning often takes a significant amount of time and computational resources, optimizing the training process is a common topic of discussion. Adam (adaptive momentum) is a method that may increase learning speed and effect by using less resources while hastening model convergence. Adam is an alternative to the standard stochastic gradient descent optimization procedure. Adadelta incorporates a moment of second-order moment estimate based on momentum's first-order moment estimation, allowing for the dynamic adjustment of the learning rate of each parameter. The use of a bias adjustment helps keep the parameters steady [24].

In order to get the optimal weights for each stage, the Adam algorithm is utilized. This algorithm is of the first order and runs quickly on the computer. The accurate bias ratings of the square gradient and the moving average gradient are reflected. Adam's performance suffers from two issues. Both the effect of updates on the network's overall matching function and the number of its parameters depend on the stochastic gradient discovered during the time of historical gradients. We suggested a modified version of the Adam algorithm as a means of overcoming such challenges. Acute local lows that don't generalize well result from the training-induced drop in weight carriers' real learning rates. Each input weight vector, rather than each individual weight, is accelerated in its learning rate to preserve the gradient's orientation. The updated Adam algorithm retains its original qualities, such as the ability to precisely regulate the learning rate and the conservation of gradient direction for each weight. In addition, it boosts efficiency [25].

Adam algorithm is an adaptive method for optimizing the learning rate. This optimization method was developed specifically for deep learning techniques. The algorithm's primary contribution is the identification of unique adaptive learning rates for different parameters. The algorithm's adaptive moment estimation process inspired the name. Each weight in a deep neural network's activation function has its learning rate adjusted based on estimates of its gradient made using the network's first and second moments. Average and Variance are the first and second moments, respectively. In every iteration, the Adam method employs exponential moving averages to estimate moments in each batch [26]. Following are some mathematical formulae that may be understood in light of the update rule for the Adam optimizer:

$$nt = \beta 1 n(t-1) + (1-\beta 1) gt \quad (1)$$

$$st = \beta 2 s(t-1) + (1-\beta 2) g^2 t \quad (2)$$

Where n and s are weighted moving averages, g is the gradient on the current batch, t is the number of iterations, and β1 and β2 are algorithmic hyper-parameters. The gradient and the squared gradient are moving averages, respectively, in Equation (1) and Equation (2).

Algorithm1. Faster RCNN with Adam optimizer

1. Start
2. Select an image file to load.
3. Use preprocessing for feature extraction.
4. Use InceptionV3 for feature extraction
5. Locate possible object-containing zones in the image. Proposed regions are what we name them here.
6. Feature extraction for CNN using proposed regions.
7. In Faster-RCNN, the search selection is replaced by a recommendation network (RPN) for candidate regions.
8. When it comes to gathering candidate boxes, computing their characteristic maps, and moving them on to the next network, ROI pooling is in charge.
9. Classify the objects using the extracted features.
10. Use the train Fast RCNN function to train a Fast R-CNN object detection
11. returns faster RCNN object detected
12. Initially, both nt and st are set to 0.
13. Both tend to be more biased towards 0 as β1 and β2 are equal to 1.
14. By computing bias-corrected nt and st this problem is corrected by the Adam optimizer.
15. while w(t) not converged do
16. t=t+1
17. use equation 1 and equation 2 to get nt and st
18. then wt=w(t-1)-(nt/st)
19. end
20. return wt

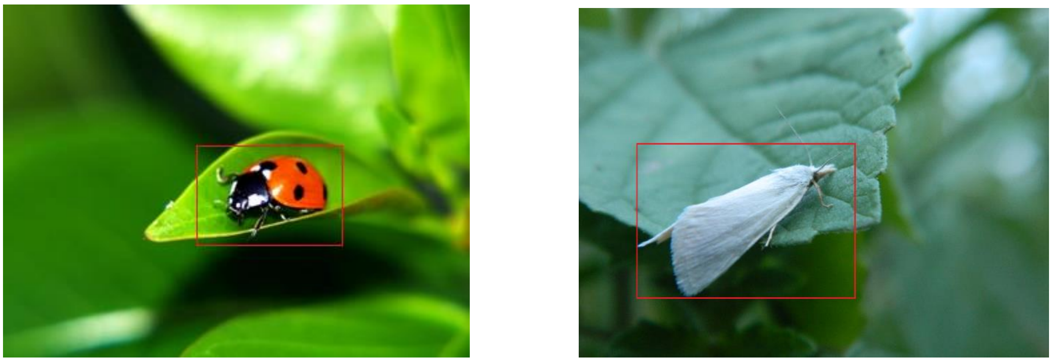

Figure 4. Detecting insect image with higher accuracy using proposed FRCNN with Adam optimizer

