Alexandria Engineering Journal (2021) 60, 4423–4432

HOSTED BY

ELSEVIER

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

A new mobile application of agricultural pests recognition using deep learning in cloud computing system

Mohamed Esmail Karar [a,b,*], **Fahad Alsunaydi** [a], **Sultan Albusaymi** [a], **Sultan Alotaibi** [a]

[a] *College of Computing and Information Technology Shaqra University, Shaqra, Saudi Arabia*
[b] *Department of Industrial Electronics and Control Engineering, Faculty of Electronic Engineering, Menoufia University, Minuf, Egypt*

Received 23 September 2020; revised 6 January 2021; accepted 9 March 2021
Available online 31 March 2021

KEYWORDS

Smart agriculture;
Crop pest;
Cloud computing;
Deep learning;
Faster R-CNN

Abstract Agricultural pests cause between 20 and 40 percent loss of global crop production every year as reported by the Food and Agriculture Organization (FAO). Therefore, smart agriculture presents the best option for farmers to apply artificial intelligence techniques integrated with modern information and communication technology to eliminate these harmful insect pests. Consequently, the productivity of their crops can be increased. Hence, this article introduces a new mobile application to automatically classify pests using a deep-learning solution for supporting specialists and farmers. The developed application utilizes faster region-based convolutional neural network (Faster R-CNN) to accomplish the recognition task of insect pests based on cloud computing. Furthermore, a database of recommended pesticides is linked with the detected crop pests to guide the farmers. This study has been successfully validated on five groups of pests; called Aphids, Cicadellidae, Flax Budworm, Flea Beetles, and Red Spider. The proposed Faster R-CNN showed highest accurate recognition results of 99.0% for all tested pest images. Moreover, our deep learning method outperforms other previous recognition methods, i.e., Single Shot Multi-Box Detector (SSD) MobileNet and traditional back propagation (BP) neural networks. The main prospect of this study is to realize our developed application for on-line recognition of agricultural pests in both the open field such as large farms and greenhouses for specific crops.

1. Introduction

Insect pests are one of major problems in the agricultural field. The Food and Agriculture Organization (FAO) reported that these pests cause between 20 and 40 percent loss of global crop production every year [1]. The pest infestation costs the global economy around $220 billion and invasive insects around US

* Corresponding author at: College of Computing and Information Technology, Shaqra University, Saudi Arabia.
E-mail addresses: mekarar@ieee.org, mkarar@su.edu.sa (M.E. Karar), s437460294@std.su.edu.sa (F. Alsunaydi), s437460397@std.su.edu.sa (S. Albusaymi), sultanrefa@su.edu.sa (S. Alotaibi).
Peer review under responsibility of Faculty of Engineering, Alexandria University.

https://doi.org/10.1016/j.aej.2021.03.009
1110-0168

$70 billion annually [1]. Therefore, the farmers use widely different pesticides to increase both the quality and the storage life of crops. But the continuous use of these pesticides resulted in the environmental contamination and potential high-risk diseases such as cancer, extreme respiratory and genetic diseases, and fetal death eventually [2]. Consequently, advanced technical solutions are highly needed in agriculture to early detect plant pests and save undesirable consumption of pesticides.

Smart agriculture has been recently introduced to apply artificial intelligence (AI) techniques, information and wireless communication technologies, e.g., Internet of Things (IoT), in all aspects of agriculture for realizing precision control of crop diseases, fertilization, irrigation and plant pests in the farm field [3–6]. Crop health monitoring is considered the main application of smart agriculture [4], defining the farm status with respect to plant pests. However, classification of crop pests is a challenging task for farmers because of complex structure and high similarity among insect pests [7]. In addition, traditional manual detection of insect pests is inefficient, expensive, and time-consuming. Nevertheless, recognizing plant pests at an early stage of infection will highly support the farmers to prevent the spread of these insects by selecting appropriate pesticides [7]. Therefore, computer vision systems and image processing using artificial intelligence techniques, e.g. machine learning and artificial neural networks, have been widely employed in previous studies to solve the above problems in the agricultural field.

Image-based recognition of insect pests have been proposed in entomology for many purposes, for example, insect counting as a biodiversity index [8]. In agriculture, deep learning methods such as convolutional neural networks (CNNs) have been recently applied as a good solution for automatic classification of crop pests [9–11]. Compared to traditional image processing methods and machine learning, the CNNs operate directly on raw pixels and have already its own feature generator. Moreover, CNNs proved robustness to handle image noise and illumination variation in many applications of medical image analysis [12,13], mechanical intelligent fault diagnosis [14], and infrastructure crack detection in construction [15].

Faster region-based convolutional neural network (Faster R-CNN) [16] is a unified deep CNN for target detection and identification in images, including feature detection, candidate regional generation, regional image classification, and location refinement. Therefore, we propose the Faster R-CNN to detect and classify crop pests using a new mobile-based application and cloud computing. Moreover, the contributions of this study are summarized as follow:

- Developing a new smartphone application to achieve image-based recognition of different classes of crop pests using Faster R-CNN in an intuitive manner.
- Unlike previous studies, our developed application saves the required resources and high-performance parallel computing of graphics processing unit (GPU) at the end-user by utilizing cloud computing services.
- Each class of detected crop pests are linked with the standard use of suggested pesticides to guide specialists and farmers.

The remainder of this paper is divided into five sections as follows. Section 2 gives a brief background of crop pests in this study, and an overview of the Faster R-CNN approach for object detection. Section 3 describes software tools in this study. The workflow of developed mobile-based application is also explained. Section 4 considers experimental setup and tested image dataset of insect pests to validate the developed recognition system. Results and comparative performance evaluation with the state-of-the art classification methods are also given. Finally, Sections 5 and 6 provide discussions and final concluding remarks with important prospects of this study.

