IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 19 May 2022, accepted 3 July 2022, date of publication 11 July 2022, date of current version 18 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3189676

RESEARCH ARTICLE

An Efficient Approach for Crops Pests Recognition and Classification Based on Novel DeepPestNet Deep Learning Model

NAEEM ULLAH[1], JAVED ALI KHAN[2], LUBNA ABDULAZIZ ALHARBI[3], ASAF RAZA[1], WAHAB KHAN[4], AND IJAZ AHMAD[5,6,7]

[1]Department of Software Engineering, University of Engineering and Technology, Taxila, Taxila 4400, Pakistan
[2]Department of Software Engineering, University of Science and Technology, Bannu, Bannu 28100, Pakistan
[3]Department of Computer Science, College of Computers and Information Technology, University of Tabuk, Tabuk 47713, Saudi Arabia
[4]Department of Electrical Engineering, University of Science and Technology, Bannu, Bannu 28100, Pakistan
[5]CAS Key Laboratory of Human-Machine Intelligence-Synergy Systems, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
[6]Shenzhen College of Advanced Technology, University of Chinese Academy of Sciences, Shenzhen 518055, China
[7]Guangdong-Hong Kong-Macao Joint Laboratory of Human-Machine Intelligence-Synergy Systems, Chinese Academy of Sciences, Shenzhen 518055, China

Corresponding authors: Javed Ali Khan (javed_ali@ustb.edu.pk) and Lubna Abdulaziz Alharbi (lualharbi@ut.edu.sa)

ABSTRACT Crop pests are to blame for significant economic, social, and environmental losses worldwide. Various pests have different control strategies, and precisely identifying pests has become crucial to pest control and is a significant difficulty in agriculture. Many agricultural professionals are interested in deep learning (DL) models since they have shown significant promise in image recognition. Pest identification approaches in literature have relatively low accuracy in pest recognition and classification due to the complexity of their algorithms and limited data availability. Misclassification of insect pests sometimes leads to using the wrong pesticides, causing harm to agricultural yields and the surrounding environment. It necessitates developing an automated system capable of more accurate pest identification and classification. This paper presents a novel end-to-end DeepPestNet framework for pest recognition and classification. The proposed model has 11 learnable layers, including eight convolutional and three fully connected (FC) layers. We used image rotations techniques to increase the size of the dataset and image augmentations techniques to test the generalizability of the proposed DeepPestNet approach. We used the popular Deng's crops data set to assess the proposed DeepPestNet framework. We used the proposed method to recognize and classify crop pests into 10-class pests, i.e., Locusta migratoria, Euproctis pseudoconspersa strand, chrysochus Chinensis, empoasca flavescens, Spodoptera exigua, larva of laspeyresia pomonella, parasa lepida, acrida cinerea, larva of S. exigua, and L.pomonella types of insects pests. The proposed method achieved optimal accuracy of 100%. We compared the proposed DeepPestNet approach with traditional pre-trained deep learning (DL) models. To verify the general adaptability of this model, we tested the proposed model on the standard Kaggle dataset "Pest Dataset" to recognize nine types of pests: aphids, armyworm, beetle, bollworm, grasshopper, mites, mosquito, sawfly, and stem borer and achieved an accuracy of 98.92%. The proposed model can provide specialists and farmers with immediate and effective aid in recognizing pests, potentially reducing economic and crop yield losses.

INDEX TERMS Insects pests, deep learning, transfer learning, fine-tuning, convolutional neural networks.

I. INTRODUCTION

By 2050, the world's population is expected to grow by almost 2 billion people. As a result, food safety appears to be among the most pressing issues in the future [1]. Increased productivity in agricultural production, which is among the growing study fields, is one solution to this challenge. Pest control is critical for increasing agricultural output and food quality while reducing costs and increasing profits, which has recently become significant. Insect pests are one of the

The associate editor coordinating the review of this manuscript and approving it for publication was Kumaradevan Punithakumar.

most common causes of agricultural damage worldwide. The mitigation of these losses could save a significant amount of the yield and increase agricultural profitability. Pest infestations on harvests cause various diseases and harm harvests, resulting in low yields [2]. Insect damage to harvest regions such as rice, wheat, and beans is one of the major reasons driving yield losses. Insecticides and other bio control techniques should be used to minimize insect crop losses and reduce insect populations and avoid them from spreading over broad areas. Pesticides and chemicals have a big impact on pest control. They will, however, have several detrimental effects on human health and the environment. As a result, agricultural scientists worldwide began working together to develop better pest control replacements for chemical pesticides. Addressing and minimizing such issues would positively result in the entire sector's economic growth to generate a large amount of food with fewer natural resources would be required

In addition, because pest management procedures change depending on the pest species, identifying the insect is critical. The capacity to recognize and classify insects, to distinguish between the healthy and dangerous ones, is the first step in preventing crop damage caused by insect pests. However, due to the intricate anatomy of insects and the resemblance between various insect species, insect classification is a difficult undertaking. Entomologists have typically classified insects by hand, which is time-consuming and difficult and necessitates a thorough understanding [3]. Because this activity necessitates ongoing and costly supervision, automatic pest classification has gained popularity in recent years. Furthermore, to solve these issues, experts have used various computer-aided categorization techniques. Many computerized pest classification methods have recently been presented based on machine learning (ML) methods [4]–[7]. Traditional ML algorithms, on the other hand, have some drawbacks. Traditional ML techniques have been shown to work well when the quantity of crop pest species is limited, but they become ineffective when various features are manually retrieved. They necessitate a further level of data preprocessing known as feature engineering, which is critical. In addition, its capacity to generalize across datasets is limited. Furthermore, their effectiveness depends on the available data; for example, limited dataset results in poor accuracy, but after a certain level of accuracy is achieved, increasing the dataset has little effect on performance. We have the same issues when it comes to insect classification.

DL models can solve these issues, particularly the CNNs DL algorithm, when the input dataset comprises images. DL is a type of ML that automatically uses multilevel neural networks to extract deep features. DL-based algorithms for weed identification [8]–[10], plant recognition [11], [12], and plant disease identification [13], [14] have increasingly been popular in agriculture. Insect categorization was another area where the CNN model outperformed traditional ML techniques. Several DL algorithms have been used to identify pests in recent years, yielding state-of-the-art results in various pest detection applications [15], [16].

The motivation behind the research study is that despite the wide range of studies on pest classification and recognition, there is still interest in developing high-accuracy automated systems for pest classification. Although a few studies on pest classification and recognition have recently been provided, this research subject remains underexplored. Transfer learning (TL) of pre-trained DL frameworks and support vector machines (SVM) is the most commonly utilized pest classification and recognition method in extant research. However, the SVM machine learning algorithm takes longer to train with larger datasets [17]. Overfitting and negative transfer are the most concerning limits in TL [18]. For this purpose, we developed the DeepPestNet model for pest identification to address these concerns in this research study. The proposed model consists of 11 learnable layers, eight of which are convolutional and three fully connected (FC).

The main contributions of the research study are:

- We introduced an effective DeepPestNet model for boosting insect classification performance by identifying insect species in field crops at an early stage, and this can be used to improve crop quality and yield.
- For accurate pest classification and recognition, the proposed new end-to-end DeepPestNet model extracts the most discriminative features automatically.
- We used the image rotations and augmentations technique to increase the size of training data and prove the generalizability of the proposed DeepPestNet model.
- For pest classification and recognition, we compared the performance of the proposed DeepPestNet model to that of Squeezenet and Googlenet deep TL methods.The rest of the paper is structured out as follows.

Section 2 describes the related work. Section 3 contained a description of the technique. Section 4 went over the specifics of the experiments as well as the outcomes. Finally, Section 5 brings our work to a close.

II. RELATED WORK

Recently, some research studies have been concentrated on pest classification and recognition. The pest classification and recognition research can be divided into ML, DL, and hybrid-based approaches. Hybrid methods include techniques that employ both DL and ML techniques. Many pest classification research works use DL-based techniques, whereas ML-based techniques are rarely used. Below, we highlighted the most recent and relevant research work in automatic pest identification and classification.

