EMITTER International Journal of Engineering Technology
Vol. 14, No. 1, June 2026, pp. 1~23
DOI : 10.24003/emitter.v14i1.984

An Attention based Vision Transformer for the Detection of Insect Pests in Castor Crop

Nitin[1], Satinder Bal Gupta[2], Pankaj Kumar Tyagi[2], Ravi Yadav[2], Amit Kumar Singh[1], Ajay Kumar[3], Shiv Kant[4]

[1]Amity Institute of Information Technology, Amity University, Rajasthan, India
[2]Indira Gandhi University, Rewari, Haryana, India
[3]S.O.E.T., Raffles University, Neemrana, India
[4]Greater Noida Institute of Technology (GNIOT), Greater Noida
Corresponding Author: satinderbal@igu.ac.in

Received February 2, 2026; Revised March 3, 2026; Accepted May 5, 2026

Abstract

Castor (Ricinus communis L.) is a significant crop valued for its non-edible oil, yet its economic importance is compromised by insect pests causing substantial yield losses of 35-40%. This paper explores the efficiency of utilization of vision transformers for efficient pest classification. We propose CASTIPestViT, a Vision Transformer-based model specifically designed for insect pest detection in castor crops. The model integrates transfer learning and fine-tuning mechanisms, leveraging a pre-trained Vision Transformer (ViT) initially trained on ImageNet1k, and is fine-tuned on a custom dataset of castor insect pests. CASTIPestViT uses the self-attention mechanism of ViTs to capture global and local features of insect pests. The performance of CASTIPestViT is compared with six different pre-trained CNN models. The results obtained by the proposed model achieve a validation accuracy of 97.60% in insect pest detection and outperforming other state-of-the-art models in terms of precision, accuracy, and f1-score. The model offers a robust solution in early-stage insect pest detection to reduce yield losses. The efficiency and accuracy of the model make it suitable in sustainable crop management and smart agriculture systems for yield optimization.

Keywords: Castor, Image Processing, Vision Transformer (ViT), Smart Agriculture, Pest Detection.

1. INTRODUCTION

Castor (Ricinus communis L.), a member of the Euphorbiaceae plant family, holds significant economic value primarily for its non-edible oil derived from its seeds. This oil finds widespread use in various industries, including biofuel production, as well as in medicine and as biomass for Eri silkworms. Globally, farmers cultivate approximately 1.30 million hectares of land for castor. Despite its economic importance, castor cultivation faces numerous challenges, particularly in tropical and subtropical regions, including climate variations, diseases, and pests. Among these, insect pests pose a significant threat to castor plant growth, leading to estimated yield losses of 35-40% of total production [1]. More than 100 species of insect pests are reported by researchers that are responsible for castor plant infestation, but some of them are very major and responsible for high yield losses. The modification in genes for plant hybridization and blind/uncontrolled uses of pesticides to control the pests may lead to some serious environmental hazards as well as affect the quality of castor production. Therefore, precise identification of insect pests is the basis for stopover and prevention to improve the quality of yield and ecological protection. Toward this, the traditional methods for insect pest detection with naked-eye field investigation require high manpower and continuous observation. The manifestations of insect pests are random, and regional in space, uncertainty in insect forms makes it difficult to predict, categorize and control.