2. Related work

For automatic deep learning-based classification of crop pests, many previous studies proposed image-based systems using different architectures of CNNs as follow. Deep residual learning has been proposed for recognizing ten classes of insect pest images in the presence of complex farmland background [11]. It achieved accurate performance of approximately 98.0% better than support vector machine and traditional BP neural networks. Also, adopted Region Proposal Network (RPN) was proposed to localized pests in the images and achieving classification process using CNNs [17]. Li et al. (2020) [18] presented a fine-tuned GoogLeNet model in a framework of deep learning-based pests detection to classify ten types of crop pests using a manually collected dataset. The resulted accuracy of pest detection is over 98.0%. However, this model requires high computing capacity, and the training time is also long. Some deep learning models such as Inception-V3 were difficult to be applied for the tested dataset. Combined saliency methods and CNNs were utilized for developing insect recognition system in [9]. It has been tested on both a small dataset and the large IP102 dataset [19], showing accuracy of 92.43% and 61.93%, respectively. But this developed method is failed in pest images with large intra-class variation. Deep CNN and transfer learning were also proposed for agricultural pest classification and tested on three public insect datasets of 24 to 40 classes of crop insect images [7]. They achieved classification accuracy over 95.0% better than pre-trained deep learning models. For cotton pest classification, deep residual networks were proposed to classify major primary and secondary pests in RGB images of the cotton field [10]. The obtained accuracy was 98.0%, outperforming other tested CNN models. Furthermore, improved deep learning pipeline was developed to detect and count crop pests in images based on self-learning saliency feature maps automatically [20]. A comparative study of machine learning against deep learning was conducted for detecting and identifying pests on tomato plants in greenhouses [21]. This study showed that the performance of deep learning solution is better than the proposed machine learning techniques to achieve accurate identification of tomato pests. Similarly, Kasinathan *et al.* [22] compared four machine learning techniques, namely artificial neural networks (ANN), support vector machine (SVM), k-nearest neighbors (KNN), naive bayes (NB) with a proposed CNN model to classify insect pests using the public IP102 dataset. The proposed CNN model achieved highest classification rates of 91.5% and 90.0% for 9 and 24 crop pest classes, respectively. For soybean fields, transfer leaning models [23–25] such as Residual Neural Network (ResNet-50) and Visual Geometry Group Network (VGG-16 and VGG-19) have been proposed to support farmers to manage the soybean pests control using acquired images by a drone with high resolution camera [26]. The resulted accuracy of highest classification rate was 93.82%.

3. Crop pests and technical background

3.1. Insect pests

This study includes five classes of crop pests, namely aphids, flea beetles, Cicadellidae, flax budworm, and red spider, as shown in Fig. 1. They are most popular pests in agriculture, especially in temperature regions. Each insect pest can be briefly defined as follows:

- **Aphids** are one of the most destructive insect pests on cultivated plants such as potatoes in the agricultural regions with high temperature [27]. They can transmit viruses from an infected plant to a healthy one, leading to damage and significant reduction in the crops production.
- **Flea Beetles** are major insect pests, which attack crops in Europe and North America [28]. Their feeding causes seriously damage young plants and loss of the crop eventually.
- **Cicadellidae** is a pest that may cause a half crop loss if severe infestation occurs, e.g. Mango hopper [29].
- **Flax budworm** causes potentially infestations to reduce crops by 40–90% in temperature regions [30].
- **Red Spider Mites** are a widespread insect pest in all agricultural regions. They attack with considerable damage to many host plants either in the open field or greenhouses; for example, eggplants, beans, melons, and tomatoes [31].

3.2. Faster R-CNN approach

The Faster R-CNN presents recent advancement of both Fast R-CNN and R-CNN by Ren *et al.* [16]. It merges a Region Proposal Network (RPN) as object bounds predictor and Fast R-CNN for target detection in processed images. The role of RPN module is to serve as the "attention" mechanism of this unified Faster R-CNN, as depicted in Fig. 2. Three basic components of the Faster R-CNN can be described as follow. First, the feature extractor network to generate feature maps from the input image (see Fig. 2). It can be a pre-trained CNN architecture such as Inception [32], Residual Neural Network (Resnet) [33], and Dense Convolutional Network (Densenet) [34]. Second, the RPN module proposes object locations of the feature maps. Third, a regressor and classifier are trained using the loss function L in (1) for the CNN detection network to adjust these proposed locations and to predict single or multi-object classes with the corresponding bounding box area in the resulted image, as shown in Fig. 2.

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i,p_i^*) + \lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}(t_i,t_i^*). \quad (1)$$

Here,p_i is the predicted probability of each anchor i as an object in the image.$p_i^*$is the ground-truth label for an anchor, such that$p_i^* = 1$ if the anchor is positive and zero for negative anchors. $t_i$is a vector to represent parameterized coordinates of the bounding box for the predicted object, and t_i^*represents the ground-truth for the bounding box linked with positive anchors. L_{cls} and L_{reg} are the classification and regression loss functions, respectively. The regression loss $p_i^* L_{reg}$ is only active for positive anchors. N_{cls} and N_{reg} are weighted factors for normalized two terms with a balanced parameterλ.$\{p_i\},\{t_i\}$are the outputs of classification and regression layers, respectively. The Faster R-CNN approach was further detailed in [16].

4. Materials and methods

4.1. Dataset

Collecting pest images is a challenging task because all insect pests have many stages during their lifetime depending on the species and the type of each pest. Therefore, we have used the pest images from the public IP102 dataset [19]. It contains more than 75,000 images belongs to 102 categories of agricultural pests. Due to the limited space of cloud hosting server and availability of recommended pesticides database, this study focused only on five classes of insect pests; called aphids, flea beetles, Cicadellidae, flax budworm, and red spider mite, as described above. To balance training phase of the Faster R-CNN, 500 images have been also chosen for all 5 classes of insect pests, as shown in Fig. 1.