Recently, advanced ML-based techniques have effectively performed well in pest categorization and detection [19]–[21]. Multiple classifiers are trained using extracted features from pests, and multiple types of pest images were categorized in these works. In [22], the UAV dataset was used to forecast armyworm contaminated and healthy corn regions, and the armyworm occurrence levels were then categorized. The best combination of image features for recognizing armyworm insects in corn-planted regions was discovered utilizing Gini-importance. The authors compared four types of ML methods: Random Forest (RF), Multilayer Perceptron (MLP), Naive Bayesian (NB), and SVM. The RF model performs the best compared to other ML approaches (MLP, NB, and SVM) for classifying the armyworm pest and normal corn. The authors [23] proposed an automatic pest detection (thrips) system based on SVM for greenhouse pest attack monitoring. Researchers used a novel image processing technique to detect parasites on strawberry plants. The SVM approach with a different kernel function was applied for parasite classification and thrips identification. The SVM structure was designed using the main diameter to minor diameter ratio as a region index, Hue, Saturation, and Intensify as colour indices. The results demonstrate that utilizing the SVM method with area index and intensify as colour index produces the best classification, with a mean percent error of under 2.25 percent. In [24], the authors used two feature extraction techniques for the identification and categorization of tomato pests, namely, Histogram of Oriented Gradient (HOG) and Local Binary Pattern techniques (LBP). HOG outperforms its competitor, according to the comparison results. However, these ML-based pest recognition systems rely on 'handcrafted' features extracted from the actual domain by comparing individual appearances. As a result, the empirical parameters must be manually changed to account for changes in image acquisition conditions. As a result, detecting pests in outside environments using handcrafted techniques based on colour, structure, and texture remains a serious difficulty.

The importance of DL methods in computer vision has inspired researchers to utilize them for pest recognition and classification [25]–[27]. But unfortunately, DL algorithms in the domain of pest recognition have been constrained by a scarcity of pest image datasets and the inexplicability of DL frameworks. A novel and robust dataset for crop pest recognition was created in [15], and three different DL models were trained to employ TL and fine-tuning. The recognition rate of the three Architectures was greater than 80.00 per cent. Using the gradient-weighted class activation technique, the authors presented appropriate visual descriptions for the most crucial portions of the recognition layers. According to this study, the recognition process concentrates more on visual details than the entire image, and general differences are overlooked. An end-to-end pest detection system combining DL and hyperspectral imaging (HSI) techniques is presented in [16]. This technique can be used to quickly recognize pests for successful pest control. To address noise and duplicate details in the HSI spectral space, one-dimensional convolution and attention techniques across spectral channels are employed to develop a spectral feature extraction unit to effectively use spectrum information. The HSI feature extractor secures rich spectral-spatial information using a three-dimensional convolution branch structure with various resolutions in parallel. The output feature map maintains its higher resolution throughout its use. Each branch contains an adjustable spectral-spatial feature extraction unit that dynamically weights different inputs, limiting the HSI's disproportionate effect and improving the network's feature extraction skills. Pest HSI was acquired utilizing hyperspectral imaging equipment, resulting in a dataset containing nine different pests.

Furthermore, it is commonly known that the hybrid models (DL and ML models) produce better classification results, which can be used to classify insects. In [28], DL models (TL technique) were used to classify eight different types of tomato pests. Using DL models, the extracted pest features were merged with three ML classifiers, i.e., discriminant analysis (DA), SVM, and the k-nearest neighbour approach (KNN). Bayesian optimization was used to effectively tune hyper-parameters. Following image augmentation, the VGG16 framework performed better than the other algorithms. The ResNet50 with discriminant analysis classifier had the best accuracy among the CNN and ML frameworks. The authors [29] developed a new DL model TPest-RCNN for pest detection. The faster regional-convolutional neural network is used as a based model in the proposed model. Moreover, VGG16 was used for feature extraction. Then, a region of tiny pests was generated by a region proposal framework. Finally, extracted features and regions of small pests are fed to RoIAlign for classification and detection.

Following a review of prior research, it is discovered that DL models, particularly those used to categorize crop pests, cannot achieve superior identification and classification performance. Instead of using real-world scenes, most models were trained and tested using images captured in highly controlled lab environments. However, in the field, the complicated surroundings, various viewpoints and postures, varying degrees of colour and texture alteration, changes in lighting conditions, and different locations of pests' wings and limbs constitute a considerable barrier to pest recognition. Although few studies addressed automatic detection in natural settings (testing is performed on natural images), most of them focused on a single species or used only one dataset for validation purposes. Detecting pests effectively and quickly and extracting traits independent of viewpoint, scale, and lighting conditions are critical for crop pest recognition. This paper developed an effective DL framework for identifying and classifying crop pests into ten different classes. Also, the data set size is boosted using image rotation and data augmentation techniques to obtain generalized results. To test the generalizability of the proposed approach, we tested it on a different dataset with 9 different types of crop pests.

III. METHODOLOGY

In the realm of image processing and computer science, DL approaches have recently achieved considerable success (more specifically, pest recognition and classification). In this study, we have proposed the DeepPestNet DL model for the effective and efficient recognition and classification of

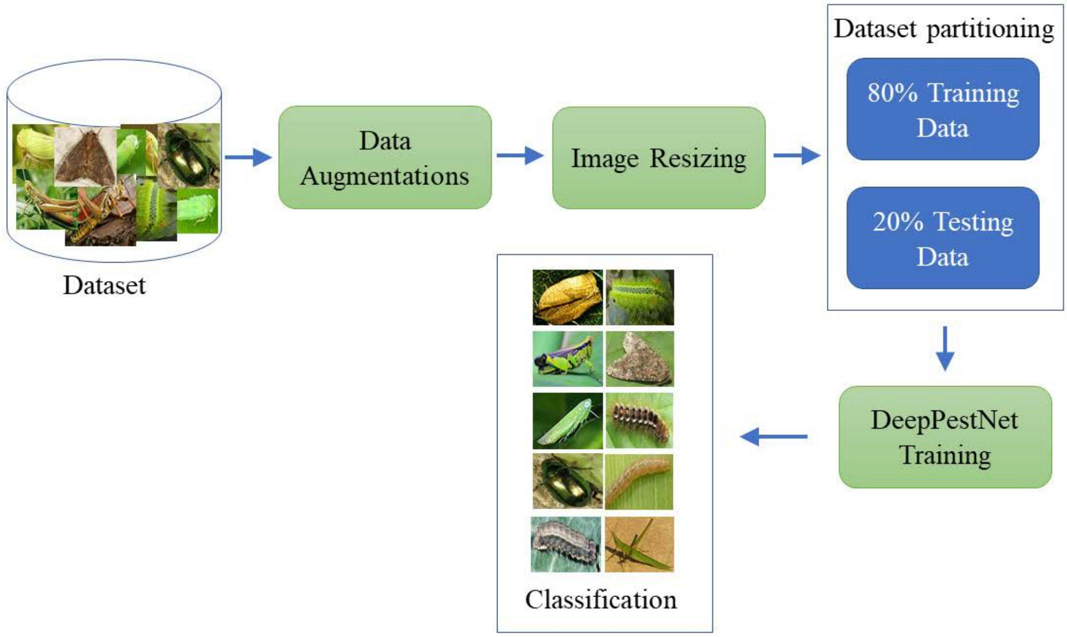


FIGURE 1. General workflow of the proposed system.

crop pests. We accomplished a ten-class classification of crop pests using (the Deng *et al.* 2018) dataset. The abstract view of the proposed DeepPestNet approach is shown in Fig. 1, which comprises four core steps. The dataset images rotations and data augmentations, image resizing, dataset splitting, and model training and testing. To further evaluate the classification performance of our method, we accomplished nine-class classifications by employing the proposed DeepPestNet model using the standard Kaggle "Pest Dataset" dataset to validate the efficacy of the model. Eleven learnt layers make up the proposed model, including eight convolutional layers and three FC layers. The details about the proposed approach are elaborated below.