The image-based recognition methods of insect pests mainly include traditional recognition, instrumental-based, and pattern recognition. Traditional manual recognition methods require agricultural experts to analyze the crop growth and damage made by insect pests. But low efficiency and subjectivity are the main limitations of traditional methods [2]. The second method is instrumental-based identification, which uses various portable detectors. It has already proved that it is capable of confirming exact identification quickly and accurately from the different samples. However, manual operations with a low degree of automation are the main limitation of this method [3]. The third method is related to machine learning and pattern recognition which use mathematical calculation for classification. It helps to automatically classify the unseen data on behalf of previous data analysis [4]. Different machine learning models such as support vector machine, random forest, and decision tree utilized by the researchers for pest identification [5]. However, the difficulty in feature extraction is the main bottleneck with these methods. The continuous advancement of artificial intelligence for smart agriculture adopts deep learning methods for image classification in large-scale datasets. In this regard, Tassis et al. [6] use different Convolutional Neural Network (CNN) networks for the automatic identification of insect pests in coffee crops with the help of semantic and instance segmentation. It helps to classify pests by extracting the features of pest images through the different layers repeatedly. The deep learning approaches have become the most appropriate methods for image recognition. Since then, various research has already been published that used different CNN architectures to beat the predecessor outcomes such as Visual Geometry Group (VGG) [7], Region-based CNN (R-CNN) [8], EfficientNet [9], etc. Throughout the recent advancement, CNN architecture becomes deeper in training layers, which increases the requirement for computing resources [10]. In pest classification with deep learning, the convolution process performs template matching, where the same template filtering is applied to different parts of the image. However, the convolution procedure represents local actions between neighboring pixels. It is very difficult for a regular CNN model to encode the position and orientation of pests in images having complex backgrounds. This limitation of the CNN model can be overcome by a vision transformer (ViT) [11]. An enhanced version of the vision transformer algorithm to detect insect pests in peanut crops proposed by the authors. The authors used small as well as large tokens/patches for processing. Broad trial with this methodology demonstrates more impressive results than other state-of-the-art CNN algorithms [11]. Chougui et al. [12] conducted a study on plant village datasets for different disease classification problems. It used Convolutional Block Attention Module (CBAM), CNN, and ViT for disease classification. In this experiment, the performance of vision transformation outperforms the other deep neural networks. Training vision transformers from scratch is one of the challenging tasks. In this regard, Peng and Weng [13] introduced a hybrid vision transformer based on transfer learning and fine-tuning for IP102 insect pest detection. It helps to overcome the limitation of the existing convolution neural network used for pest recognition. It combines transformation with convolution block for pest recognition with pre-trained features. The transfer learning and fine-tuned procedure provide promising results in image recognition tasks. As per our best knowledge with the investigation of broad literature in agriculture development, very few of the studies are related to insect pest recognition. The research in the related field covers various crops such as orange, rice, lychee, peanuts, and many more, but there is no related literature found that covers insect pests of castor crops. The identification of various limitations and research gaps prompted us to develop a novel model for detecting insect pests in castor crops. The contributions of this study are summarized as follows:

- A Castor Pests dataset was collected through on-site observations in South Haryana, India, from August 2021 to March 2022, utilizing various cameras and smartphones. The dataset was meticulously curated with the assistance of agricultural experts for subsequent analysis.
- A hybrid data augmentation methodology was employed to enhanced the robustness of the dataset.
- A Vision Transformer based classification framework specifically optimized with transfer learning and fine-tuning approaches was proposed for classification of insect pests.
- An explainable AI framework using class activation mapping techniques was also incorporated to visualize the model attention regions and improve interpretability of pest classification decisions.

The structure of this paper is organized as follows: Section 2 provides an overview of relevant research on images recognition, Section 3 describes the details of the dataset, architecture, and methodologies employed in our research, while Section 4 presents the experimental results and comparisons with existing methods. Section 5 offers an in-depth discussion of findings, and finally, the paper concludes with insights into future research directions.

2. RELATED WORKS

The early identification of insect pests in crops is very critical for cheerful yield growth. The traditional machine vision approaches depend on time-consuming manual feature extraction. In recent years, object recognition and detection approaches based on CNN have been used by researchers for agricultural image analysis. Before making a discussion about the targeted deep learning method for object detection, the primary state-of-the-art recognition methods have been discussed in this section. The broad literature is divided into two subsections. Subsection 2.1 is related to the Convolution neural network used for pest detection, and subsection 2.2 explores the vision transformer used for different state of art applications.