Fig. 1 A sample of five classes of insect pest images; called aphids, flea beetles, Cicadellidae, flax budworm, and red spider mite. The images are collected from public dataset [19]. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

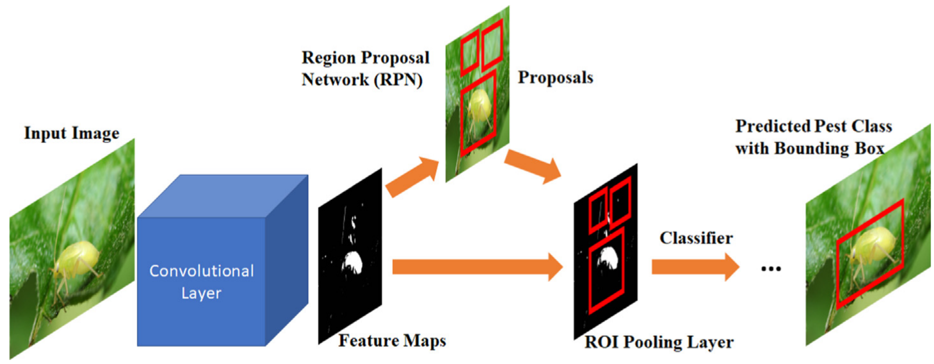

Fig. 2 The basic workflow of Faster R-CNN for detection and classification of target object class, i.e., crop pests in this study.

4.2. Software tools

Our mobile application of insect pest recognition has been developed based on different open-source toolkits and modules using Python programming [35] as follows:

- **Apache Cordova** framework [36] is used in this study to develop the mobile application using markup languages such as HTML. Moreover, it allows to be distributed and installed on different mobile operating systems such as Android and iOS.
- **Flask web** [37] is applied for handling the HTTP POST requests sent by the developed mobile-based user interface. Then, the recognition result of crop pests is retrieved and displayed.
- **MySQL database management system** [38] has been used to save all information and ground-truth images of the insect pests, e.g. the pest name and the appropriate use of pesticides for crop protection.
- **PythonAnywhere** [39] is a platform solution consisting of a Python online integrated development environment (IDE) and web hosting environment to create and run Python code in the cloud.

4.3. Developed pest recognition system

A new mobile-based application has been developed for identifying the crop insects in the cloud, as depicted in Fig. 3. The framework of our developed deep learning-based cloud computing system is composed of three main modules to accomplish automatic detection and classification of insect pests as follows.

4.3.1. User interface module

A farmer in the open field or a greenhouse can easily identify the crop pest via the developed user interface (UI) on a smartphone with any platform, i.e., Android-based or iOS devices. The UI sends a HTTP POST request that carries the captured image of unknown insect pest by the user. The flask web handles all requests by saving the input pest images in the cloud and passing them to the deep learning module for further processing. At the end of image-based recognition procedure, the results of pest classification and related pesticides are displayed on the developed UI to guide the farmers with standard agricultural instructions.

4.3.2. Deep learning module

The deep learning module is responsible for image processing to detect and classify crop pests using Faster R-CNN approach, as shown above in Fig. 2. The convolutional layers of InceptionV2 model [40] has been used in this study as a feature extractor for the Faster R-CNN because of its advantage to balance between the processing speed and accuracy in modern deep learning-based detection systems [41]. The network weights of InceptionV2 was pre-trained on the COCO dataset [42] with minimum and maximum image resolutions of 600 and 1024 pixels, respectively. Here, the size of captured input image or the pest dataset is fixed to 224 $\times$ 224 pixels and is resized on-line to the above aspect ratio of min–max dimensions [600, 1024] during training and testing phases. As earlier described, the InceptionV2 model generates the feature maps of input image. Then, the RPN takes this feature maps image as input to provide a set of rectangular region proposals as output. In this step, a grid-anchor of size 16 $\times$ 16 pixels is initiated with scales [0.25, 0.5, 1.0, 2.0]. The Intersection over Union (IoU) is given in (2), where A and B are two areas of the region proposals. To detect object successfully, a non-maximum suppression (NMS) is usually used to ignore small overlapping bounding boxes and returning those with large overlaps. Therefore, the NMS-IoU threshold is set to 0.7.

$$IoU = \frac{A \cap B}{A \cup B} \tag{2}$$

Next, the Region of Interest (RoI) pooling layer takes the region corresponding to a proposal from the backbone feature map and dividing it into sub-windows. The maximum pooling is performed over these sub-windows to give the output of RoI pooling layer, which has a size of (N, 7, 7, 512), and N is the number of generated region proposals by the RPN mechanism, as shown in Fig. 2. After passing through two fully connected layers (FCs), the features are fed into the classifier and regressor branches as follow. The classification layer calculates the probability of the region proposal contains an object such that the probability Pi of each element in the feature map i contains the target object using softmax function. The regions with the top score of 300 Pi in the total rankings present the detected target object. In addition, the regressor provides the IoU index

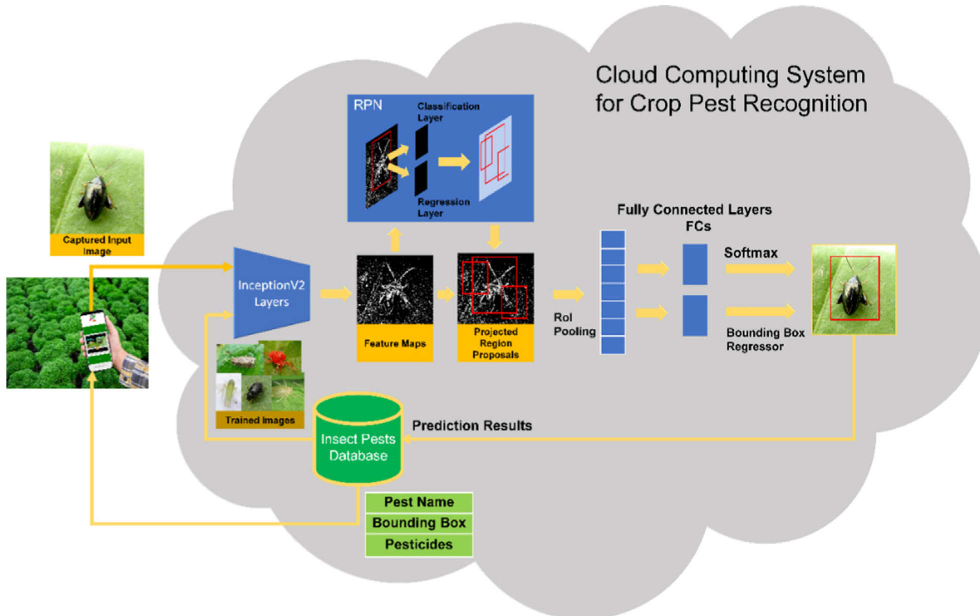

Fig. 3 Schematic diagram of the developed mobile-based system for recognizing crop pests using Faster R-CNN and cloud computing.