A. DATA AUGMENTATIONS

The unavailability of a large amount of data for training the DL frameworks is one of the challenges when intending to use DL approaches to pest recognition and classification problems. More crop pest data is difficult and expensive to get, both in terms of time and resources. Data augmentation, or increasing the amount of available data without acquiring new data by applying multiple processes to current data, has been proven advantageous in image classification [30]. The ImageNet classifier challenge winners adopted this method [31], [32] and used it academically to improve training data and reduce overfitting [33].

Due to the limited number of images in the dataset, we applied image rotations, and data augmentation approaches in this study. For this purpose, we rotated all of the dataset's images (both training and testing) by 90 degrees twice. The dataset's image count was raised threefold through this image rotation procedure. Additionally, the images in the training set were rotated at a random angle between −20 and 20 degrees, arbitrarily translated up to thirty pixels vertically and horizontally. They randomly translated the images between [0.9 and 1.1] to create additional images.

It's also worth noting that the imageDataAugmenter function creates sets of augmented images dynamically during each training phase. The number of images in the training set was significantly expanded using this data augmentation method, enabling more effective use of our DL model by training with a much higher number of training images. Furthermore, the augmented images are only used to train the proposed framework, not to test it; hence, only real images from the dataset are utilized to test the learned framework.

B. IMAGE RESIZING

The input images in the datasets are of different sizes. To ensure uniformity and speed up the processing, we applied certain pre-processing to resize the input images to 224 × 224 pixels according to the input image requirements of our model.

C. DATASET PARTITIONIN

For each experiment, the dataset is separated into training and testing sets. More precisely, 90% of the dataset was used for model training, and 10% was used for testing.

D. DEEPESTNET ARCHITERCTURE DETAILS

This paper proposed the DeepPestNet DL model for pest recognition and classification. The architecture of the proposed framework is elaborated in Table 1. The proposed

DeepPestNet approach is deeper than standard CNN with eleven learnable layers, i.e., eight convolution layers followed by three FC layers. The architecture has a total of thirty-three layers, including one input layer, nine leaky relu (LR) layers, one relu layer, five maximum pooling layers, 6 batch normalization (BN) layers, one cross channel normalization layer, three dropout layers, four average pooling layers, a softmax layer, and a classification layer. The input layer of the proposed DeepPestNet model is the initial layer, and it accepts 224 × 224 input crop pest images for processing. The first convolution layer extracts the feature from the 224×224 input image by applying 64 kernels (filters) of size 7 × 7 with a stride of 2 × 2 to generate the feature map. The convolutional kernels help extract feature maps (patterns) by splitting the image into smaller parts. The kernels (with certain weights) move over the pest images and apply a dot product with the sub-region of input data to obtain the output feature maps. The working of the convolutional layer is expressed as:

$$f_c^k(m, n) = \sum_d \sum_{r,s} j_d(r, s) \cdot i_c^k(v, w) \tag{1}$$

f_c^k represents the output feature map, j_d (r, s) represents the pest input image multiplied by $i_c^k(v, w)$ index of the kth kernel of the cth layer. After employing convolutions on the input pest image, the size output is produced. Where *i* represents input, *p* means padding, *k* represents kernel size, and *s* represent steps.

Activation functions frequently precede convolutional layers. The activation function describes how a layer node translates the weighted sum of the input into an output. The first convolutional layer is followed by Rectified Linear Unit (relu) activation function (to improve model efficiency). Relu activation is used because it is efficient and straightforward. Relu works as follows:

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \leq 0 \end{cases} \tag{2}$$

LR activation functions follow the remaining convolutional layers (except the first layer). We used LR to define the ReLU activation function (RAF) as a tiny linear fraction of x instead of expressing it as 0 for negative input values (x<0). The following is how this activation function is calculated:

$$f(x) = 0.01 \times x, x \tag{3}$$

If the input is non-negative, this function yields x, but if it is smaller than 0 (negative), it yields 0.01 times x. Additionally, maximum pooling layers follow all convolutional layers to decrease the overall computational complexity. Pooling layers perform down-sampling to minimize the spatial size. This layer decreases the number of parameters and calculations, improving the architecture's efficacy and preventing over-fitting.

$$f(x) = \max(\mathrm{x1, x2, x3}, \ldots, \mathrm{xk}) \tag{4}$$

where $f(x)$ is a feature vector that has been optimized, Max-pooling is a down-sampling technique that employs a kernel (k) and a stride (s) to extract the highest value from a pest image with a size of h × w. The primary purpose of layering a max-pooling layer between the convolutional layers is to steadily shrink the size of the spatial representation, i.e., h and w, lowering the number of parameters to train and the network's ultimate computations. It also helps to avoid overfitting issues.

The first convolutional layer's output is passed to the second convolutional layer (after employing RAF, cross channel normalization, and max-pooling). The following convolutional layer uses 64 filters of 1 × 1. The third convolutional layer filters the inputs by employing 192 kernels of size 3 × 3 with a 1-pixel padding value. The fourth convolutional layer applies 512 kernels of size 3 × 3 to the inputs with a padding value of 1 pixel and a stride of 2 pixels. The fifth convolutional layer applies 384 kernels of size 3 × 3 with padding and a stride of 1 pixel to the inputs. The subsequent three convolutional layers, i.e., sixth, seventh, and eighth, apply 256 kernels of size 3 × 3 with a default stride and padding of 1 pixel and are not followed by the pooling layers. These layers also extract features, and these feature maps are submitted to FC layers. The nodes of each FC layer are all joined to the nodes of the higher layers. FC layers are utilized to translate the extracted two-dimensional feature map into a one-dimensional feature vector. Equation 5 represents the working of the FC layer.

$$a_i = \sum_{j=0}^{m\times n-1} w_{ij} \times x_i + b_i \tag{5}$$

where *i* denotes the index of the FC layer's output; *n, m, d,* and *i* denote the height, width, depth, and index of FC layers output. Furthermore, b and w denote the bias and weights, respectively. The LR and dropout layers (to prevent overfitting) come after the first two FC layers, while the softmax and classification layers come after the last fully connected layer. The output of the last FC layer is fed to a 10-way softmax as we have ten classes in our dataset.

E. HYPER PRAMETERS

The accuracy of DL frameworks is highly reliant on the choice of hyper-parameters [34], which are generally determined using a trial-and-error method. The selection of hyper-parameters is crucial because it determines how the algorithm works [35]. We examined the proposed DeepPestNet model performance using multiple hyper-parameters values to identify the ideal value for each hyper-parameter, given the numerous options for hyper-parameters. Table 2 demonstrates the details of the hyper-parameters that were chosen. The training of our DeepPestNet model was done using stochastic gradient descent (SGD). The framework is trained on 80 epochs for pest recognition and classification, with overfitting taken into account.

IV. RESULTS

We provide a detailed description of the findings of numerous studies conducted to determine the efficacy of our PestDetNet

TABLE 1. DeepPestNet architecture details.

Sr. No	Layer	Filters	Size	Stride	Padding
1	Input				
2	Convolutional-1 (Relu + Cross channel Normalization)	64	7×7	2×2	[3 3 3 3]
3	Max pooling		3×3	2×2	[0 1 0 1]
4	Convolutional-2 (LeakyRelu)	64	1×1	1×1	[0 0 0 0]
5	Convolutional-3 (LeakyRelu + BN)	192	3×3	1×1	[1 1 1 1]
6	Max pooling		3×3	2×2	[0 1 0 1]
7	Convolutional-4 (LeakyRelu + BN)	512	3×3	1×1	[2 2 2 2]
8	Max pooling		3×3	2×2	[0 0 0 0]
9	Convolutional-5 (LR)	384	3×3	1×1	[1 1 1 1]
10	Max pooling		3×3	2×2	[0 0 0 0]
11	Convolutional-6 (BN + LR)	256	3×3	1×1	[1 1 1 1]
12	Convolutional-7 (BN + LR)	256	3×3	1×1	[1 1 1 1]
13	Convolutional-8 (BN + LR)	256	3×3	1×1	[1 1 1 1]
14	Max pooling		3×3	2×2	[0 0 0 0]
15	FC + LR + Dropout				
16	FC + LR + Dropout				
17	FC + LR + Dropout				
18	Softmax				
19	Classification				

TABLE 2. Hyper-parameters of DeepPestNet architecture.