2.1. CONVOLUTION NEURAL NETWORK:

Pest and disease detection with CNN in plants is an active research area that shows immense contribution. In the last few decades, many researchers have proposed various deep-learning methods for insect pest detection in varieties of crops. A basic setup for 15 categories of beetle species classification was made by the researchers for rice, corn, and wheat crops [14]. The proposed setup used a beetle elytra image dataset that consisted of different insect pest classes that generally affect grain crops. In all of these, 15 classes of beetles are selected by the authors for the targeted experiment. After image pre-processing, VGG16 architecture was selected for classification. Due to the limited availability of beetle images, the authors used the transfer learning method for model training. In this procedure, 5 starting blocks of convolution are set frozen, and the remaining blocks are used for gradient training. The proposed model used 4 fully connected heads for pest classification. After the whole setup and training, it achieved 83.8% of overall accuracy for beetle classification. Monitoring of pests is one of the important steps that is directly related to crop production. Light traps for pest monitoring are one of the traditional and popular methods but are very time-consuming and labor-intensive. An automatic light trap system has been developed by researchers for pest monitoring in rice crops [15]. To make this task possible a machine vision algorithm based on a residual network was proposed for accurate classification. The authors proposed ResNet18 and ResNet50 for the final setup. In this research, ResNet18 consisted of three output classes including 2 classes of rice planthopper species. On the other hand, ResNet50 consisted of four output classes including 3 moth species. This proposed model achieved a 90.3% precision ratio for pest classification. In another study, Dong et al. [16] proposed AlexNet-based neural network architecture for pest and disease detection in strawberry crops. The goal of this study was to improve the overall performance of the existing system with some state-of-the-art methods. The authors adopted transfer learning to accelerate the training time of the model and fine-tune it for parameter reduction. After the training, the model achieved 97.35% top 1 accuracy in categorizing 9 different classes. To make this step possible, it takes 363.6 seconds to train. Muppala and Guruviah, [17] proposed an automatic trap for insect pest detection in paddy crops. The trap used three light sources for pest image collection such as ultraviolet, blue, violet, and green. The authors proposed a four-layer neural network architecture that has contrast enhancement, noise removal, and segmentation phases. The proposed neural network also used a rescue optimization algorithm to improve the training accuracy in pest classification. The proposed model achieved 98.29% accuracy followed by ResNet50 and AlexNet. Zhu et al. [18] developed an automated system to classify insect pests in lychee crops. The proposed system used a knowledge graph for data collection and feature extraction. After the feature extraction of the first phase, VGG16 was used for accurate classification. This system consisted of 10 different types of lychee pest categories with 94.9% accuracy after 100 epochs. Dongmei et al. [19] proposed a deep-learning neural network based on inceptionv3 for citrus pest classification. The authors used self-collected data belonging to five pest classes such as moth, blowing scale, Starscream, citrus fruit fly, and beetle for this experiment. The proposed combination of inceptionv3 architecture successfully classifies citrus insect pests with 96.81% accuracy.

2.2. VISION TRANSFORMER:

After the successful implementation of the attention mechanism in NLP, it is utilized with computer vision that entirely changed the conventional network. In recent times, the attention mechanism for the classification of insect pests has been widely used by researchers. The conventional methods have been fully investigated through different approaches used for insect pest detection in varieties of crops. To address the insect pests with these existing methods is not sufficient due to the similar shapes and changeable appearances of pests. In this regard, Li et al. [20] proposed the SAFFPest network based on self-attention feature fusion to detect the exact category of pests in rice crops. This model used a deformable convolution module to extract primary features from images. It helps to increase feature extraction capability with different changeable shapes. In the second step, the model introduced a self-attention mechanism to refine the multiple feature maps. Subsequently, the proposed model replaced batch normalization with group normalization to reduce the impression of batch size in training. After the combination of all these strategies, the final model achieved impressive results and produced 97.3% mAP to classify rice insect pests.

The vision transformer is used by many researchers to perform multiple types of tasks because it reflects higher performance than its CNN competitors. The capability of the vision transformer to connect patches and analyze deep features led the vision transformer to be deployed in classification tasks. Zhang et al. [21] applied a hybrid vision transformer for orange disease and pest classification. The authors used DenseNet and the attention mechanism to classify six different categories of diseases and pests. This experiment achieved an accuracy of 96.90% for accurate classification. Peng and Wang,

