An Efficient System for Real-time Mobile Smart Device-based Insect Detection

Thanh-Nghi Doan
Faculty of Information Technology, An Giang University
Vietnam National University Ho Chi Minh City, An Giang, Vietnam

***Abstract*—In recent years, the rapid development of many pests and diseases has caused heavy damage to the agricultural production of many countries. However, it is difficult for farmers to accurately identify each type of insect pest, and yet they have used a large number of pesticides indiscriminately, causing serious environmental pollution. Meanwhile, spraying pesticides is very expensive, and thus developing a system to identify crop-damaging pests early will help farmers save a lot of money while also contributing to the development of sustainable agriculture. This paper presents a new efficient deep learning system for real-time insect image recognition on mobile devices. Our system achieved an accuracy of mAP@0.5 with the YOLOv5-S model of 70.5% on the 10 insect dataset and 42.9% on the IP102 large-scale insect dataset. In addition, our system can provide more information to farmers about insects such as biological characteristics, distribution, morphology, and pest control measures. From there, farmers can take appropriate measures to prevent pests and diseases, helping reduce production costs and protecting the environment.**

Keywords—Deep learning; real-time insect pest detection; YOLOv5; mobile devices

I. Introduction

Climate change has caused pests to multiply, grow quickly, and cause significant damage to the world's agricultural economy [1]. Pests are estimated to cost up to 40% of worldwide agricultural output each year, according to the Food and Agriculture Organization. At present, plant diseases cost the global economy almost $220 billion each year, while invading insects cost at least $70 billion [2]. Therefore, farmers in many countries have used a large number of different pesticides to protect crops and ensure the quality of agricultural products. However, due to a lack of specialized knowledge, many farmers have difficulty detecting and correctly identifying pests and diseases that cause crop damage. As a result, most farmers did not have reasonable pest control measures, including the indiscriminate and improper use of a large number of pesticides on a large scale. This not only increases production costs but also seriously pollutes the environment, destroys beneficial insects, disrupts ecosystem balance, and damages the health and living environment of humans and many other species. As a result, it is critical to research information technology systems in order to accurately, efficiently, quickly, and conveniently identify pests and diseases that harm crops. This system will aid in the resolution of the aforementioned issues, thereby contributing significantly to long-term agricultural development. Such a system must be designed for real-time identification, be simple to install and use, and be appropriate for farmers' level of knowledge and actual working conditions, where each farmer typically has a smartphone with a basic configuration. Therefore, an automatic system to identify pests on plants using inexpensive smart phones must be developed and deployed. The primary goal is to efficiently detect insects in real-time manner, providing farmers with greater convenience and mobility in early pest treatment. Although smartphones have penetrated a variety of industries, including manufacturing, medicine, and health care, use of mobile devices in agriculture has been slower. Farmers understand the need for mobile agriculture as technology advances, which not only allow farmers to execute agricultural activities more effectively using their phones, but also transform arable farming into smart agriculture. In this research, a real-time insect object detection system is built in the context of large-scale insect pest datasets. Our system is based on the YOLOv5-S model and has been integrated onto mobile devices with limited hardware configurations, making it ideal for farmers in the field.