in (2) to measure the accuracy of the bounding box around the object. The bounding box regression predicts the center point coordinates (x, y) of the anchor box, defining its width w and the height h.

4.3.3. Database module

As shown in Fig. 3, the database module provides all pest images of five classes for training Faster R-CNN in the cloud computing system. Also, it has a major role to store and retrieve all data of the resulted pest prediction containing the name and bounding box of detected pests in the tested images, and the use of recommended pesticides.

4.4. Classification performance analysis

To quantify the performance of the developed mobile application for recognizing insect pests, cross validation estimator [43] has been used for comparing between the true crop pest labels of tested images and predicted classification results of Faster R-CNN model. That generates an error matrix or a confusion matrix with four expected outcomes, namely True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN), as illustrated in Table 1.

Consequently, we have calculated four well-known evaluation metrics as follows. First, ***accuracy*** is one of the important metrics for image-based classifiers. It is estimated by summation of true positives (TP) and true negatives (TN) divided by all possible outcomes of classification process, as given in (3). Second, ***precision*** is given in (4) to represent relationship between the true positive predicted values and all positive predicted values. Third, the sensitivity or ***recall*** in (5) presents the ratio between the true predictive positive values and the summation of both predicted true positive and predicted false values. Finally, the fourth evaluation metric is ***F1-score***, which demonstrates the overall accuracy measure of the developed classifier by combining the double ratio of both the precision and recall metrics in (6).

$$Accuracy(\%) = \frac{TP + TN}{TP + FP + FN + TN} 100\% \quad (3)$$

$$precision = \frac{TP}{TP + FP} \quad (4)$$

$$recall = \frac{TP}{TP + FN} \quad (5)$$

$$F1 - score = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \quad (6)$$

Table 1 A confusion matrix.

		Predicted pest classes	
		positives	negatives
Actual pest classes	positives	True Positive (TP)	False Negative (FN)
	negatives	False Positive (FP)	True Negative (TN)

5. Experiments

5.1. Experimental setup

All experiments of our developed mobile-based crop pest recognition system were done using the Tensorflow object detection API framework [36] on a GPU of NVIDIA GAMINGX GTX 1080 8 GB and DDR4-16 GB memory. Before starting the training-step of the deep learning module, the NVIDIA CUDA Deep Neural Network library (cuDNN) was installed to accelerate the GPU computations for deep