[13] proposed a vision transformer-based hybrid model for insect pests' classification. The proposed system used the IP102 benchmarking dataset for model training. The proposed model combined the convolution block with the transformer and followed two levels of feature extraction. At the first level, the convolutional blocks are used for primary feature extraction from the input images. In the second level, the attention head of the transformer is used by the researchers for spatial feature extraction. To make this task possible data augmentation and transfer learning were also utilized by the researchers for better training. The final proposed model sufficiently classified different pest categories and achieved accuracies of 75.83% and 74.89% for 480 × 480 and 224 × 224 pixels of images. In another research, Berka et al. [22] proposed a hybrid deep learning-based image processing model named CactiViT for the detection of infection in cactus plants. This research presented a new dataset of cochineal infestation related to the different locations in Morocco. The proposed vision transformer model achieved 88.62% accuracy for 20% of test data, 89.25% accuracy for 25% of test data, 88.70% accuracy for 33% of test data, and 88.34% accuracy for 50% of test data. The authors also developed a mobile application with the help of a proposed model that demonstrated the health status of the cactus plant's images captured by the users.

The investigation reflected different CNN and vision transformer architectures used for insect pest classification and other agriculture applications. In this, transfer learning with fine-tuning also demonstrates remarkable results for classification tasks. It motivates the researchers to tackle the castor insect pest's classification and proves that a transfer learning-based vision transformer with fine-tuning can distinguish these additional classes with considerably higher accuracy. To the best of our knowledge, there is no similar method that was previously used by any researchers for castor insect pest classification.

3. ORIGINALITY

The research makes a significant and original contribution by developing a ViT-based model for castor insect pest detection, backed by a primary dataset, robust augmentation method, and comprehensive performance benchmarking. The study presents a novel application of vision transformer for the classification of insect pests in castor crops. The prior study for pest detection has utilized Convolutional Neural Network (CNN)-based models on different crops such as rice, lychee, peanuts, etc. As per the best of our knowledge and literature survey, this is the first known application of ViT for castor-specific insect pest detection. This work also contributed to a new dataset of castor insect pests comprising six major classes that were collected and labeled under the collaboration of agricultural experts. The study addresses the limitation due to dataset size. To do so, the study implements a hybrid data augmentation technique combining multiple transformations. The work also integrated transfer learning (ImageNet 1K) with fine-tuning. The fine-tuning adopted the retraining of the developed model with the CASTIPest dataset. The trained model benefitted from generalized visual features and specific pest-related features. This dual-stage learning approach empowered the model to boost classification performance. The performance of CASTIPestViT outperforms the other benchmarking model evaluated across different metrics. The proposed model also demonstrated its effectiveness in handling complex agricultural backgrounds and visual differences among insect pest species.

4. SYSTEM DESIGN

4.1. DATA SET DESCRIPTION AND AUGMENTATION

The image dataset analyzed in this paper is one of the key contributions of this work and was collected in the region of North Haryana, India. Most of the collected images are captured by onsite observation in daylight by electronic camera devices from the castor's farm. As we already discussed, the same pests showed different features due to their life cycle. The dataset collected was aggregated and labeled manually by the agriculture expert during the period of different stages. The agriculture expert classified the pests into 6 categories which showed a severe impact on overall production we already discussed above. The selected categories of insect pests included castor semilooper (Achaea Janata), whitefly (Trialeurodes ricini Misra), thrips (Retithrips syriacus (Mayet)), leaf miner (Liriomyza trifolii Bergess.), bihar hairy caterpillar (Spilosoma obliqua Wlk), and leaf hooper (Empoasca flavescens (Fabr.)).

Table 1. Dataset Description

S.N.	Scientific name	Common name of insect pest	Count	Augmented dataset
1	Achoea janata L.	Castor Semilooper	54	864
2	Spilosoma oblique Wlk.	Bihar hairy caterpillar	84	1332
3	Empoasca flavescens(Fabr.)	Leafhopper	49	780
4	Retithrips syriacus(Mayet)	Thrips	69	1100
5	Trialeurodes ricini Misra	Whitefly	72	1136
6	Liriomyza trifolii Bergess	Leaf miner	44	704

Despite the availability of a moderate number of images in the dataset for model training, a hybrid data augmentation strategy was adopted by the authors. The adopted strategy used a manipulation-based data augmentation process for new image generation [23]. It uses a combination of image enhancement, noise addition, rotation, and scaling operations on images. The final dataset consists of a total of 5916 images of insect pests for model training. The description of the CASTIPest dataset is shown in Table 1. The original castor pest images had different capturing conditions and complex backgrounds. The 6 different types of castor insect pests of the final dataset are shown in Figure 1 with their famous names.