II. Background Study

Much of the prior research has presented real-time image-based recognition systems for mobile devices based on various CNN architectures. To recognize leaves from images, the authors of [3] have developed a novel extraction and classification technique. The insect population and illness regions in the segmented images are then calculated using a region-labeling technique. A mathematical morphological algorithm is utilized to separate the items in the zones of adhesion. The proposed solution is tested in the field and deployed on mobile smart devices. The experimental findings reveal that the suggested technique has high efficiency and strong recognition performance. The authors of [4] have created a pest infestation early warning system for paddy farming that includes an Android application and a web-based application. The Agriculture Department will use the technology to identify insect infestations, locate them, and alert the early warning system. The technology will be able to enter the farmers' infestation data into databases. The data will be utilized by the agronomist to assess the paddy plot's risk in four stages. The number of pests, kind of pest, location, and present circumstances will be used to classify each stage. After the agronomist has completed their review, the system will send an email to the farmers informing them of the quality of their current paddy plot. The researchers from [5] suggested an image processing technique and a smartphone application to recognize and count insects. The nonuniform brightness of insect images obtained with mobile phones is released using a sliding window-based binarization, and then connected domain-based histogram statistics are utilized to identify and count the insects in stored grain. Finally, testing using an Android application shows that the proposed technique can count random bug photographs from mobile phones with 95% accuracy, which is superior to the previous method. In [6], MAESTRO, a novel grasshopper identification framework that employs deep learning to recognize insects in RBG pictures, is demonstrated. MAESTRO uses a state-of-the-art two-stage deep learning training approach. The framework may be used on cellphones as well as desktop PCs. The authors of [7] offer an AI-based pest detection system that addresses the challenge of identifying scale pests using photos. Scale pests are detected and localized in the image using deep-learning-based object identification models such as faster region-based convolutional networks, single-shot multibox detectors, and YOLOv4. Among the algorithms, YOLOv4 had the highest classification accuracy, with 100% in mealybugs, 89% in Coccidae, and 97% in Diaspididae. A smartphone application based on the trained scale insect detection model has been developed to assist farmers in identifying pests and administering appropriate pesticides to reduce crop losses. The researchers at [8] have studied the best machine learning approach for developing a pest detection model for mobile information systems. The article [9] proposed a novel smartphone application that uses a deep-learning method to automatically categorize pests for the benefit of professionals and farmers. Faster R-CNN is used in the created application to do insect pest recognition using cloud computing. To assist farmers, a database of suggested pesticides is linked to the reported crop pests. This research has been validated for five distinct pest species. The suggested Faster R-CNN had the greatest accuracy in identification rate of 99% for all pest images analyzed. The study [10] provided a novel method for establishing the use of hand-held image capture of insect traps for pest detection in vineyards by embedding artificial intelligence into mobile devices. Their solution integrates many computer vision technologies to enhance numerous areas of picture quality and appropriateness. The extensive review [11] examines deep learning framework methodologies and applications in smart pest monitoring, with a focus on insect pest categorization and detection using field photos. The methodology and technical information created in insect pest classification and detection using deep learning are consolidated and distilled during multiple processing stages: picture collection, data preprocessing, and modeling strategies. Finally, a generic framework for smart insect monitoring is proposed, and future challenges and trends are discussed. In AlertTrap [12], SSD architecture implementation with different cutting-edge backbone feature extractors, such as MobileNetV1 and MobileNetV2, appears to be a viable solution to the real-time detection problem. SSD-MobileNetV1 and SSD-MobileNetV2 work well, with AP@0.5 rates of 0.957 and 1.0, respectively. YOLOv4-tiny surpasses the SSD family in AP@0.5 with 1.0; nevertheless, its throughput velocity is significantly slower, indicating that SSD models are better candidates for real-time implementation. They also ran the models via synthetic test sets that simulated predicted environmental disruptions. The YOLOv4-tiny tolerated these disruptions better than the SSD variants. By combining EfficientNet [13] and Power mean SVM [14], the authors of the research [15] published the state of the art on insect image classification on the large-scale IP102 dataset with an accuracy of up to 71.84%. However, the abovementioned systems still have some limitations, such as the small number of pest identifications; the accuracy is not high; the equipment configuration requirements are high; and it is difficult to deploy in practice. They lack aspects such as geolocation recoding of recognized harmful pests, information about identified dangerous pests, and robust distributed mobile information frameworks. Currently, there is no real-time existing identification system for mobile devices. Therefore, this paper proposes a new real-time insect identification system with reasonable cost, efficiency, easy installation, and practical deployment on mobile devices with limited hardware configuration. Furthermore, this study also looks at lightweight network models and embedded terminal realizations, both of which are increasingly relevant and promising. The paper's main contributions are as follows:

- A novel real-time insect identification system that is ideal for mobile devices with restricted hardware configuration, easy to install, inexpensive, and user-friendly.
- The most current identification results using YOLOv5-S from the large-scale dataset IP102 are presented.
- A new system captures images and uses GPS to determine the distribution of insects in the field. This contributes to the development of a large insect database and insect distribution maps.

The rest of the article is arranged as follows. Section III describes the materials and methods used to evaluate our approach, including an overview of our system, the YOLOv5 model, and the pest insect image datasets. The experimental results and discussion are reported in Section IV. Section V presents the conclusions, limitations, and recommendations for future research.

III. Materials and Methods

A. Overview of our System

An overview of our real-time insect identification system is shown in Fig. 1. Users can first use their mobile phones to photograph insects in a real-time manner, or they can use insect photographs found on the internet or images captured by bug traps. The YOLOv5-S model, which is already embedded into the mobile application, then identifies the insect image in real time, resulting in a very quick insect identification time. When an insect image is properly identified, the system will provide the user with detailed information on the insect, such as its name, biological characteristics, distribution, morphology, and control strategies. Our new insect recognition system can work in both online and offline mode. In the online mode, the insect identification information is sent to the Web server, which then processes and returns detailed insect information in JSON format [16]. Insect information can be viewed alongside similar images in the data warehouse. The user can also see a list of all insects, complete with detailed information and images. Users can upload insect images and shooting locations to update the data warehouse at the same time in this mode.