Parameter	Value
Optimization algorithm	SGDM
Shuffle	Every epoch
Maximum Epochs	80
learning rate	0.001
Verbose	false
Validation frequency	30
Train Size	0.8
Test Size	0.2

model. This section also includes more information regarding the dataset used in this study. Deng 1018 is used to evaluate the performance of our technique. We also tested our model's performance using the publicly available "Pest Dataset" to ensure that it is generalizable.

A. DATASET

To assess the performance of the proposed PestDetNet framework, we used the dataset presented in Deng *et al.* (2018) [42]. It comprises ten distinct pest categories primarily seen in tea plants and other plants throughout Europe and Central Asia. More specifically, the dataset's ten different pests images include Locusta migratoria, Euproctis pseudoconspersa Strand, Chrysochus Chinensis, Empoasca flavescens, Spodoptera exigua, larva of Laspeyresia pomonella, Parasa lepida, Acrida cinerea, larva of S. exigua, and L.pomonella. A sample representation of each pest type is shown in Fig. 2. The total number of pests images in the Deng data is 562. Each class of pests consists of 40 to 70 images. The number of images against each pest's category is shown in Table 3. Fig. 3 shows sample images after rotations. We rotated the images of the dataset twice by 90 degrees. The number of images against each pest's category after rotation is also mentioned in Table 3. The pest images in the dataset were gathered from Mendeley and other online sources and include pest images acquired with a Single Lens Reflex camera (SLR). The rest of the images were from Insert Images, IPM Images, Dave's Garden, and others. The dataset contains RGB images of different resolutions. The size, posture, angle, lighting conditions, and backgrounds of the sample images vary greatly.

B. EVALUATION MATRICS

The following assessment measures are used to assess the proposed method's performance: Accuracy, Precision, Sensitivity (Recall), specificity, and F1_score. The accuracy of the presented framework is given by equation 4, defined as "the number of correctly detected or classified images (COVID-19 or normal) to the total number of sample images". The precision of the proposed model is identified as "the number of correctly detected or classified images (COVID-19) to the total number of (COVID-19) positive images detected

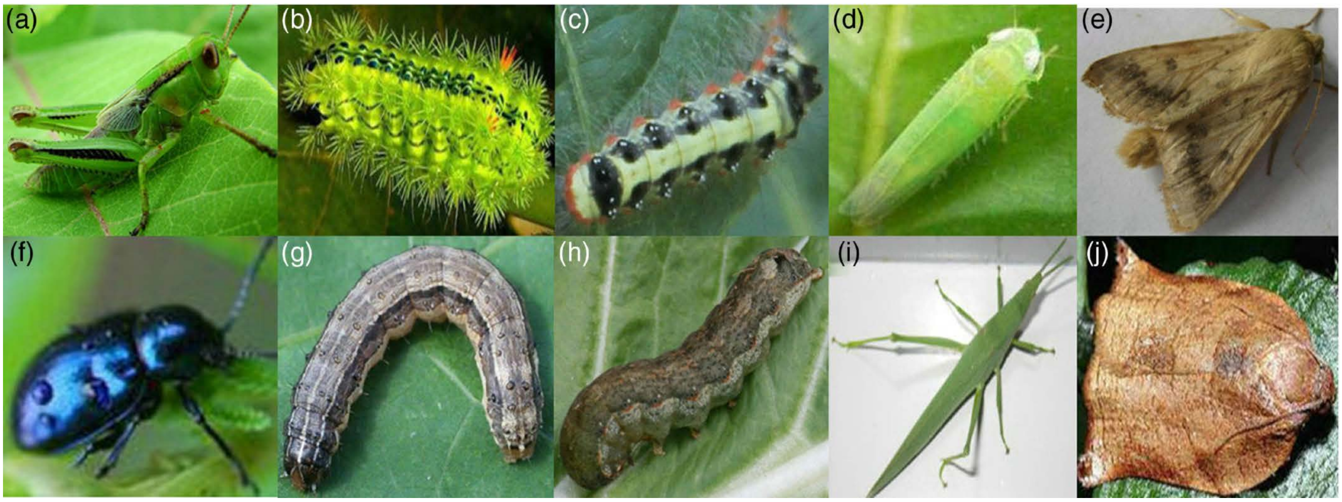


FIGURE 2. Sample images from Deng *et al.* dataset (a) Locusta migratoria, (b) Parasa lepida, (c) Gypsy moth larva, (d) Empoasca flavescens, (e) Spodoptera exigua, (f) Chrysochus chinensis, (g) Laspeyresia pomonella larva, (h) Spodoptera exigua larva, (i) Atractomorpha sinensis, (j) Laspeyresia pomonella.

FIGURE 3. Sample images from Deng *et al.* dataset after rotations.

TABLE 3. Details about Deng *et al.* data set for pests' classification.

Pests Category	Without rotations	After rotations
Locusta migratoria	72 (1)	216
Parasa lepida	59 (2)	177
Chrysochus chinensis	50 (6)	150
Spodoptera exigua larva	56 (8)	168
Atractomorpha sinensis	62 (9)	186
Empoasca flavescens	40 (4)	120
Laspeyresia pomonella	65 (10)	195
Spodoptera exigua	68 (5)	204
Gypsy moth larva	40 (3)	120
Laspeyresia pomonella larva	50 (7)	150
Total No. of Images	562	1686

(correctly or erroneously) by the model''. The recall is calculated as ''the number of correctly classified images (COVID-19) to the total number of COVID-19 images in the dataset''. Similarly, specificity is calculated as ''the number of correctly detected negative images (normal) to the total number of negative (normal) images in the dataset''. Whereas F1_score combines precisions and recall and calculates the weighted average of both. The equations to estimate these metrics are:

$$Accuracy = (TN + TP)/TS \tag{6}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{7}$$

$$Sensitivity\,(recall) = \frac{TP}{TP + FN} \tag{8}$$

$$Specificity = \frac{TN}{TN + FP} \tag{9}$$

$$F1\_score = 2 \cdot \frac{Precision \times Recall}{Precision + Recall} \tag{10}$$

TP, TS, FP, TN, and FN stand for true positive, total samples, false positive, true negative, and false negative, respectively.

C. EXPERMENTAL SETUP

The proposed method is tested and validated on a system with an Intel (R) Core (TM) i5-5200U processor and 8GB of RAM. The software and hardware resources used in the proposed PestDetNet method are shown in Table 4. Models require input images to be shrunk; thus, input images are resized accordingly. 90% of the images were used for training, while 10% were utilized for testing. For all the experiments, the training and testing sets are being used to train and test the proposed PestDetNet approach and other contemporary models using the same experimental settings for pest recognition and classification as those given in Table 2 for pest recognition and classification. A set of tests are carried out to assess the proposed PestDetNet framework for multiclass classification pest classification's classification performance.

D. PERFORMANCE EVALUATION ON PEST RECOGNITION AND CLASSIFICATION

This experiment aims to verify the usefulness and effectiveness of the proposed pest recognition and classification method. We used all 1686 (after rotations of original images) crop pest images of Deng *et al.* (2018) dataset in this experiment (1518 pest images for model training and the rest 168 images for model testing). Table 5 shows the total number of images used for training and testing in each category. Using the Deng *et al.* (2018) dataset, the proposed framework took 536 minutes and 46 seconds to train for pest classification. This time, however, is proportional to the number of epochs and iterations per epoch. The total number of iterations in the training stage for DeepPestNet is 1100 (11 iterations per epoch), and the number of epochs is 100. We also created a confusion matrix assessment to precisely explain the proposed technique's classification performance in terms of actual and predicted classes. The proposed DeepPestNet approach confusion matrix is shown in Table 6. It is concluded from the confusion matrix that the proposed

TABLE 4. Hardware and software configurations.