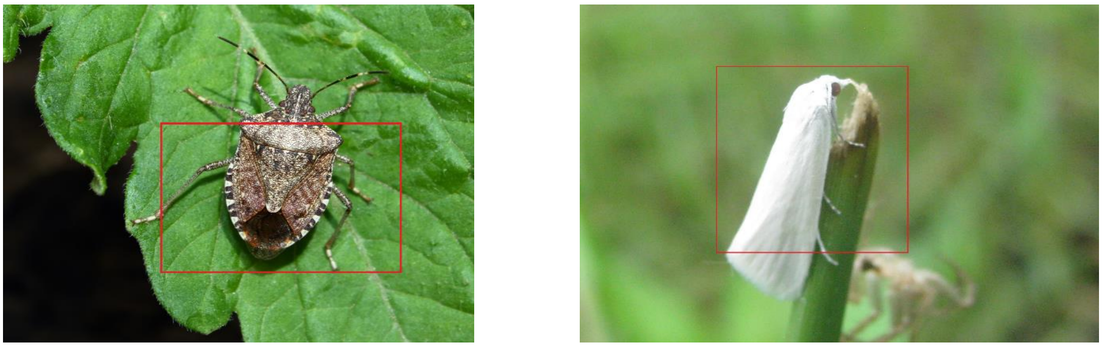

Figure 5. Detecting insect image with moderate accuracyAdam optimizer

Figure 6. Detecting insect image with poor Accuracy with general FRCNN

A total of 320 image frames were utilized in the training process, with a 70% train:30% test split. To achieve a 95% success rate, we started with a learning rate of 0.05. Training continued until the typical drop in performance was almost continuous. Several loss functions were considered for this analysis: the RPN objectness localization and classification loss, the RPN localization loss, the total loss, and the clone loss. After training the model, we put it through its paces on a set of 80 test images. Each anticipated location was given a detection score to indicate how confident we were in our prediction [22].

4. Results

The suggested approach for insect detection was tested using 56 images from the study. With 131 images used for training and the corresponding predictions for the confusion matrix and validation parameters provided below.

Table I. Detected and non Detected image count

Insect categories	Detected images		Non Detected images	
	FRCNN+ADAM	FRCNN	FRCNN+ADAM	FRCNN
Chilo suppressalis	21	16	7	12
Cicadellidae	17	12	8	13
Coleoptera	38	31	12	19
Africanized Honey Bees (Killer Bees)	35	28	13	20
Armyworms	8	6	4	6
Brown Marmorated Stink Bugs	9	6	2	5
Colorado Potato Beetles	7	5	5	8

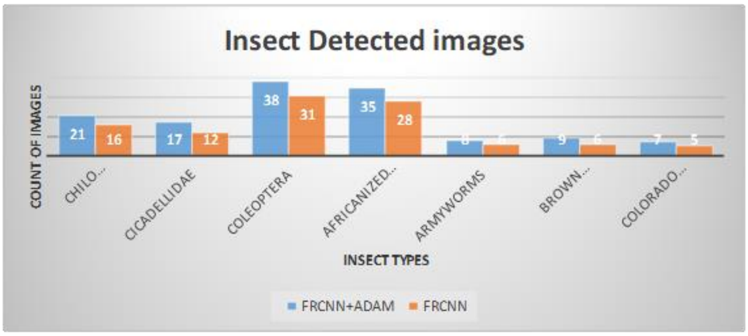

Figure7. No of images detected by FRCNN-ADAM and FRCNN

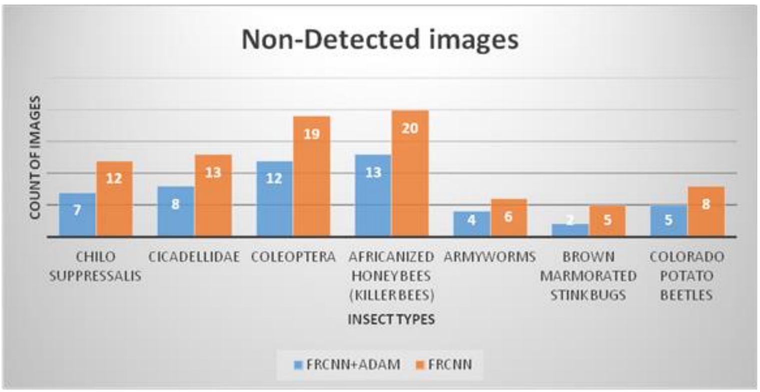

Figure7. No of images non-detected by FRCNN-ADAM and FRCNN

Figure 7 and Table I show the number of correct and incorrect insect detections made by the proposed FRCNN after being optimized with the ADAM optimizer and having its features extracted using the inception v3 method. This model is then compared to the standard FRCNN, and the results show that the improved model is more accurate at making insect detections. With a lesser ability to avoid detection.