The entire database will be stored on the server in the online mode, making it suitable for mobile devices with limited hardware configuration and ensuring that information is always up-to-date. The application's speed, however, is determined by the available network bandwidth. In the offline mode, SQLite [17], a C-language package that creates a compact, fast, self-contained, high-reliability, full-featured SQL database engine, is used for storing insect information data on mobile devices. This mode will be very useful in cases where farmers' working environments do not have internet, such as in the fields far from urban areas, where internet, 4G, and 5G coverage are not yet available. However, in this mode, some application functions will be restricted.

B. YOLOV5

YOLOv5 [18] is a single-stage object detection system. In one-stage object identification approaches, object detection is considered as a regression issue. It estimates the class probability and the coordinates of the bounding box that will contain the object in a single step on the input picture. The backbone, neck, and head are the three main components. YOLO is another name for the head layer. The model backbone's duty is to draw attention to the image's unique features. In YOLOv5, the model backbone is a CSPNet [19] structure. The CSPNet approach divides the feature map in the base layer into two parts; some reach the transition layer through the dense block, while the other half is directly integrated with the transition layer. This not only reduces model size but also increases inference speed [20]. In this study, the YOLOv5-S model is used to develop applications on mobile devices due to its small size and model parameters, GFLOPs calculation speed and high accuracy, and lack of requirement for high hardware configuration when compared to other YOLO models such as YOLOv4 [21], YOLOX [22]. As shown in Table I, the YOLOv5-S model is relatively small in size, with a network parameter of 7.3M and a disk size of 14.2 MB, making it suitable for mobile devices with limited hardware configuration. With a GFLOPs index of 17.1, the calculating speed of the YOLOv5-S is adequate. Furthermore, when compared to other YOLO models, the indicators of mAP^{val}@0.5 and the speed of the YOLOv5-S model in Table IV and Table V are quite excellent.

TABLE I. NETWORK PARAMETERS OF YOLO MODELS

Models	Params (M)	Size on disk (MB)	GFLOPs
YOLOv4	27.6	245.0	59.6
YOLOv4-tiny	5.88	23.1	6.8
YOLOv5-S	7.2	14.2	17.1
YOLOv5-M	21.2	40.8	51.4
YOLOv5-L	46.5	89.3	115.6
YOLOv5-X	86.7	167.1	219.0
YOLOX-S	9.0	68.5	26.8
YOLOX-M	25.3	193.0	73.8
YOLOX-L	54.2	413.0	155.6
YOLOX-X	99.1	757.0	281.9

C. Datasets

To create the insect pest database for machine learning models, 2,335 photos of 10 distinct pest kinds were collected from internet data sources, as shown in Fig. 2. The dataset was then split into the following proportions: 70% of the samples were utilized for training, 20% for model evaluation, and the remainder for testing. As a consequence, the result dataset has 1634 images for training, 467 images for validation, and 234 images for testing, as shown in Table II. The LabelImg program [23] is utilized to manually label the insect objects and generate the .xml file containing object position information, which is then transformed into the .txt file that YOLOv5 can read. Because the IP102 data set has some constraints, such as the same class with numerous different insect stages such as larvae, caterpillars, and moths, achieving high identification efficiency is challenging. Therefore, the YOLOv5-S model was tested with 10 insect classes that were gathered by the agriculture expert volunteers.