Figure 1. Sample Images of Final Dataset

4.2. VISION TRANSFORMER

Transformers were first utilized in Natural Language Processing for translation tasks. It used a self-attention mechanism to focus on more relevant tokens in natural language processing. This capability of attention mechanism was also used by the researchers for CNN architecture. The implementation of transformers in computer vision relies on an impressive performance as compared to other state-of-the-art CNN architecture.

The overview of the pure ViT [24] is demonstrated in Figure 2 and adopted for castor insect pests' classification. The conventional transformer used for NLP takes a 1D input sequence of the token, where inputs are used as a token embedding. To work with 2D images, it replaces the token embedding with patch embedding and reshapes the image ($I \in R^{h \times w \times c}$) into 2D patches ($p \times p$), where h is the height of the image, w is the width of the image and c is the number of channels in the image, () is the height and width of the image portion(patch). The total number of patches is related to the image size and patch size as shown at equation (1).

$$P_n = hw/(p \times p) \tag{1}$$

Where p_n is the number of patches, hw is the resolution of the input image, $(p \times p)$ is related to the patch resolution. The sequence of patches is then flattened and performs linear projection. The output of linear projection is patch embeddings shown by equation (2).

$$z_0 = \left[I_{class}; I_p^1 E, I_p^2 E, I_p^3 E, \ldots, I_p^N E\right] + \left[E_{pos}\right] \tag{2}$$

I_{class} represent class token, I_p^i, where i = 1,2,3…N represents the image patches, E represents embedding, and Epos represents the position of patches.

The output of the patch embedding block is used by the transformer encoder as an input. The transformer encoder is shown in Figure. 2 using the residual connection and implemented by adding Layer Normalization, Multihead Self Attention, and Multilayer Perceptron block. The input of Mulihead Self Attention (MSA) and Multilayer Perceptron (MLP) is followed by the normalization function residual connection shown by equations 3 and 4.

$$z_l' = MSA(\text{LNorm}(z_{l-1})) + z_{l-1} \tag{3}$$
$$z_l'' = MLP(\text{LNorm}(z_{l-1}')) + z_{l-1}' \quad , \tag{4}$$

where l = {1,2,3…}, related to input layers. LNorm is the layer normalization function, equation (3) uses MSA operation on layers followed by layer normalization and forming an output with skip connection of the same features. Equation (4) shows MLP wrapping a layer normalization operation and performing an addition for skip connection.

$$y = LNorm(z_0^L) \tag{5}$$

In equation (5), y represented a set of sequences of output produced by the decoder and followed the layer normalization function.

In the transformer, multi-head attention helps to boost the overall performance of the self-attention layer. For example, in the case of a word, when processing a sentence self-attention head layers focus on specific positions without affecting the attention of other positions represented in different subspaces at the same time. For this, each head transforms the input vector into 3 distinct vectors- Query(Q), Key(K), and Value(V). Subsequently, the attention function is shown in equation (6) and demonstrated in Figure. 1 is used to calculate the degree of attention for each pair.

$$\text{Attention (Q,K,V)}= Softmax\left(\frac{Q.K^T}{\sqrt{d_k}}\right).V \tag{6}$$

Equation (6), first computes the scores of different vectors, normalizes them to enhance gradient stability, translates the normalized value into probabilities with the help of the softmax function, and at the final step the value vector(V) multiplied with the sum of probabilities, the vector with higher probabilities represent additional focus.

Figure 2 demonstrated the proposed ViT-based classification framework where a pretrained ViT model initialized with ImageNet weights is employed as the backbone. The traditional classification head is replaced by task specific dense layer corresponding to the pest categories. In the proposed system the lower encode layers are frozen for visual representation, while the higher and classification layers are fine tuned using the targeted dataset.