Table II. Accuracy and Error of detection

Insect categories	Accuracy of detection		Error detection	
	FRCNN+ ADAM	FRCNN	FRCNN+ADAM	FRCNN
Chilo suppressalis	0.75	0.57	0.28	0.42
Cicadellidae	0.68	0.48	0.32	0.52
Coleoptera	0.76	0.62	0.24	0.38
Africanized Honey Bees (Killer Bees)	0.72	0.58	0.27	0.42
Armyworms	0.66	0.5	0.33	0.5
Brown Marmorated Stink Bugs	0.81	0.54	0.18	0.45
Colorado Potato Beetles	0.54	0.38	0.38	0.62

Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 5s (2025)

Table III. Area comparison of detected data

Insect categories	Sample size	Target Size	
		FRCNN+ADAM	FRCNN
Chilo suppressalis	805	1246	1533
Cicadellidae	621	1526	1825
Coleoptera	1253	2056	2630
Africanized Honey Bees (Killer Bees)	945	1463	1924
Armyworms	1543	2543	2854
Brown Marmorated Stink Bugs	1026	1254	1869
Colorado Potato Beetles	756	906	1325

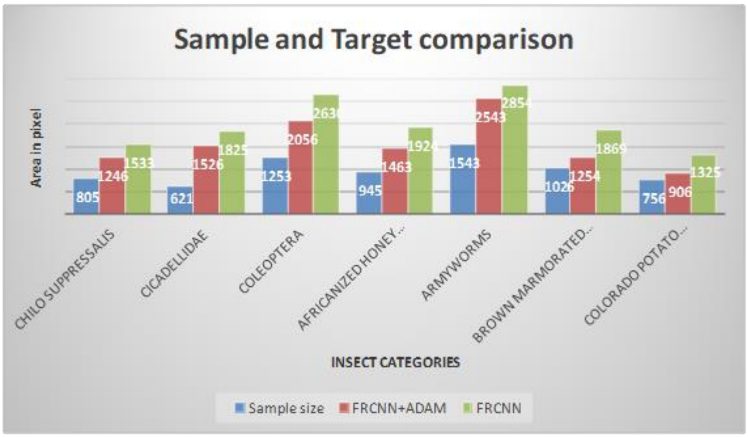

Figure8. Sample space and target space comparison of proposed and existing model

Figure 8 and table III above show the expected target picture and the example image with insect labels. The number of pixels is shown inside the border marker's bounds, which represents the region where the bug was found. Additionally, the labeling feature in the toolbox allows for sample marking, and the chart demonstrates that the suggested model has spotted areas closer to the sample.

5. Conclusion

Interest in automated insect identification is growing in many areas of study, including entomological science, environmental science, and agricultural engineering. A number of contributors discussed the many influences on automated insect identification studies. Low detection accuracy and high processing time are addressed by using the faster-RCNN model for insect target identification. In this study, we provide a novel method for identifying insects using picture segmentation. Insect images were segmented using Faster RCNN using the ADAM optimizer, and features were extracted using InceptionV3. According to the findings, our suggested model outperformed the state-of-the-art method.

Effective Deep Learning methods may be used to construct a future-proof categorization system and expand the available dataset.