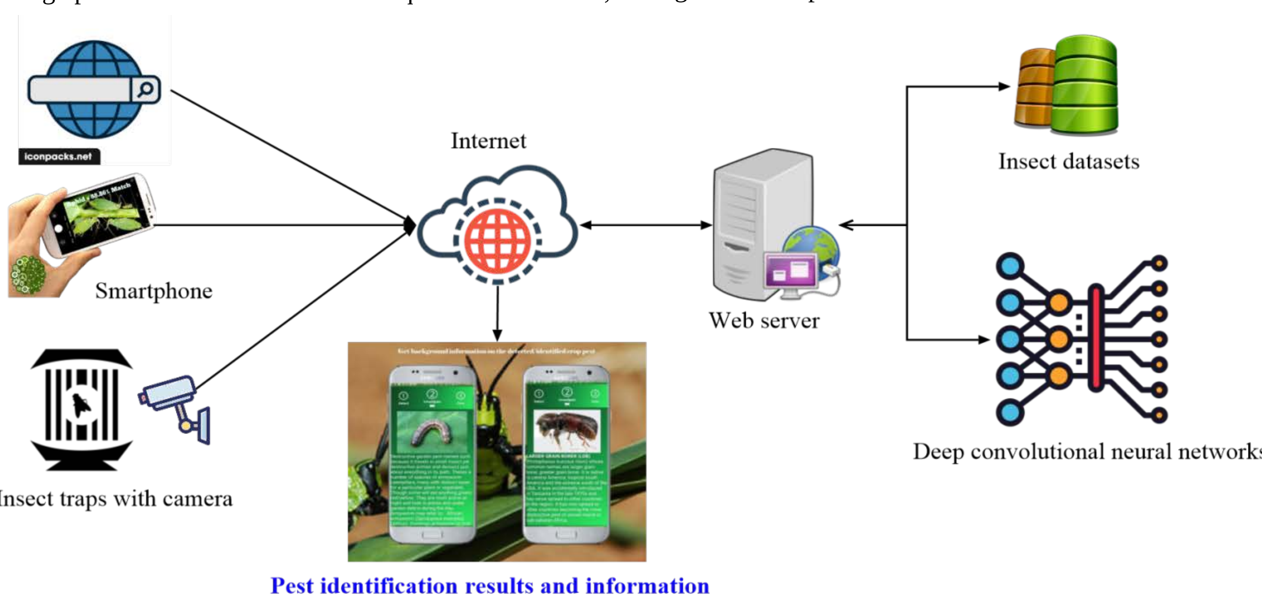


Fig. 1. Overview of our Real-time Insect Image Recognition System by Mobile Devices.

Fig. 2. Some Images of Insect Samples in the Insect10 Dataset.

TABLE II. THE NUMBER OF IMAGES IN THE INSECT10 DATASETS WITH 10 INSECT SPECIES

No	Insect name	Train	Validation	Test
1	Acalymma_vittatum	116	33	17
2	Achatina_fulica	258	74	37
3	Alticini	193	55	28
4	Asparagus_beetles	89	25	13
5	Aulacophora_similis	113	32	16
6	Cerotoma_trifurcata	86	25	12
7	Dermaptera	111	32	16
8	Leptinotarsa_decemlineata	234	67	33
9	Mantodea	185	53	26
10	Squash_bug	249	71	36
	Total	1634	467	234

In this paper, the new system was also evaluated on large-scale insect image datasets. However, collecting a large-scale insect pest image dataset is difficult due to the fact that, depending on the species and kind of insect pest, all insect pests go through several phases during their lifecycle. As a result, the insect pest pictures from the publicly available IP102 dataset [24] are used for evaluating the system. It comprises almost 75,000 photos from 102 agricultural insect pest categories. The IP102 collection includes 75,222 photos and 102 insect pest classifications, while the smallest category comprises just 71 samples. There are 18,983 annotated photos for the job of object detection. As in [24], the images with bounding box annotations were divided into training and testing sets of 15,178 and 3,798 images, respectively. Some sample images of the IP102 dataset are shown in Fig. 3.

Fig. 3. Some Images of Insect Samples in the IP102 Dataset.

IV. RESULT AND DISCUSSION

A. Experimental Setup and Training

All YOLO model training experiments were carried out on Google Colab using a Tesla K80 24 GB GPU. Algorithms are written in the Python and Keras programming languages. To train the models, the experimental setup is as follows: a learning rate of 0.01, an image size of 640 pixels, a batch size of 16, and 150 epochs for YOLOv5, YOLOX, and 2,000 epochs for YOLOv4. The Stochastic Gradient Descent [25] is used as the optimization algorithm. Devices with low configuration are utilized to conduct tests on mobile devices, as indicated in Table III.