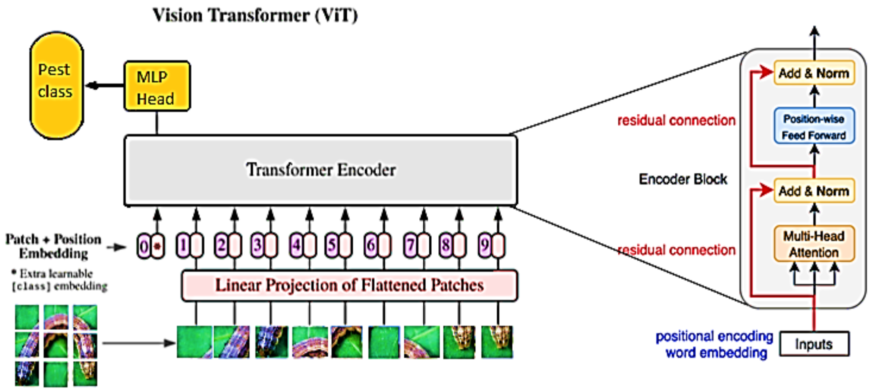

Figure 2. Vision Transformer

4.3. FINE TUNING WITH TRANSFER LEARNING TECHNIQUE

Vision transformer for pest classification is a very deep network that consists of different neural network blocks. Other than this, the quantity of the original dataset is also small which may lead to overfitting in real-life scenarios. To solve this limitation, we applied data augmentation and fine-tuning process with transfer learning for our model. Fine-tuning allows the neural network to transfer the pre-trained features of some task into another task and allows retraining of the head as well as the entire network for the required task [25]. In this research, we used pre-trained features of a vision transformer that is trained on a benchmark named ImageNet dataset [26]. The training of the proposed model is divided into two stages: the first stage follows the training of vit on the ImageNet dataset, and the Second step follows a refinement process with fine-tuning that concludes insect pest recognition. The step first of implementation allows ViT to adopt a broad category of features from generic imaging. The second stage is refined by the ViT to perform specific tasks. The nature of these state-of-the-art procedures makes this data suitable for castor's insect pest classification.

4.4. EigenCAM

Eigen Class Activation Mapping (EigenCAM) helps to enhance the model interpretability and transparency. In this regard, GradCAM is a popular method where Grad stand for Gradient-weighted and CAM associated with

Class Activation Mapping [27]. It helps to interpret the decision-making behavior of the Vision Transformer (ViT) model to generate class-discriminative localization maps. Grad-CAM utilizes the target class gradients flowing into a selected intermediate layer and produces a coarse localization map that highlights the regions of interest in the input image responsible in final prediction. But GradCAM have some limitations like it rely on gradient based backpropagation which leads to produce noisy and unstable attention map. In this regard, EigenCAM helps to show the perfect attention behavior. EigenCAM utilizes principal component analysis (PCA) on feature activations to identify dominant spatial patterns responsible in classification [27].

It extracts feature maps from a selected intermediate transformer block and compute principal eigenvector of the activation covariance matrix. The final projection captures the direction of most significant variance in feature space which effectively highlights the spatial regions contributing most strongly to the model output. In this inference the input image first preprocessed using the same pipeline applied during the model training. The heatmap produced by the EigenCAM computed using the predicted class output and corresponding feature activations.

5. EXPERIMENT AND ANALYSIS

The initial step of this work consisted of training and testing of benchmarking models over the generated dataset. All the models were trained by exploiting the fine-tuning technique, which allows feature transfer of pre-trained models and retraining network weights according to the targeted dataset.