Software/Hardware	Content
Operation system	Windows 10
CPU	Intel (R) Core (TM) i5-5200U
Programming language	MATLAB
RAM	8GB
MATLAB version	R2020a
ROM	500GB

DeepPestNet system achieves the optimal results with a true-positive rate of 100% for all the pest classes in the Deng *et al.* (2018) dataset, indicating that the proposed DeepPestNet framework correctly classified all pest image samples. The loss function shows how well the DeepPestNet framework can predict the dataset. To assess the training performance of the presented method, we have demonstrated accuracy and loss in Fig. 4, elaborating that accuracy and loss after epoch ''55'' almost remain the same (approximately equal to 100%), which means we can obtain satisfactory results even at lower classification epochs. The proposed approach attained the ideal accuracy, precision, recall, specificity, and F1-score of 100%, demonstrating its effectiveness for multiclass classification of pest images.

To further identify the effectiveness and validity of the proposed approach, precise recognition and classification of many crop pests are required. For this purpose, we assess the usefulness of the proposed PestDetNet approach in identifying the class of each crop pest. Table 7 shows the precision, recall, and F1-score performance of the proposed PestDetNet approach in class-wise crop pest classification. The proposed technique provides state-of-the-art performance in terms of all evaluation parameters, as revealed in Table 7. The results show that all pests images are correctly classified, resulting in optimal accuracy. The robustness of the introduced DL model, which better reflects each class, is the critical cause for the improved pest recognition accuracy.

The difference in the initial number of received pest images in each image category of the dataset can be seen obviously in table 5 (class imbalance problem). This class imbalance potentially led to a range of issues, including overfitting, in which the method fails to generalize well to new datasets. Overfitting occurs when a model learns the details of the training dataset to the extent where it is unable to generalize appropriately; as a result, regularization approaches such as data augmentation are utilized in this study to combat overfitting. As a result, augmentation techniques are used to enhance the number of images of all pest classes to avoid the model from overfitting.

The proposed PestDetNet approach has ideal classification accuracy because the model uses convolutional layers with kernels of various sizes (7×7, 3×3, 1×1). It allows the network to learn different spatial patterns and recognize characteristics at various scales. 1×1 filters discover patterns throughout the depth of the input pest images. Whereas 3×3 and 7×7 filters learn spatial patterns over the input's three

TABLE 5. The number of training and testing samples of each pest category.

Class number	Pest name	After rotations	Training	Testing
1	Locusta migratoria	216	195	21
2	Laspeyresia pomonella	195	175	20
3	Parasa lepida	177	160	17
4	Gypsy moth larva	120	108	12
5	Empoasca flavescens	120	108	12
6	Spodoptera exigua	204	183	21
7	Chrysochus chinensis	150	135	15
8	Laspeyresia pomonella larva	150	135	15
9	Spodoptera exigua larva	168	151	17
10	Atractomorpha sinensis	186	168	18
	Total No. of Images	1686	1518	168

TABLE 6. Confusion matrix obtain by proposed framework utilizing Deng *et al.*, dataset.

Predicted class											
	Pest class	1	2	3	4	5	6	7	8	9	10
	1	21	0	0	0	0	0	0	0	0	0
	2	0	20	0	0	0	0	0	0	0	0
	3	0	0	17	0	0	0	0	0	0	0
	4	0	0	0	12	0	0	0	0	0	0
True class	5	0	0	0	0	12	0	0	0	0	0
	6	0	0	0	0	0	21	0	0	0	0
	7	0	0	0	0	0	0	15	0	0	0
	8	0	0	0	0	0	0	0	15	0	0
	9	0	0	0	0	0	0	0	0	17	0
	10	0	0	0	0	0	0	0	0	0	18

TABLE 7. Class-wise performance (%) of the proposed method.

Classs	Pest **name**	n(truth)	n(classified)	Accuracy	Precision	Recall	F1-score
1	Locusta migratoria	21	21	100	100	100	100
2	Laspeyresia pomonella	20	20	100	100	100	100
3	Parasa lepida	17	17	100	100	100	100
4	Gypsy moth larva	12	12	100	100	100	100
5	Empoasca flavescens	12	12	100	100	100	100
6	Spodoptera exigua	21	21	100	100	100	100
7	Chrysochus chinensis	15	15	100	100	100	100
8	Laspeyresia pomonella larva	15	15	100	100	100	100
9	Spodoptera exigua larva	17	17	100	100	100	100
10	Atractomorpha sinensis	18	18	100	100	100	100

dimensions (width, depth, and height). Hence, at different scales, different convolutional filter sizes learn various spatial patterns and extract distinctive features from pest images with better accuracy.

E. COMPARISION WITH STATE OF THE ART DEEP LEARNING MODELS

The key aim of this experiment is to validate the efficacy of the proposed PestDetNet technique for pest recognition and classification over the existing DL models. For this purpose, we compared the classification performance of the proposed PestDetNet framework with two pre-trained DL models, i.e., Squeezenet [36] and Googlenet [37]. The frameworks are trained on many images from the ImageNet database in a TL configuration. All networks' pre-trained versions can classify images into 1000 separate classes. The last three layers are fine-tuned to separate the crop pest images into ten classes. The image input size of the models varied, so we resized the pests images of the dataset according to the input image requirement of the models. We utilized the identical experimental set-up, as shown in Table 2, to tune the models as we did for the proposed PestDetNet model. We used all 1686 (after rotations of original images) crop pest images of the Deng *et al.* (2018) dataset for this experiment. Also, 1518 pest images are used for model training and the remaining 168 images for model testing. Table 5 shows the total pest images utilized for training and validation in each category. From the findings shown in table 8, it is

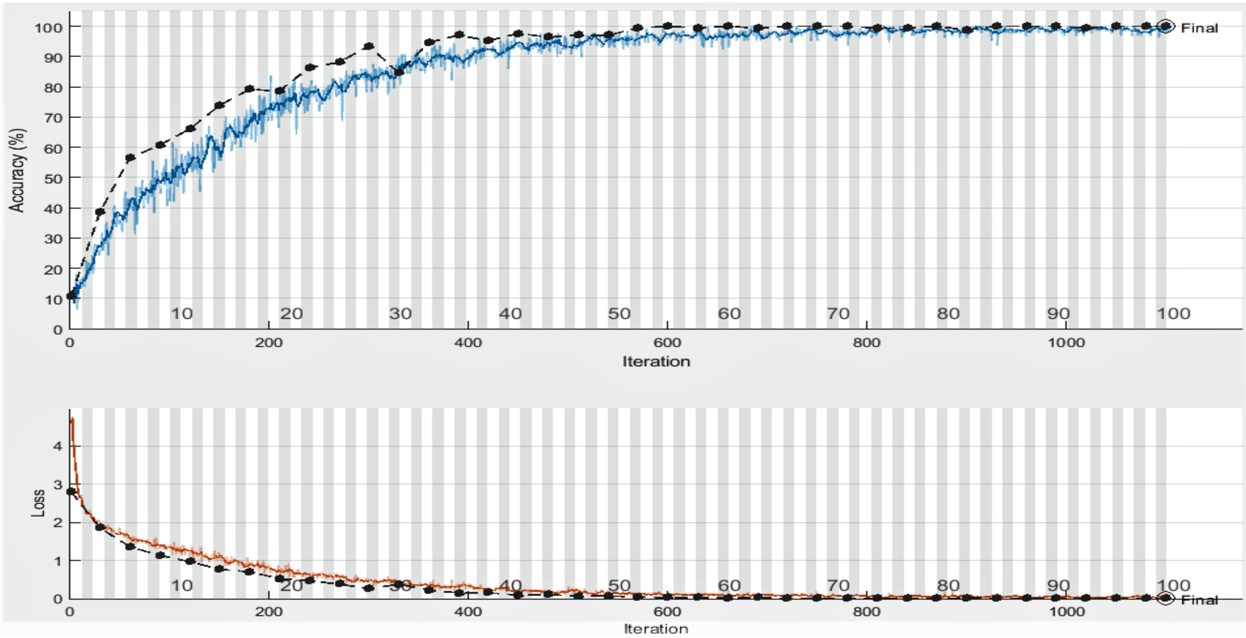


FIGURE 4. Training accuracy and loss of Deeppstnet model.

evident that SqueezeNet and Googlenet achieved the lowest performance results compared to the proposed PestDetNet model in terms of all performance metrics. It is essential to mention that both Squeezenet and Googlenet achieved the same classification results in all performance measures. Based on the results, we noticed that the proposed PestDetNet framework performed better than the other DL frameworks by achieving accuracy, precision, recall, and an F1 score of 100% for pest recognition and classification. The Squeezenet and Googlenet model has a lower accuracy than the proposed model since the RAF follows each convolutional layer in these models. The Relu sets all values smaller than (x<0), i.e., negative values, to zero for all neurons with negative values. There's no assurance that all neurons will be activated all of the time., which leads to the dying Relu problem. The model does not learn in this scenario because the optimization algorithm does not work. The dying ReLU problem is problematic because it causes a significant part of the network to become inactive over time. The proposed PestDetNet model uses the LR activation function instead of RAF to address the dying relu problem. Hence the PestDetNet model learns even in the presence of negative values and keeps all neurons activated. Furthermore, the proposed model applies BN after each convolutional layer. The significance of each feature is preserved, even though some feature has a higher numerical value than others. As a result, the proposed model will be completely unbiased (to higher-value features). In addition, as compared to a framework that does not use BN, the framework that uses this technique is trained faster and has higher accuracy.