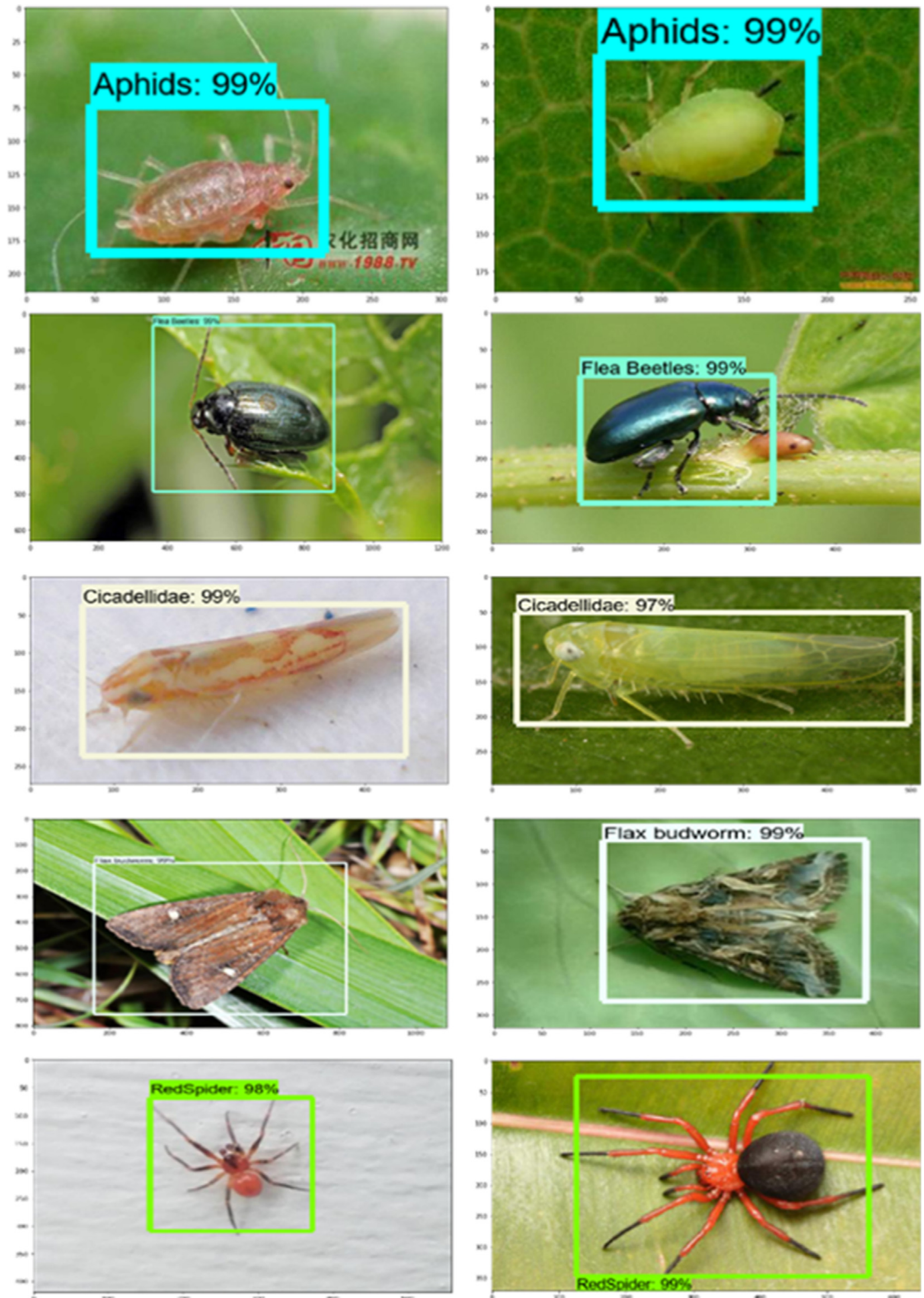

Fig. 4 A sample of classification results and accuracy of the developed mobile application for all tested crop pests.

neural networks. A smartphone with iOS V13.5 was used to test the detection and classification of crop pest images in the cloud.

In this study, total number of images for five classes of insect pests (see Fig. 1) are 500 collected from the public IP102 dataset [17]. These images are split into 80–20% for training and testing steps, respectively. The training hyperparameters of Faster R-CNN are carefully chosen to achieve accurate identification of pests using Stochastic Gradient Descent (SGD) optimizer with the momentum value of 0.9. The initial value of learning rate is set to 0.001 and decreases to 0.0004 after 40 K steps.

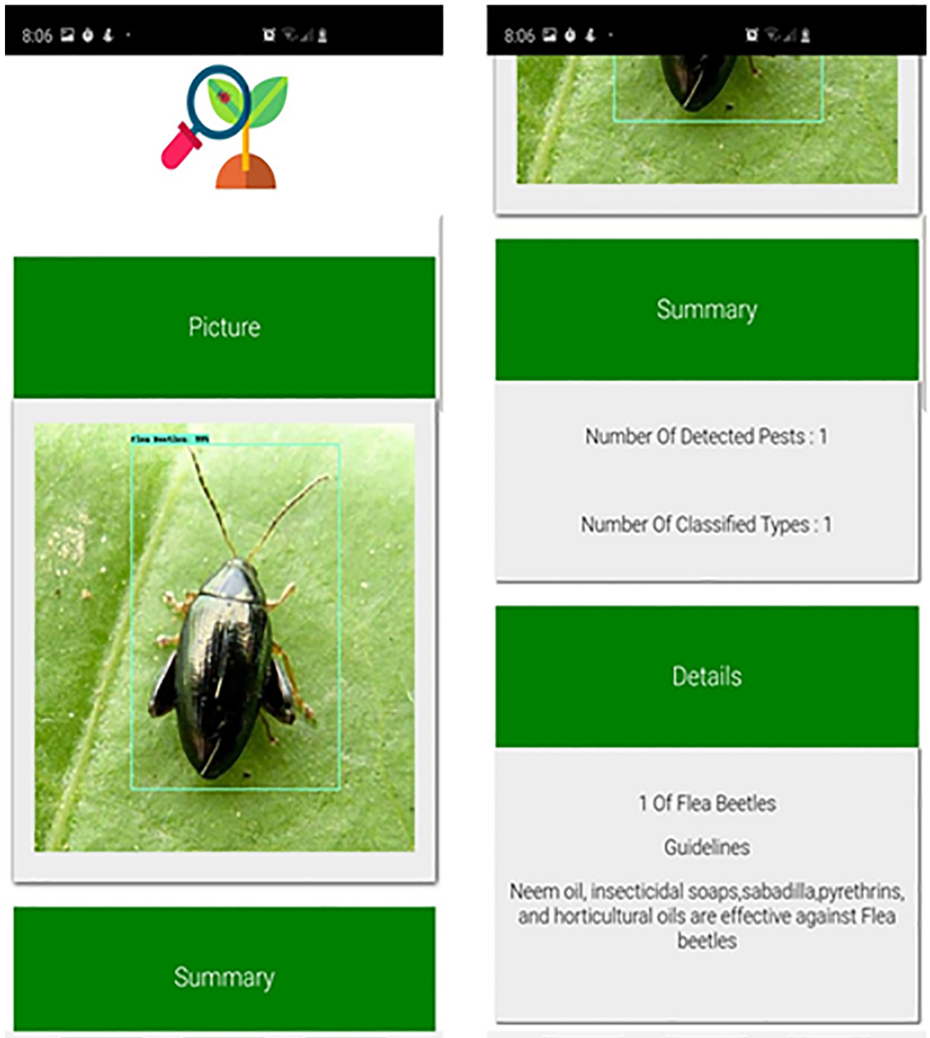

Fig. 5 User Interfaces (UI) of developed mobile application for recognizing crop pests with related pesticides.

Table 2 Comparative performance evaluation of crop pest classifiers using the developed system.

Crop pest	**BP Neural Network** [44]			**SSD MobileNet** [46]			**Proposed Faster R-CNN**		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
Aphids	0.82	0.90	0.86	**1.00***	0.70	0.82	**0.87**	**1.00**	0.93
Flea beetles	0.62	0.75	0.68	**1.00**	0.80	0.89	**1.00**	**1.00**	**1.00**
Cicadellidae	0.50	0.10	0.17	0.91	**1.00**	**0.95**	**1.00**	0.85	0.92
Flax budworm	0.40	**1.00**	0.57	0.89	0.80	0.84	**1.00**	**1.00**	**1.00**
Red spider mites	0.00	0.00	0.00	0.91	**1.00**	0.95	**1.00**	**1.00**	**1.00**
Accuracy (average)	50%			86%			**98%**		

*Best performance values are indicated in bold.