Refrences

[1] Wu, F., & Li, Y. (2023). Lightweight Field Insect Recognition and Classification Model Based on Improved Deep Learning under Complex Background. Security and Communication Networks, 2023.
[2] Rong, M., Wang, Z., Ban, B., & Guo, X. (2022). Pest identification and counting of yellow plate in field based on improved mask r-cnn. Discrete Dynamics in Nature and Society, 2022, 1-9.
[3] Wang, J., Lin, C., Ji, L., & Liang, A. (2012). A new automatic identification system of insect images at the order level. Knowledge-Based Systems, 33, 102-110.
[4] Du, Y., Liu, Y., & Li, N. (2020, April). Insect detection research in natural environment based on faster-r-cnn model. In Proceedings of the 2020 5th International Conference on Mathematics and Artificial Intelligence (pp. 182-186).
[5] Zhu, C., Wang, J., Liu, H., & Mi, H. (2018). Insect identification and counting in stored grain: image processing approach and application embedded in smartphones. Mobile Information Systems, 2018, 1-5.
[6] Yang, H., Liu, W., Xing, K., Qiao, J., Wang, X., Gao, L., & Shen, Z. (2010, August). Research on insect identification based on pattern recognition technology. In 2010 Sixth International Conference on Natural Computation (Vol. 2, pp. 545-548). IEEE.
[7] Júnior, T. D. C., & Rieder, R. (2020). Automatic identification of insects from digital images: A survey. Computers and Electronics in Agriculture, 178, 105784.
[8] Dongmei, Z., Ke, W., Hongbo, G., Peng, W., Chao, W., & Shaofeng, P. (2020, November). Classification and identification of citrus pests based on InceptionV3 convolutional neural network and migration learning. In 2020 International Conference on Internet of Things and Intelligent Applications (ITIA) (pp. 1-7). IEEE.
[9] Hansen, O. L., Svenning, J. C., Olsen, K., Dupont, S., Garner, B. H., Iosifidis, A., ... & Høye, T. T. (2020). Species-level image classification with convolutional neural network enables insect identification from habitus images. Ecology and evolution, 10(2), 737-747.
[10] Wen, C., & Guyer, D. (2012). Image-based orchard insect automated identification and classification method. Computers and electronics in agriculture, 89, 110-115.
[11] Cleetus, L., Raji Sukumar, A., & Hemalatha, N. (2021). Computational prediction of disease detection and insect identification using xception model. bioRxiv, 2021-08.
[12] Martineau, M., Conte, D., Raveaux, R., Arnault, I., Munier, D., & Venturini, G. (2017). A survey on image-based insect classification. Pattern Recognition, 65, 273-284.
[13] Nieuwenhuizen, A. T., Hemming, J., & Suh, H. K. (2018). Detection and classification of insects on stick-traps in a tomato crop using Faster R-CNN.
[14] Tiwari, V., Gupta, S., Roy, P., Karda, C., Agrawal, S., Rathore, M. S., ... & Pal, A. (2022). Soybean crop non-beneficial insect identification using mask RCNN. In Information and Communication Technology for Competitive Strategies (ICTCS 2020) ICT: Applications and Social Interfaces (pp. 301-311). Springer Singapore.
[15] Shen, Y., Zhou, H., Li, J., Jian, F., & Jayas, D. S. (2018). Detection of stored-grain insects using deep learning. Computers and Electronics in Agriculture, 145, 319-325.
[16] Wang, J., Li, Y., Feng, H., Ren, L., Du, X., & Wu, J. (2020). Common pests image recognition based on deep convolutional neural network. Computers and Electronics in Agriculture, 179, 105834.
[17] Tetila, E. C., Machado, B. B., Astolfi, G., de Souza Belete, N. A., Amorim, W. P., Roel, A. R., & Pistori, H. (2020). Detection and classification of soybean pests using deep learning with UAV images. Computers and Electronics in Agriculture, 179, 105836.
[18] Souza, W. S., Alves, A. N., & Borges, D. L. (2019, October). A deep learning model for recognition of pest insects in maize plantations. In 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC) (pp. 2285-2290). IEEE.
[19] Xia, D., Chen, P., Wang, B., Zhang, J., & Xie, C. (2018). Insect detection and classification based on an improved convolutional neural network. Sensors, 18(12), 4169.
[20] Zhao, R., Li, C., Ye, S., & Fang, X. (2019, March). Butterfly recognition based on faster R-CNN. In Journal of Physics: Conference Series (Vol. 1176, No. 3, p. 032048). IOP Publishing.
[21] Lin, T. L., Chang, H. Y., & Chen, K. H. (2020). The pest and disease identification in the growth of sweet peppers using faster R-CNN and mask R-CNN. Journal of Internet Technology, 21(2), 605-614.

[22] Mohd-Isa, W. N., Nizam, A., & Ali, A. (2019, July). Image segmentation of meliponine bee using faster r-cnn. In 2019 Third World Conference on Smart Trends in Systems Security and Sustainablity (WorldS4) (pp. 235-238). IEEE.
[23] Zhang, M., Chen, Y., Zhang, B., Pang, K., & Lv, B. (2020). Recognition of pest based on faster rcnn. In Signal and Information Processing, Networking and Computers: Proceedings of the 6th International Conference on Signal and Information Processing, Networking and Computers (ICSINC) (pp. 62-69). Springer Singapore.
[24] Jiang, X., Hu, B., Chandra Satapathy, S., Wang, S. H., & Zhang, Y. D. (2020). Fingerspelling identification for Chinese sign language via AlexNet-based transfer learning and Adam optimizer. Scientific Programming, 2020, 1-13.
[25] Chandriah, K. K., & Naraganahalli, R. V. (2021). RNN/LSTM with modified Adam optimizer in deep learning approach for automobile spare parts demand forecasting. Multimedia Tools and Applications, 80(17), 26145-26159.
[26] ŞEN, S. Y., & ÖZKURT, N. (2020, October). Convolutional neural network hyperparameter tuning with adam optimizer for ECG classification. In 2020 innovations in intelligent systems and applications conference (ASYU) (pp. 1-6). IEEE.