TABLE 8. Pest recognition and classification performance comparison with state-of-the-art deep learning frameworks.

Model	Accuracy	Precision	Recall	F1-score
Googlenet	99.87	99.4	99.4	99.4
Squeezenet	99.87	99.4	99.4	99.4
Proposed model	100	100	100	100

F. PERFORMANCE EVALUATION ON PEST DATASET HAVING NINE CLASSES OF PESTS

To further evaluate and assess the performance and generalizability of the proposed DeepPestNet framework, we validated the model on another standard freely available Kaggle dataset, "Pest Dataset" [38]. The data comprises nine crop pests: Aphids, Armyworm, Beetle, Bollworm, Grasshopper, Mites, Mosquito, Sawfly, and Stem borer. The dataset is balanced and contains enough images for model training, and we have not performed rotations operation on the dataset. However, we applied data augmentations to verify the generalizability power of the proposed model. The dataset contains 350 images of each pest class, i.e., aphids, Armyworm, beetle, bollworm, grasshopper, mites, mosquito, sawfly, and stem borer pests. Also, an automatic script is used to scrape images of pests from Google through Selenium and Chrome Driver. We used all 3150 (without rotating the original images) crop pest images of the "Pest Dataset" dataset in this experiment (2520 pest images for model training and the rest 630 images for model testing). Using the "Pest Dataset" dataset, the proposed framework took 1380 minutes and 12 seconds to train for pest classification. This time is proportional to

TABLE 9. Class-wise performance (%) of the proposed method.

Classs	Pest **name**	n(truth)	n(classified)	Accuracy	Precision	Recall	F1-score
1	Aphids	35	35	100	100	100	100
2	Armyworm	36	35	100	100	100	100
3	Beetle	35	35	100	100	100	100
4	Bollworm	35	35	100	100	100	100
5	Grasshopper	35	35	100	100	100	100
6	Mites	34	35	99.68	97.0	100	99.0
7	Mosquito	35	35	100	100	100	100
8	Sawfly	35	35	100	100	100	100
9	Stem borer	35	35	100	100	100	100

the maximum number of epochs and iterations per epoch. The total number of iterations in the training stage for DeepPestNet is 1760 (22 iterations per epoch), and the number of epochs is 80. We achieved average accuracy, precision, recall, and F1-score of 99.96%, 99.66%, 100%, and 99.82, respectively. The accuracy of 99.96% proves the effectiveness and generalization ability of the proposed DeepPestNet model for pest recognition and classification.

We used an experiment to assess the usefulness of the proposed technique in identifying the class of each crop pest. Table 9 demonstrates the precision, recall, and F1-score performance of the presented approach in class-wise crop pest classification. The proposed technique provides state-of-the-art performance in terms of all evaluation parameters, as demonstrated in Table 9. Also, the results show that the proposed model misclassifies only one pest, i.e., one Mite pest is misclassified as Armyworm. The robustness of the proposed DL model, which better reflects each class, is the crucial cause for the improved pest recognition accuracy.

We achieved better results as the proposed DeepPestNet model captures more detailed features from the pest images. We used a tiny filter with a size of 3×3, which ensures that detailed features are extracted. The BN strategy utilized in the proposed model's feature map normalizes the inputs to each mini-batch, offers regularization, and lowers the generalization error. In addition, the dropout strategy utilized in the proposed model's classification unit enables regularization by removing a fraction of the preceding layer's outputs to avoid overfitting and promote generalization. These findings demonstrate the efficacy of the proposed pest recognition and classification approach.

G. COMPARISION OF THE PROPOSED DEEPPESTNET WITH STATE OF THE ART METHODLS

Furthermore, we experimented with comparing the proposed DeepPestNet and existing state-of-the-art pest recognition and classification methods to verify the proposed model's superiority. We compared the proposed approach to the most recent DL frameworks and presented the results in Table 10. For pest recognition, Nanni *et al.* [39], proposed ensembles of CNNs based on multiple architectures (ResNet50, EfficientNetB0, ShuffleNet, GoogleNet, DenseNet201, and MobileNetv2) optimized with various Adam versions. Two novel Adam algorithms based on DGrad for deep network optimization are proposed, each with a scaling factor in the learning rate. Six CNN models with different optimization functions are trained on the Deng (SMALL) dataset, big IP102, and Xie2 (D0) pest datasets. On the Deng dataset, the best scoring ensemble competed with domain experts' classifications. It achieved the best results on all three pest datasets: 95.52 per cent on Deng, 74.11 per cent on IP102, and 99.81 per cent on Xie2. In [40], the authors presented a deep CNN framework for classifying insects on three widely accessible insect datasets. The first insect dataset employed was the National Bureau of Agricultural Insect Resources (NBAIR), which comprises 40 classes of crop pest photos. In contrast, the second and third datasets (Xie1, Xie2) comprise 24 and 40 classes of pests, respectively. Data augmentation methods such as scaling, reflection, translation, and rotation are used to prevent the network from overfitting. To increase accuracy, the effectiveness of hyperparameters was investigated in the proposed model. The proposed CNN framework achieved the best accuracy of 96.75%, 97.47%, and 95.97% for the NBAIR insect dataset, Xie insect dataset, and Xie2 insect dataset, respectively. In [28], DL models were used to categorize eight different types of tomato pests. The extracted pest features are merged with three ML classifiers using the DL models, including discriminant analysis (DA), SVM, and k-nearest neighbour (KNN) approach. Bayesian optimization was used to tune hyper-parameters automatically. With an accuracy of 94.95 per cent after image augmentation, the VGG16 model performed better than the other models. The ResNet50 with DA framework achieved 97.12 per cent classification accuracy in the CNN + ML models.

This comparison also shows how successful the proposed DeepPestNet model is compared to other approaches. It's worth noting that these methods are more computationally expensive than the proposed because they use deeper frameworks, which can inevitably lead to overfitting. These findings indicate the effectiveness of the proposed technique and its additional advantages, such as computing efficiency. As the proposed DeepPestNet model only has eleven layers, followed by BN and the LR activation function, all CNN layers' biases are not active. As a result, we may say that the proposed DeepPestNet approach is more efficient and effective in identifying and classifying pests images.

TABLE 10. Pest recognition and classification performance comparison with state-of-the-art methods.

S. No	Work	Dataset	Method	Date	Accuracy
1	Loris Nanni et al., [39]	Deng et al.,	Ensemble CNN	2022	95.52%
2	K. Thenmozhi et al., [40]	Xie1 (24 classes)	Deep CNN model	2019	97.47%
3	Huang, M.L et al., [28]	IPM Images + NBAIR datasets	ResNet50 with DA	2022	97.12
4	Wang Dawei et al., [41]	Deng et al.,	TL of Alexnet	2019	93.84
5	Deng et al., [42]	Deng et al.,	Local Configuration Patterns plus SVM	2018	85.5
6	Proposed DeepPestNet approach	Deng et al.,	DeepPestNet model	2022	100%

TABLE 11. Accuracy rates on two medical datasets that further validate the performance of the DeepPestNet.