TABLE III. SMARTPHONE DEVICE CONFIGURATION AND APPLICATION DEVELOPMENT ENVIRONMENT

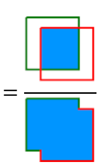
Smartphone hardware configuration	The Samsung Galaxy A30 is powered by a Samsung Exynos 7 Octa 7904 processor with MHZ and 8 cores. The powerful processor and 3000.0 MB of RAM give incredible performance, ensuring trouble-free operation of even the most complex program or game. The Samsung Galaxy A30 uses a microSDXC memory card. The phone carries over the 15.93-megapixel rear camera sensor at the back of the device. The front camera of the Samsung has 15.93. It gives us very high quality photos and videos with a great camera interface. The device has a 6.4-inch SUPER AMOLED display. It gives a decent display quality and a great gradation between warm and cold colors. The OS is Android 10.
Programinng language to build applications	Programing language: Java, Development Environment: Android Studio
The light	Normal luster intensity

B. Evaluation Metrics

Mean Average Precision (mAP) is a popular metric for assessing the performance of object detecting systems. The mAP computes a score by comparing the ground-truth bounding box to the detected box. The higher the score, the more precise is the model's detections. The mAP formula is based on the following sub metrics: Confusion Matrix, Intersection over Union (IoU), Recall, Precision. To create a confusion matrix, the experiments present four attributes: True Positives (TP): The model predicted a label and matched it correctly as per ground truth. True Negatives (TN): The model does not predict the label and is not a part of the ground truth. False Positives (FP): The model predicted a label, but it is not a part of the ground truth. False Negatives (FN): The model does not predict a label, but it is part of the ground truth.

In Equation (1), IoU denotes the overlap of anticipated bounding box coordinates with ground truth box coordinates. It explains how an object identification algorithm creates prediction scores. The definition of IoU is described in Fig. 4. Higher IoU implies that the anticipated bounding box coordinates are similar to the ground truth box coordinates.

$$IOU = \frac{\text{area of overlap}}{\text{area of union}} = \tag{1}$$



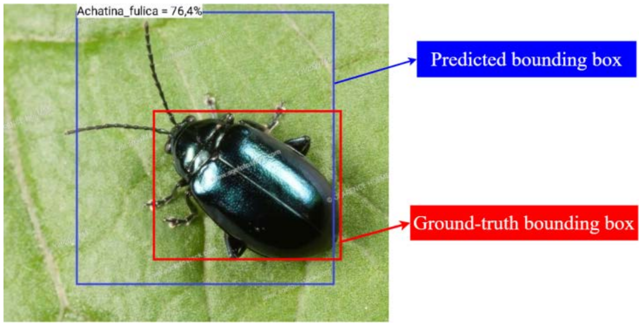


Fig. 4. IoU Definition.

In Equation (2), Precision refers to how successfully you can identify true positives (TP) from all positive predictions. In Equation (3), Recall measures how well you can find true positives (TP) out of all predictions (TP+FN).

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

In Equation (4), Average Precision is calculated as the weighted mean of precision at each threshold; the weight is the increase in recall from the prior threshold. In Equation (5), Mean Average Precision is the average of the AP of each class. However, the interpretation of AP and mAP varies in different contexts. On the validation datasets, the mAP^{val}@0.5 means the average mAP with IoU thresholds over 0.5. The mAP^{val}@0.5:0.95 means average mAP over different IoU thresholds, from 0.5 to 0.95, step 0.05.

$$AP = \sum_{k=0}^{k=n-1}[Recall(k) - Recall(k+1)] * Precision(k) \quad (4)$$

$$n = number\ of\ thresholds$$

$$mAP = \frac{1}{n}\sum_{k=1}^{k=n} AP_k \quad (2)$$

$$AP_k = the\ AP\ of\ class\ k, and\ n = the\ number\ of\ classes$$

C. Experimental Results and Discussion

The experiment was conducted to analyze the backbone of models, input image size, mAP@IoU:0.5 and mAP@IoU:0.5:0.95 metrics as a result of the training. Table IV and Fig. 5 show the results of four different model variations on the Insect10 dataset. On the Insect10 dataset, the numerical results in Fig. 5 demonstrate that the new mobile application has a relatively high success rate in precision, recall and mAP for pest object recognition. For instance, the detection performance of the Alcalymma insect has the lowest mAP@IoU:0.5 identification accuracy of 0.45, while the detection performance of Leptinotarsa has the highest at 0.979. Our application is based on the YOLOv5-S model, which was trained on Insect10 datasets with 10 different insect species. The actual results show that, when compared to other object detection methods, YOLO has a faster recognition speed and can almost identify objects in real-time manner. Fig. 7 shows some examples of successful insect recognition on mobile devices using the Insect10 datasets.