5.2. Results

Fig. 4 shows a sample of detection and classification results using the developed recognition system based on Faster R-CNN. High classification accuracy was achieved for all tested images of insect pests in the range of 97.0 to 99.0%. Moreover, the developed UI of our mobile application is depicted in Fig. 5, monitoring the identification results; for example, in case of the flea beetles with a summary including the suggested pesticides such as Neem oil and insecticidal soaps for guiding farmers.

5.3. Comparative performance evaluation

The overall performance of developed system for insect pest recognition has been evaluated using four basic classification metrics in (3–6); called accuracy, precision, recall, and F1-score. Furthermore, Table 2 illustrates a comparison of the resulted pest classification performance with other methods in previous studies, i.e. back propagation (BP) neural network [44] and Single-Shot Detector (SSD) MobileNet [45,46]. These results verify the outperformance of our proposed Faster R-CNN for automatic identification of crop pests. The BP neural networks algorithm is relatively succeeded to classify aphids, but it is completely failed to detect the red spider mites, as illustrated in Table 2. The SSD MobileNet showed a good performance for identifying all tested images of insect pests. However, the proposed Faster R-CNN achieved the highest value of average accuracy (98.0%) better than the accuracies of BP neural network and SSD MobileNet of 50.0% and 86.0%, respectively.

In addition, the classification performance of our proposed Faster R-CNN model has been validated on other different sizes of tested pest images by varying the percentage of dataset split into 70–30% and 90–10% for training and testing data, respectively. Fig. 6 depicts the comparative classification accuracy of BP neural network, SSD MobileNet and proposed Faster R-CNN models under 10%, 20% and 30% test data of the full dataset in this study. Generally, the performance of crop pest classifiers is improved by increasing the percentage of training image data from 70% to 90% of the full dataset. The classification accuracy of both BP neural network and SSD MobileNet become 43.0% and 85.6%, respectively at 30% testing data, while the classification accuracy of our proposed Faster R-CNN is still high 94.0% at the same test data size.

Moreover, Table 3 illustrates a comparison between our proposed pest recognition method using Faster R-CNN and other methods in previous studies with respect to the pest image dataset, number of pest classes and classification accuracy scores. Deep residual networks and transfer learning models have been used to identify the cotton and soybean pests, respectively [10,26]. They achieved accuracy scores of 98.0% for primary and secondary cotton pests, and 93.8% for 13 classes of the soybean pests. Although different machine learning techniques [22] and bio-inspired methods [9] have been applied to classify 9 to 24 insect pest classes based on the images of IP102 dataset [19], they achieved accuracy score less than 93.0%. Obviously, our proposed Faster R-CNN classifier achieved the best performance accuracy score of 98.8% on five crop pest classes using the same image dataset.

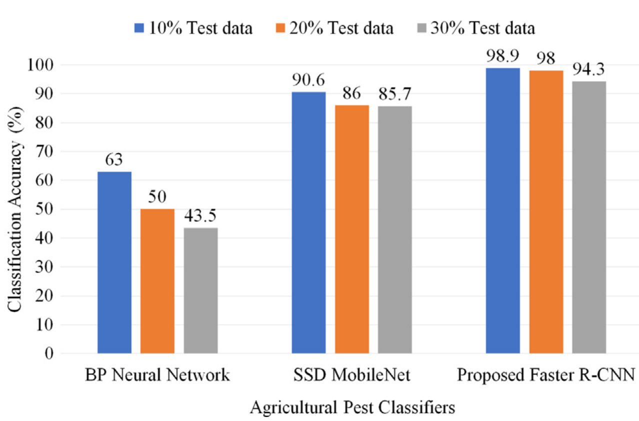

Fig. 6 Comparative accuracy of agricultural pest classifiers at three different sizes of testing data as a percentage of the full dataset. The proposed Faster R-CNN classifier showed the best performance for identifying crop pests using all tested data.

Table 3 Comparison between our proposed agricultural pest classifier and other methods in the literature.

Method	Dataset	Number of Pest Classes	Accuracy (%)
Bio-inspired methods [9]	IP102 dataset [19]	10	92.4%
Deep residual networks [10]	Cotton pests	2	98.0%
Transfer learning models [26]	Soybean pests	13	93.8%
Machine learning techniques [22]	IP102 dataset	9 24	91.5% 90.0%
Our proposed Faster R-CNN	IP102 dataset	5	**98.9%**

*Best performance value is indicated in bold.

6. Discussions

The results of this study demonstrated that our developed mobile application is successfully valid for automated management control of crop pests, as shown in Figs. 4 and 5. The developed application is mainly based on Faster R-CNN to identify five popular agricultural pests (see Fig. 1). Compared to machine learning and deep learning classifiers in previous studies, the proposed Faster R-CNN showed a good performance to achieve the highest classification score of 98.9% as illustrated in Tables 2 and 3. The advantages of our proposed Faster R-CNN has been verified as follows. First, crop pests have been successfully detected in similar and complex background of tested images. Second, the Faster R-CNN is suitable for real-time identification of agricultural pests in the field with no prior knowledge about the number of objects in acquired images. Finally, predicting the categories and positions of different crop pests are accurately identified because of the outperformance utilization of the RPN module with the bounding box regression, as depicted in Fig. 2. Moreover, the computational time of this deep learning approach is not a problem in this study, because the detection and classification procedures of agricultural insect pests can be executed in a few seconds in the cloud computing system, saving all required hardware resources.

Although this study is limited to use only Faster R-CNN InceptionV2 as a proposed deep learning module of our developed recognition system, it can be easily extended to include other pre-trained deep feature extractors such as Resnet and Densenet models. In cloud computing environment, some types of faults can be occurred such as network fault of internet connection, physical faults of hardware resources, faults in the processing of developed software. Therefore, the fault tolerance is a critical requirement to keep the functionality and expected performance of cloud computing services and mobile applications regardless of these possible failures [47–49]. Two reactive fault tolerance mechanisms have been applied in this study as follows. First, the recognition task of insect pests is executed repetitively until it gives the successful result. It is also possible for the user to retry the failed and/or unsuccessful task manually [50]. Second, the failed task of agricultural pest recognition is re-submitted to the same resource or another mobile device for completing the software execution correctly [47].