Dataset	Accuracy	Precision	Recall	F1-score
Breast Grading Carcinoma dataset	96.67	93.33	94.0	93.66
Laryngeal dataset	96.21	92.5	93.0	92.75

H. MORE EVALUATION OF DEEPPESNET APPROCH

Although the experiments show that the DeepPestNet model performed well on two pest datasets, more testing on various datasets in a different domain is required to prove the proposed model's strength, robustness, and universal adaptability. We tested the proposed DeepPestNet model on two medical imaging benchmarks that represented various categorization tasks. The first dataset is Breast Grading Carcinoma data set. This dataset covers breast carcinoma histology specimens acquired from Thessaloniki's "Agios Pavlos" General Hospital's Department of Pathology. The collection contains 300 annotated images with a resolution of 1280 × 960 that belong to 21 individual patients with grade 1-3 invasive ductal carcinoma of the breast (grouped in the corresponding folders). There are 107 images in grade 1, 102 images in grade 2, and 91 in grade 3. We used all 300 images of the dataset, where 270 images are used for model training, and the remaining 30 images are used for model testing. The Laryngeal dataset, which contains 1320 patches of photographs with a size of 100 × 100 pixels, is the second dataset. Tissue with interpupillary capillary loops (IPCL), tissue with leukoplakia (Le), tissue with hypertrophic vessels (Hbv), and healthy tissue are all categorized into four categories (He). There are 330 images in each class. The dataset is separated into three subfolders for cross-validation purposes. We utilized all 1320 images of the dataset, where 1188 images are utilized for framework training, and the rest 132 images are utilized for testing. In Table 11, the accuracy attained by the proposed DeepPestNet approach on the two medical datasets. According to the results (Table 11), the proposed DeepPestNet approach achieved satisfactory performance (accuracy greater than 96%) in the medical domain as well, which validates the effectiveness of the proposed approach.

V. CONCLUSION AND FUTURE WORK

Globally, contamination is the major cause of agricultural loss and financial loss. The detection and removal of exotic insects would be greatly accelerated if invading insects could be identified automatically. This paper presented a DeepPestNet framework for effective pest recognition and classification. The accuracy of 100% for pest recognition and classification has confirmed the superiority of the proposed framework over contemporary methods. Moreover, experimental results on the standard Kaggle dataset (Pest Dataset) and two datasets from the medical domain (Breast Grading Carcinoma dataset and Laryngeal dataset) have confirmed the effectiveness and robustness of the proposed framework for pest recognition and classification. However, only major pests are investigated in this study. There are many different types of insects, and there are distinctions between larvae and adults. For example, noctuid pests have only been identified during their most dangerous stage of development (i.e., the larval stage). In future, we are interested in expanding the classification size by including more pests types to be effectively identified by the proposed DeepPestNet framework. This research could help specialists and farmers identify pests more quickly and effectively, thus reducing economic and crop output losses.

REFERENCES

[1] R. Perroy, "World population prospects," *United Nations*, vol. 1, no. 6042, p. 587, 1966.
[2] V. Malathi and M. P. Gopinath, "Classification of pest detection in paddy crop based on transfer learning approach," *Acta Agriculturae Scandinavica, B Soil Plant Sci.*, vol. 71, no. 7, pp. 552–559, Oct. 2021.
[3] M. Martineau, D. Conte, R. Raveaux, I. Arnault, D. Munier, and G. Venturini, "A survey on image-based insect classification," *Pattern Recognit.*, vol. 65, pp. 84–273, May 2017.
[4] D. Brunelli, A. Albanese, D. d'Acunto, and M. Nardello, "Energy neutral machine learning based IoT device for pest detection in precision agriculture," *IEEE Internet Things Mag.*, vol. 2, no. 4, pp. 10–13, Dec. 2019.
[5] A. Albanese, D. d'Acunto, and D. Brunelli, "Pest detection for precision agriculture based on IoT machine learning," in *Proc. Int. Conf. Appl. Electron. Pervading Ind.* Cham, Switzerland: Springer, Sep. 2019, pp. 65–72.
[6] P. Boissard, V. Martin, and S. Moisan, "A cognitive vision approach to early pest detection in greenhouse crops," *Comput. Electron. Agricult.*, vol. 62, no. 2, pp. 81–93, Jul. 2008.
[7] T. Kasinathan, D. Singaraju, and S. R. Uyyala, "Insect classification and detection in field crops using modern machine learning techniques," *Inf. Process. Agricult.*, vol. 8, no. 3, pp. 446–457, Sep. 2021.
[8] X. Jin, Y. Sun, J. Che, M. Bagavathiannan, J. Yu, and Y. Chen, "A novel deep learning-based method for detection of weeds in vegetables," *Pest Manage. Sci.*, vol. 78, no. 5, pp. 1861–1869, May 2022.
[9] P. Saini, "Recent advancement of weed detection in crops using artificial intelligence and deep learning: A review," in *Advances in Energy Technology*. 2022, p. 631.
[10] A. Subeesh, S. Bhole, K. Singh, N. S. Chandel, Y. A. Rajwade, K. V. R. Rao, S. P. Kumar, and D. Jat, "Deep convolutional neural network models for weed detection in polyhouse grown bell peppers," *Artif. Intell. Agricult.*, vol. 6, pp. 47–54, Jan. 2022.

[11] A. Hassan, S. Islam, M. Hasan, S. B. Shorif, T. Habib, and M. S. Uddin, "Medicinal plant recognition from leaf images using deep learning," in *Computer Vision and Machine Learning in Agriculture*, vol. 2. Singapore: Springer, 2022, pp. 137–154.

[12] Z. Yang, W. He, X. Fan, and T. Tjahjadi, "PlantNet: Transfer learning-based fine-grained network for high-throughput plants recognition," *Soft Comput.*, vol. 5, pp. 1–10, Jan. 2022.

[13] J. Annrose, N. H. A. Rufus, C. R. E. S. Rex, and D. G. Immanuel, "A cloud-based platform for soybean plant disease classification using archimedes optimization based hybrid deep learning model," *Wireless Pers. Commun.*, vol. 122, no. 4, pp. 2995–3017, Feb. 2022.

[14] M. Z. U. Rehman, F. Ahmed, M. A. Khan, U. Tariq, S. S. Jamal, J. Ahmad, and I. Hussain, "Classification of citrus plant diseases using deep transfer learning," *Comput., Mater. Continua*, vol. 70, no. 1, pp. 1401–1417, 2022.

[15] K. Wang, K. Chen, H. Du, S. Liu, J. Xu, J. Zhao, H. Chen, Y. Liu, and Y. Liu, "New image dataset and new negative sample judgment method for crop pest recognition based on deep learning models," *Ecol. Informat.*, vol. 69, Jul. 2022, Art. no. 101620.

[16] Z. Xiao, K. Yin, L. Geng, J. Wu, F. Zhang, and Y. Liu, "Pest identification via hyperspectral image and deep learning," *Signal, Image Video Process.*, vol. 6, pp. 1–8, Jan. 2022.

[17] J. P. Zhang, Z. W. Li, and J. Yang, "A parallel SVM training algorithm on large-scale classification problems," in *Proc. Int. Conf. Mach. Learn. Cybern.*, vol. 3, Aug. 2005, pp. 1637–1641.

[18] W. Zhao, "Research on the deep learning of the small sample data based on transfer learning," in *Proc. AIP Conf.*, vol. 1864, no. 1, Aug. 2017, Art. no. 020018.

[19] S. Yousefi, F. Haghighian, M. Naghdyzadegan-Jahromi, and H. R. Pourghasemi, "Pest-infected oak trees identify using remote sensing-based classification algorithms," in *Computers in Earth and Environmental Sciences*. Amsterdam, The Netherlands: Elsevier, Jan. 2022, pp. 363–376.

[20] D. Wei, J. Chen, T. Luo, T. Long, and H. Wang, "Classification of crop pests based on multi-scale feature fusion," *Comput. Electron. Agricult.*, vol. 194, Mar. 2022, Art. no. 106736.