Our approach has also been evaluated on the large-scale dataset IP102 [24] to see how well it scales on these datasets. As shown in Table V and Fig. 6, our system has achieved a promising performance of mAP^{val}@0.5 accuracy of 42.9% with the YOLOv5-S model. This result shows that the new approach outperforms several previous approaches that were reported in [24]. However, insect object detection was still more challenging using the IP102 dataset. The reason is that the insect pests in the image are difficult to detect due to their color appearance and the image backgrounds are very similar. In addition, the morphology of an insect pest issue, such as a moth, can vary substantially as it develops. Fig. 8 depicts some images of successful insect recognition using the IP102 dataset on a mobile device. This indicates that our approach offers several benefits over existing methods, including the ability to handle massive data sets with excellent accuracy. Moreover, this new system may also be implemented on low-cost mobile devices with minimal hardware configuration. In addition, as illustrated in Fig. 9, the usage of matching pesticides is integrated with the pest categorization findings to advise professionals and farmers. In the near future, this system will be implemented on new devices like the NVIDIA Jetson Nano Developer Kit [26], which have a higher hardware configuration, a lower cost, a smaller footprint, and a better level of durability.

TABLE IV. SIMULATION RESULTS OF YOLOV4, YOLOV5, AND YOLOX MODELS ON THE INSECT10 DATASET

Models	**Backbone**	**mAP^{val}@0.5**	**mAP^{val}@0.5:0.95**
YOLOv4	CSPDarknet53	84.9	63.2
YOLOv4-tiny	CSPDarknet53	64.4	48.3
YOLOv5-S	Darknet-53	70.5	35.9
YOLOv5-M	Modified CSP v5	76.6	42.7
YOLOv5-L	Modified CSP v5	78.9	46.8
YOLOv5-X	Modified CSP v5	73.0	40.9
YOLOX-S	Darknet-53	84.8	58.5
YOLOX-M	Modified CSP v5	82.3	61.9
YOLOX-L	Modified CSP v5	84.0	65.0
YOLOX-X	Modified CSP v5	83.0	64.0

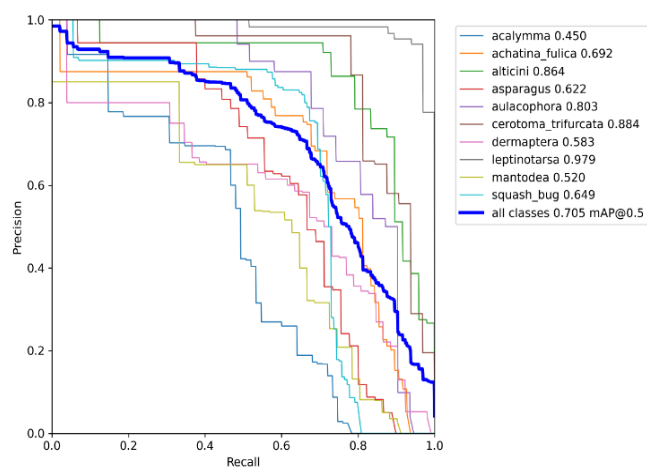


Fig. 5. Precision and Recall of Insect Recognition Results on the Insect10 Dataset using the YOLOv5-S Model.

TABLE V. SIMULATION RESULTS OF YOLOV4, YOLOV5, AND YOLOX MODELS ON THE IP102 DATASETS

Models	**Backbone**	**mAP^{val}@0.5**	**mAP^{val}@0.5:0.95**
YOLOv4	CSPDarknet53	39.2	20.1
YOLOv4-tiny	CSPDarknet53	36.1	19.0
YOLOv5-S	Darknet-53	42.9	24.0
YOLOv5-M	Modified CSP v5	47.4	27.9
YOLOv5-L	Modified CSP v5	50.1	29.9
YOLOv5-X	Modified CSP v5	54.0	32.5
YOLOX-S	Darknet-53	52.3	34.1
YOLOX-M	Modified CSP v5	54.2	35.1
YOLOX-L	Modified CSP v5	53.9	34.7
YOLOX-X	Modified CSP v5	54.1	34.9

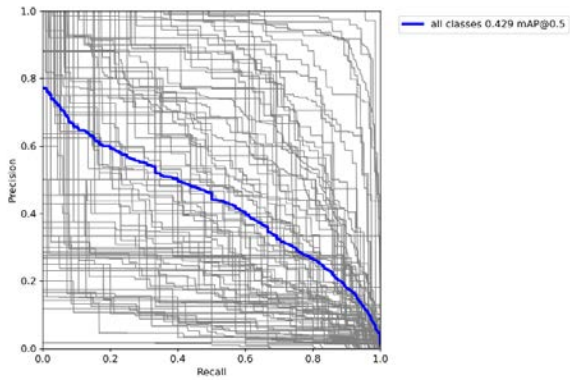


Fig. 6. Precision and Recall of Insect Recognition Results on the IP102 Dataset using the YOLOv5-S Model.