We are currently working on extending the number of insect pest classes in our database, considering specific crops either in the open field or greenhouses. That will also support the farmers to perform multi-class classification of targeted crop pests simultaneously, as presented in Ref. [51]. Moreover, security and privacy algorithms of analyzed images [52,53] for transmission over public mobile networks can be also involved in the proposed cloud computing system , as shown in Fig. 4. Nevertheless, the current version of our developed mobile-based application is still valid and accurate to identify agricultural pests successfully.

7. Conclusions

This study presented a new mobile application for detection and classification of crop pests based on Faster R-CNN and cloud computing system. Five classes of well-known crop pests are successfully classified using the developed image-based recognition system, as illustrated in Table 2. The evaluation results of insect pest classification using the proposed Faster R-CNN showed superior performance compared to the state-of-the art methods, namely BP neural networks and SSD MobileNet. Moreover, the use of corresponding pesticides is integrated with the pest classification results to guide specialists and farmers, as shown in Fig. 5. The main prospect of this research work is adding new agricultural pest classes with recommended pesticides for specific crops. Furthermore, building-up a wireless motion sensor network for real-time detection of insect pests will be considered in the future work of our developed mobile-based recognition system.

8. Compliance with ethical standards**a

Funding: This research received no external funding.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This project is supported by Shaqra University, Saudi Arabia.

References

[1] Food and Agriculture Organization of the United Nations (FAO) (2020) New standards to curb the global spread of plant pests and diseases. http://www.fao.org/news/story/en/item/1187738/icode/. [Last Access: 1-07-2020].
[2] S.T. Narenderan, S.N. Meyyanathan, B. Babu, Review of pesticide residue analysis in fruits and vegetables. Pre-treatment, extraction and detection techniques, Food Res. Int. 133 (2020) 109141.
[3] Z. Hu, L. Xu, L. Cao, S. Liu, Z. Luo, J. Wang, X. Li, L. Wang, Application of Non-Orthogonal Multiple Access in Wireless Sensor Networks for Smart Agriculture, IEEE Access 7 (2019) 87582–87592.
[4] M. Ayaz, M. Ammad-Uddin, Z. Sharif, A. Mansour, E.M. Aggoune, Internet-of-Things (IoT)-Based Smart Agriculture: Toward Making the Fields Talk, IEEE Access 7 (2019) 129551–129583.
[5] M.S. Farooq, S. Riaz, A. Abid, K. Abid, M.A. Naeem, A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming, IEEE Access 7 (2019) 156237–156271.
[6] M.E. Karar, M. al-Rasheed, A. Al-Rasheed, O. Reyad, IoT and Neural Network-Based Water Pumping Control System For Smart Irrigation, Inform. Sci. Lett., 9 (2020) 107–112.
[7] K. Thenmozhi, U. Srinivasulu Reddy, Crop pest classification based on deep convolutional neural network and transfer learning, Comput. Electron. Agric. 164 (2019) 104906.
[8] M. Martineau, D. Conte, R. Raveaux, I. Arnault, D. Munier, G. Venturini, A survey on image-based insect classification, Pattern Recogn. 65 (2017) 273–284.
[9] L. Nanni, G. Maguolo, F. Pancino, Insect pest image detection and recognition based on bio-inspired methods, Ecol. Inf. 57 (2020) 101089.
[10] A.N. Alves, W.S.R. Souza, D.L. Borges, Cotton pests classification in field-based images using deep residual networks, Comput. Electron. Agric. 174 (2020) 105488.
[11] X. Cheng, Y. Zhang, Y. Chen, Y. Wu, Y. Yue, Pest identification via deep residual learning in complex background, Comput. Electron. Agric. 141 (2017) 351–356.
[12] R.A. Zeineldin, M.E. Karar, J. Coburger, C.R. Wirtz, O. Burgert, DeepSeg: deep neural network framework for automatic brain tumor segmentation using magnetic resonance FLAIR images, Int. J. Comput. Assist. Radiol. Surg. 15 (2020) 909–920.
[13] M.E. Karar, Robust RBF neural network–based backstepping controller for implantable cardiac pacemakers, Int. J. Adapt Control Signal Process. 32 (2018) 1040–1051.
[14] K. Zhang, J. Chen, T. Zhang, Z. Zhou, A Compact Convolutional Neural Network Augmented with Multiscale Feature Extraction of Acquired Monitoring Data for Mechanical Intelligent Fault Diagnosis, J. Manuf. Syst. 55 (2020) 273–284.
[15] Q. Yang, W. Shi, J. Chen, W. Lin, Deep convolution neural network-based transfer learning method for civil infrastructure crack detection, Autom. Constr. 116 (2020) 103199.
[16] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, IEEE Trans. Pattern Anal. Mach. Intell. 39 (2017) 1137–1149.
[17] D. Xia, P. Chen, B. Wang, J. Zhang, C. Xie, Insect Detection and Classification Based on an Improved Convolutional Neural Network, Sensors (Basel) 18 (2018) 4169.
[18] H. Chen, A. Chen, L. Xu, H. Xie, H. Qiao, Q. Lin, K. Cai, A deep learning CNN architecture applied in smart near-infrared analysis of water pollution for agricultural irrigation resources, Agric. Water Manag. 240 (2020) 106303.
[19] X. Wu, C. Zhan, Y. Lai, M. Cheng, J. Yang, IP102: A Large-Scale Benchmark Dataset for Insect Pest Recognition, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2019 (2019) 8779–8788.
[20] W. Li, P. Chen, B. Wang, C. Xie, Automatic Localization and Count of Agricultural Crop Pests Based on an Improved Deep Learning Pipeline, Sci. Rep. 9 (2019) 7024.
[21] A. Gutierrez, A. Ansuategi, L. Susperregi, C. Tubío, I. Rankić, L. Lenža, A Benchmarking of Learning Strategies for Pest Detection and Identification on Tomato Plants for Autonomous Scouting Robots Using Internal Databases, J. Sens. 2019 (2019) 5219471.