[21] L. Jiao, C. Xie, P. Chen, J. Du, R. Li, and J. Zhang, "Adaptive feature fusion pyramid network for multi-classes agricultural pest detection," *Comput. Electron. Agricult.*, vol. 195, Apr. 2022, Art. no. 106827.

[22] W. Tao, X. Wang, J. Xue, W. Su, M. Zhang, D. Yin, D. Zhu, Z. Xie, and Y. Zhang, "Monitoring the damage of armyworm as a pest in summer corn by unmanned aerial vehicle imaging," *Pest Manage. Sci.*, vol. 78, no. 6, pp. 2265–2276, Mar. 2022.

[23] M. A. Ebrahimi, M. H. Khoshtaghaza, S. Minaei, and B. Jamshidi, "Vision-based pest detection based on SVM classification method," *Comput. Electron. Agricult.*, vol. 137, pp. 52–58, May 2017.

[24] G. Pattnaik and K. Parvathi, "Automatic detection and classification of tomato pests using support vector machine based on HOG and LBP feature extraction technique," in *Progress in Advanced Computing and Intelligent Engineering*. Singapore: Springer, 2021, pp. 49–55.

[25] K. N. Dharmasastha, K. S. Banu, G. Kalaichevlan, B. Lincy, and B. K. Tripathy, "Classification of pest in tomato plants using CNN," in *Proc. Int. Conf. Metaheuristics Softw. Eng. Appl. (METASOFT)*, Bhubaneswar, India, 2022, pp. 1–10.

[26] T. Wang, L. Zhao, B. Li, X. Liu, W. Xu, and J. Li, "Recognition and counting of typical apple pests based on deep learning," *Ecol. Informat.*, vol. 68, May 2022, Art. no. 101556.

[27] D. J. A. Rustia, J. Chao, L. Chiu, Y. Wu, J. Chung, J. Hsu, and T. Lin, "Automatic greenhouse insect pest detection and recognition based on a cascaded deep learning classification method," *J. Appl. Entomol.*, vol. 145, no. 3, pp. 206–222, Apr. 2021.

[28] M.-L. Huang, T.-C. Chuang, and Y.-C. Liao, "Application of transfer learning and image augmentation technology for tomato pest identification," *Sustain. Comput., Informat. Syst.*, vol. 33, Jan. 2022, Art. no. 100646.

[29] W. Li, D. Wang, M. Li, Y. Gao, J. Wu, and X. Yang, "Field detection of tiny pests from sticky trap images using deep learning in agricultural greenhouse," *Comput. Electron. Agricult.*, vol. 183, Apr. 2021, Art. no. 106048.

[30] W. Li, D. Wang, M. Li, Y. Gao, J. Wu, and X. Yang, "Field detection of tiny pests from sticky trap images using deep learning in agricultural greenhouse," *Comput. Electron. Agricult.*, vol. 183, Apr. 2021, Art. no. 106048.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[33] S. C. Wong, A. Gatt, V. Stamatescu, and M. D. McDonnell, "Understanding data augmentation for classification: When to warp?" in *Proc. Int. Conf. Digit. Image Comput., Techn. Appl. (DICTA)*, Nov. 2016, pp. 1–6.

[34] H. Cui and J. Bai, "A new hyperparameters optimization method for convolutional neural networks," *Pattern Recognit. Lett.*, vol. 125, pp. 828–834, Jul. 2019.

[35] W. W. Tso, B. Burnak, and E. N. Pistikopoulos, "HY-POP: Hyperparameter optimization of machine learning models through parametric programming," *Comput. Chem. Eng.*, vol. 139, Aug. 2020, Art. no. 106902.

[36] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50X fewer parameters and <0.5 MB model size," 2016, *arXiv:1602.07360*.

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

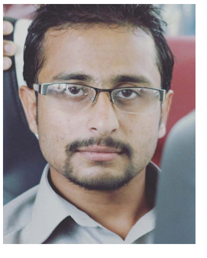
[38] *Pest Dataset*. Accessed: Jan. 11, 2022. [Online]. Available: https://www.kaggle.com/datasets/simranvolunesia/pest-dataset

[39] L. Nanni, A. Manfè, G. Maguolo, A. Lumini, and S. Brahnam, "High performing ensemble of convolutional neural networks for insect pest image detection," *Ecol. Informat.*, vol. 67, Mar. 2022, Art. no. 101515.

[40] K. Thenmozhi and U. S. Reddy, "Crop pest classification based on deep convolutional neural network and transfer learning," *Comput. Electron. Agricult.*, vol. 164, Sep. 2019, Art. no. 104906.

[41] W. Dawei, D. Limiao, N. Jiangong, G. Jiyue, Z. Hongfei, and H. Zhongzhi, "Recognition pest by image-based transfer learning," *J. Sci. Food Agricult.*, vol. 99, no. 10, pp. 4524–4531, Aug. 2019.

[42] L. Deng, Y. Wang, Z. Han, and R. Yu, "Research on insect pest image detection and recognition based on bio-inspired methods," *Biosyst. Eng.*, vol. 169, pp. 139–148, May 2018.



NAEEM ULLAH was born in Jafarabad, Charguli, Mardan, Khyber Pakhtunkhwa, Pakistan, in 1997. He received the B.S. degree in software engineering from the University of Engineering and Technology, Peshawar, in 2019, and the M.S. degree in software engineering from the University of Engineering and Technology, Taxila, in 2022.

He is the author of three articles. His research interests include software quality, image processing, image detection, image classification, image recognition, and deep learning. He was a recipient of the International Association of Geomagnetism and Aeronomy Young Scientist Award for Excellence, in 2008, and the IEEE Electromagnetic Compatibility Society Best Symposium Paper Award, in 2011.

JAVED ALI KHAN received the B.Sc. degree in computer software engineering from the University of Engineering and Technology, Peshawar, Pakistan, in 2009, the M.Sc. degree in software engineering from Bahria University, Islamabad, Pakistan, in 2013, and the Ph.D. degree in software engineering from Tsinghua University, China. He has been working as an Assistant Professor with the Department of Software Engineering, University of Science and Technology, Bannu, Pakistan, since December 2011. He has published more than 15 papers in reputable journals and conferences in requirements and software engineering. His research interests include requirements engineering, CrowdRE, argumentation and argument mining, feedback analysis, health analytics, empirical software engineering, sentiment analysis, and opinion mining.

LUBNA ABDULAZIZ ALHARBI received the B.Sc. degree in computer science from Taibah University, Saudi Arabia, in 2011, the M.Sc. degree in computer science from Coventry University, U.K., in 2015, and the Ph.D. degree in computer science from the University of Liverpool, U.K., in 2021. She is currently working as an Assistant Professor with the Department of Computer Science, College of Computers and Information Technology, University of Tabuk. Her research interests include artificial intelligence applications and algorithms, sentiment analysis and opinion mining, and machine and deep learning.

ASAF RAZA received the B.Sc. degree in computer software engineering from the University of Science and Technology, Bannu, Pakistan, and the M.Sc. degree in software engineering from the University of Engineering and Technology (UET), Taxila, Pakistan. His research interests include image and video processing, biomedical image processing, computer vision, machine learning, and deep learning.

WAHAB KHAN received the M.Sc. degree in electrical engineering from the COMSATS Institute of Information Technology, Pakistan. He is currently pursuing the Ph.D. degree in information and communication engineering with the Beijing Institute of Technology, China. Since 2013, he has been a Lecturer with the Department of Electrical Engineering, University of Science and Technology, Bannu, Pakistan. His research interests include UWSNS, wireless channel measurement and modeling, and wireless sensor networks.

IJAZ AHMAD received the M.E. degree in computer applied technology from the Changchun University of Science and Technology, China, in 2020. He is currently pursuing the Ph.D. degree in pattern recognition and intelligent systems with the Shenzhen Institute of Advanced Technology (SIAT), University of Chinese Academy of Science (UCAS), China. He has published several research articles under a wide area of research topics, including machine learning and deep learning (ML/DL) for biomedical diseases (EEG signals and systems, cancer, brain tumor, and COVID-19) and cybersecurity, network security, block chain, software-defined networking (SDN), and the Internet of Things (IoT).