Fig. 7. Some Images were Successfully Detected on Mobile Devices using the Insect10 Dataset.

Fig. 8. Some Images were Successfully Detected on Mobile Devices using the IP102 Dataset.

The information on insect GPS location and density will be extremely useful for several Integrated Pest Management systems. Therefore, our systems are designed to allow users to automatically record this information. Then, a real-time insect distribution density map is created using this data, as illustrated in Fig. 10. This map will assist expert users in tracking and forecasting the density and evolution of insect infections over large areas. At the same time, it is possible to evaluate the potential effects of insect pests on agriculture and ecosystem production.

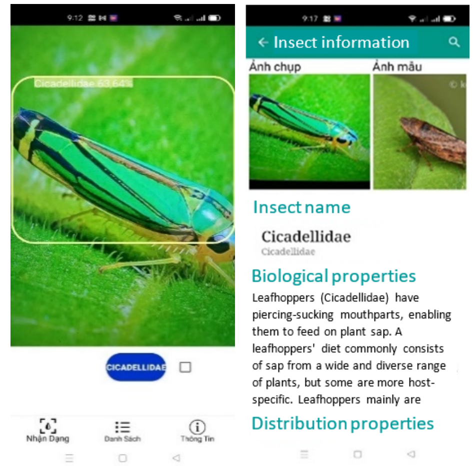


Fig. 9. The user Interface Screen shows the Successful Insect Recognition and Detailed Insect Information on a Mobile Device.

Fig. 10. The Insect Distribution Map was constructed based on GPS Location Information from the user's Insect Photos.

V. CONCLUSION AND FUTURE RESEARCH WORK

This paper presents an efficient system for real-time mobile smart device-based insect detection. Our system was developed based on the YOLOv5-S model because of its lightweight convolutional neural network and is thus suitable for mobile devices with limited hardware configuration. Moreover, insect pest detection and classification may be incorporated into hardware that farmers can utilize across a wide range of situations to safeguard their farms from pests. Therefore, our method has numerous advantages in terms of real-time insect identification, low cost, simple implementation, and practical implementation. The numerical results showed that the new system achieved 70.5% classification accuracy with mAP@0.5 on the Insect10 dataset and 42.9% accuracy with the large dataset IP102. This is the best insect pest detection result with YOLOv5-S ever reported from the largest insect dataset, IP102. However, these mAP accuracy results are still low when compared to the accuracy required for actual insect detection for agricultural production. Consequently, the next task will be to investigate more efficient recognition models in order to improve the accuracy and number of insects. Simultaneously, this work will be continued to study on better mobile devices, such as the NVIDIA Jetson Nano Developer Kit, which has a central processing unit, a graphical processing unit, a web camera, and currently only a low charge, allowing larger convolutional neural network models to be installed.

ACKNOWLEDGMENT

This study was funded by the National Geographic Society Exploration Grants (NGS-KOR-59552T-19), Microsoft AI for Earth, and the support of agriculture experts from An Giang University and Vietnam National University in Ho Chi Minh City, Vietnam.

REFERENCES

[1] S. Skendžić, M. Zovko, I. P. Živković, V. Lešić, and D. Lemić, The impact of climate change on agricultural insect pests, vol. 12, no. 5. 2021.

[2] "New standards to curb the global spread of plant pests and diseases." Food and Agriculture Organization of the United Nations (FAO), 2020, [Online]. Available: http://www.fao.org/news/story/en/item/ 1187738/icode/. [Last Access: 1-07-2020].

[3] K. Wang, Z. Shuifa, Z. Wang, Z. Liu, and F. Yang, "Mobile smart device-based vegetable disease and insect pest recognition method," Intell. Autom. Soft Comput., vol. 19, 2013, doi: 10.1080/10798587.2013.823783.

[4] H. Nasir, A. N. Aris, A. Lajis, K. Kadir, and S. I. Safie, "Development of Android Application for Pest Infestation Early Warning System," in 2018 IEEE 5th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA), 2018, pp. 1–5, doi: 10.1109/ICSIMA.2018.8688774.

[5] C. Zhu, J. Wang, H. Liu, and H. Mi, "Insect Identification and Counting in Stored Grain: Image Processing Approach and Application Embedded in Smartphones," Mob. Inf. Syst., vol. 2018, no. ii, 2018, doi: 10.1155/2018/5491706.