[22] T. Kasinathan, D. Singaraju, S.R. Uyyala, Insect classification and detection in field crops using modern machine learning techniques, Inform. Process. Agric. (2020).
[23] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-Decem (2016), pp. 770–778.
[24] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, arXiv:1409.1556 [cs.CV], (2015).
[25] M.E. Karar, E.E.-D. Hemdan, M.A. Shouman, Cascaded deep learning classifiers for computer-aided diagnosis of COVID-19 and pneumonia diseases in X-ray scans, Complex Intell. Syst. 7 (2021) 235–247, https://doi.org/10.1007/s40747-020-00199-4.
[26] E.C. Tetila, B.B. Machado, G. Astolfi, N.A.d.S. Belete, W.P. Amorim, A.R. Roel, H. Pistori, Detection and classification of soybean pests using deep learning with UAV images, Comput. Electron. Agric., 179 (2020) 105836.
[27] Y. Xu, S.M. Gray, Aphids and their transmitted potato viruses: A continuous challenges in potato crops, J. Integr. Agric. 19 (2020) 367–375.
[28] O. Lundin, No-till protects spring oilseed rape (Brassica napus L.) against crop damage by flea beetles (Phyllotreta spp.), Agric. Ecosyst. Environ. 278 (2019) 1–5.
[29] S. Seetha, S. Sahu, B.B. Mahapatra, M. Mishra, Post Embryonic Changes in the Eye of an Economic Mango Plant Pest Amritodus atkinsoni Leth. (Hemiptera: Membracoidea: Cicadellidae), in: A.Y. Oral, Z.B. Bahsi Oral (Eds.), 3rd International Multidisciplinary Microscopy and Microanalysis Congress (InterM), Springer International Publishing, Cham, 2017, pp. 25–31.
[30] D.S. Hill, Agricultural Insect Pests of Temperate Regions and Their Control, Cambridge University Press, 1987.
[31] E. Piccinini, V. Ferrari, G. Campanelli, F. Fusari, L. Righetti, R. Matteo, L. Lazzeri, Effect of two bio-based liquid formulations from Brassica carinata in containing red spider mite (Tetranychus urticae) on eggplant, Ind. Crops Prod. 75 (2015) 36–41.
[32] C. Szegedy, S. Ioffe, V. Vanhoucke, A.A. Alemi, Inception-v4, inception-ResNet and the impact of residual connections on learning, Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI Press, San Francisco, California, USA, 2017, pp. 4278–4284.
[33] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[34] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 2261–2269.
[35] I. Vasilev, Advanced Deep Learning with Python: Design and implement advanced next-generation AI solutions using TensorFlow and PyTorch, Packt Publishing, 2019.
[36] J.M. Wargo, Apache Cordova 4 Programming, Pearson Education, 2015.
[37] M. Grinberg, Flask Web Development: Developing Web Applications with Python, O'Reilly Media, 2018.
[38] V. Siahaan, R.H. Sianipar, A PRACTICAL GUIDE TO Database Programming with PHP/MySQL, Independ. Publ. (2019).
[39] A.F. Villán, Mastering OpenCV 4 with Python: A practical guide covering topics from image processing, augmented reality to deep learning with OpenCV 4 and Python 3.7, Packt Publ. (2019).
[40] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the Inception Architecture for Computer Vision, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, 2016, pp. 2818–2826.
[41] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, K. Murphy, Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, 2017, pp. 3296–3297.
[42] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft COCO: Common Objects in Context, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, Springer International Publishing, Cham, 2014, pp. 740–755.
[43] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Inf. Process. Manage. 45 (2009) 427–437.
[44] Y. Liu, W. Jing, L. Xu, Parallelizing Backpropagation Neural Network Using MapReduce and Cascading Model, Comput. Intell. Neurosci. 2016 (2016) 2842780.
[45] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, SSD: Single Shot MultiBox Detector, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, Springer International Publishing, Cham, 2016, pp. 21–37.
[46] A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, (2017).
[47] P. Kumari, P. Kaur, A survey of fault tolerance in cloud computing, J. King Saud Univ. – Comput. Inform. Sci. (2018).
[48] L. Ravindranath, S. Nath, J. Padhye, H. Balakrishnan, Automatic and scalable fault detection for mobile applications, Proceedings of the 12th annual international conference on Mobile systems, applications, and services, Association for Computing Machinery, Bretton Woods, New Hampshire, USA, 2014, pp. 190–203.
[49] S.M. Darwish, M.N. El-Dirini, I.A. Abd El-Moghith, An adaptive cellular automata scheme for diagnosis of fault tolerance and connectivity preserving in wireless sensor networks, Alexandria Eng. J. 57 (2018) 4267–4275.
[50] S. Prathiba, S. Sowvarnica, Survey of failures and fault tolerance in cloud, 2017 2nd International Conference on Computing and Communications Technologies (ICCCT), 2017, pp. 169–172.
[51] L. Liu, R. Wang, C. Xie, P. Yang, F. Wang, S. Sudirman, W. Liu, PestNet: An End-to-End Deep Learning Approach for Large-Scale Multi-Class Pest Detection and Classification, IEEE Access 7 (2019) 45301–45312.
[52] O. Reyad, K. Hamed, M.E. Karar, Hash-enhanced elliptic curve bit-string generator for medical image encryption, J. Intell. Fuzzy Syst. 39 (2020) 7795–7806.
[53] O. Reyad, M.E. Karar, Secure CT-Image Encryption for COVID-19 Infections Using HBBS-Based Multiple Key-Streams, Arabian J. Sci. Eng. 46 (2021) 3581–3593, https://doi.org/10.1007/s13369-020-05196-w.