[6] P. Chudzik et al., "Mobile Real-Time Grasshopper Detection and Data Aggregation Framework," Sci. Rep., vol. 10, no. 1, p. 1150, 2020, doi: 10.1038/s41598-020-57674-8.

[7] J. W. Chen, W. J. Lin, H. J. Cheng, C. L. Hung, C. Y. Lin, and S. P. Chen, "A smartphone-based application for scale pest detection using multiple-object detection methods," Electron., vol. 10, no. 4, pp. 1–14, 2021, doi: 10.3390/electronics10040372.

[8] S. A. Lakmal Perera, "Pest Detecting Mobile Information System," 2021.

[9] M. E. Karar, F. Alsunaydi, S. Albusaymi, and S. Alotaibi, "A new mobile application of agricultural pests recognition using deep learning in cloud computing system," Alexandria Eng. J., vol. 60, no. 5, pp. 4423–4432, 2021, doi: 10.1016/j.aej.2021.03.009.

[10] P. Faria, T. Nogueira, A. Ferreira, C. Carlos, and L. Rosado, "AI-Powered Mobile Image Acquisition of Vineyard Insect Traps with Automatic Quality and Adequacy Assessment," Agronomy, vol. 11, no. 4, 2021, doi: 10.3390/agronomy11040731.

[11] W. Li, T. Zheng, Z. Yang, M. Li, C. Sun, and X. Yang, "Classification and detection of insects from field images using deep learning for smart pest management: A systematic review," Ecol. Inform., vol. 66, p. 101460, 2021, doi: 10.1016/j.ecoinf.2021.101460.

[12] A. D. Le, D. A. Pham, D. T. Pham, and H. B. Vo, "AlertTrap: {A} study on object detection in remote insects trap monitoring system using on-the-edge deep learning platform," CoRR, vol. abs/2112.1, 2021, [Online]. Available: https://arxiv.org/abs/2112.13341.

[13] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," 36th Int. Conf. Mach. Learn. ICML 2019, vol. 2019-June, pp. 10691–10700, 2019.

[14] J. Wu, "Power mean SVM for large scale visual classification," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 2344–2351, 2012, doi: 10.1109/CVPR.2012.6247946.

[15] D. T. Nghi, "Pest insect classification using efficientnet and power mean svm," 2021, doi: 10.15625/vap.2021.0050.

[16] F. Pezoa, J. L. Reutter, F. Suarez, M. Ugarte, and D. Vrgoč, "Foundations of JSON schema," in Proceedings of the 25th International Conference on World Wide Web, 2016, pp. 263–273.

[17] R. D. Hipp, "SQLite." 2020, [Online]. Available: https://www.sqlite.org/index.html.

[18] G. Jocher, "YOLOv5." Zenodo, 2020, doi: 10.5281/zenodo.4154370.

[19] C. Y. Wang, H. Y. Mark Liao, Y. H. Wu, P. Y. Chen, J. W. Hsieh, and I. H. Yeh, "CSPNet: A new backbone that can enhance learning capability of CNN," IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work., vol. 2020-June, pp. 1571–1580, 2020, doi: 10.1109/CVPRW50498.2020.00203.

[20] R. Xu, H. Lin, K. Lu, L. Cao, and Y. Liu, "A forest fire detection system based on ensemble learning," Forests, vol. 12, no. 2, pp. 1–17, 2021, doi: 10.3390/f12020217.

[21] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," 2020, [Online]. Available: http://arxiv.org/abs/2004.10934.

[22] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "{YOLOX:} Exceeding {YOLO} Series in 2021," CoRR, vol. abs/2107.0, 2021, [Online]. Available: https://arxiv.org/abs/2107.08430.

[23] Tzutalin, "LabelImg." 2015, [Online]. Available: https://github.com/tzutalin/labelImg.

[24] X. Wu, C. Zhan, Y. K. Lai, M. M. Cheng, and J. Yang, "IP102: A large-scale benchmark dataset for insect pest recognition," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2019-June, pp. 8779–8788, 2019, doi: 10.1109/CVPR.2019.00899.

[25] S. Ruder, "An overview of gradient descent optimization algorithms," arXiv Prepr. arXiv1609.04747, 2016.

[26] NVIDIA, "Jetson Nano Developer Kit User Guide." 2021, [Online]. Available: https://developer.download.nvidia.com/ embedded/L4T/r32-3-1_Release_v1.0/Jetson_Nano_ Developer_Kit_User_Guide.pdf.