After the data acquisition and generation, the input size of the images has been rescaled by $224 \times 224 \times 3$ by the ImageNet training procedure. Vision Transformer is applied for the initial feature extraction, where fine-tuning using transfer learning used by the ViT for initial feature extraction and targeted data is used for the final classification task. In the first step, image partitioning that split an image into 16×16 non-overlap patches was performed. This operation returns 196 total patches against one image. Next, the linear projection with position embedding is performed that returns a dimension of (768, 196), where 768 represents the total number of pixels in one patch ($16 \times 16 \times 3 = 768$) and 196 is related to the total number of patches in one image. In the second step, the position information of each patch is embedded with the patch embedding procedure. This procedure aims to process an image patch as per its position. In the third step, the embedding information is processed by the transformer encoder block is executed 12 times and helps in deep feature extraction. The transformer-encoder block contains two components named multi-head self-attention and MLP. This block also used normalization and skip connection as a performance enhancer. At the final stage, a head of MLP is used for classification.

All the experiments of this paper were carried out on the Google Colab platform, with the configuration shown in Table 2. Other than these, the Cuda version-11.8.89 parallel computing platform has been adopted for the efficient communication between CPU-GPU.

Table 2. Hardware Specification

Name	Specification
Python	3.10.11
PyTorch	2.0.1+cu118
GPU	Tesla T4
RAM	12GB
CPU	@2.30 GHz

The experiment has been used manipulation-based data augmentation (discussed above), and the targeted dataset split into training, validation, and testing datasets at the ratio of 60:20:20. All the aforementioned models were fine-tuned with 10 epochs with Adam optimizer [28] and a learning rate of 0.003 with a weight decay ratio of 0.3 and a batch size of 32 has been selected for training. Since insect pests are a multiclassification problem, the cross-entropy loss function has been used.

During the fine-tuning phase, the training dataset was used to update model's weights using the Adam optimizer with the specified learning rate, weight decay, batch size, and epoch configuration. On the other hand, the validation dataset was used to monitor model generalization and prevent model overfitting problem during fine-tuning. At last, the testing dataset was strictly reserved for final performance evaluation after completion of training.

5.1. DIFFERENT STATE OF THE ART METHODS

In image classification tasks, CNN is the most prominent method to achieve efficient results. In recent years, CNN noticed remarkable performance for image classification in agriculture. In this section, we discussed some state-of-the-art CNN networks due to their deep consideration in insect pest classification in a variety of crops [29] and used them as a benchmarking model for this research.

- **VGG Network:** In the ImageNet competition in 2014 a network introduced by Simonyan and Zisserman produced favorable performance [30]. This proposed model was introduced into two versions named VGG 16 and VGG19. In VGG16, 13 layers from the beginning are related to the deep feature extraction, and the last three layers are associated with fully connected layers used for classification. At the same in VGG19, 16 layers were for feature extraction, and the last 3 layers are associated with the fully connected layers. Different research made by the researchers demonstrates that it achieved good results in insect pests' classification [31-33].
- **ResNet:** ResNet is also named a residual network. It introduced residual blocks that make connections across the layers. In this, skip connection is one of the most unique features that are responsible for maintaining

global features. It helps in managing gradient decent problems cleverly and boosts the learning speed of the network. Undoubtedly it is one of the groundbreaking research in the field of deep learning [34]. Different versions of ResNet such as ResNet18, ResNet50, and RestNet101 have been utilized by the researchers for insect pests' [35].

- **InceptionV3:** The first version of inception (GoogLeNet) was proposed in 2014 which shows remarkable performance in the image classification tasks. But number of feature channels and high training time were some of the main bottlenecks for implementation. In this regard, Szegedy et al. [36], introduced an enhanced version of this architecture named inceptionV3. It allows the network to scale up in a way that it can utilize added computation efficiently. It uses multiple 1x1 convolution kernels to limit the feature channels. The feature of the inception block comprises different sizes of convolution filters making it suitable to scale input images in different sizes and resolutions. It shows remarkable performance in the field of insect pest classification, where images comprise information on different sizes, quality, and resolution [37, 38].

5.2. RESULTS AND DISCUSSION

This section demonstrates the performance of the proposed vision transformer network for the identification of castor insect pests and compares the achieved results with the state-of-the-art neural network. All the experiments performed in this work used the CASTIPest dataset. As a performance indicator, the authors selected some evaluation metrics such as F1-Score, Precision, Recall, Accuracy, and confusion matrix. These evaluation metrics are explained as follows:

- Accuracy: It is a metric that informally indicates the fraction of the total right prediction of the model and is evaluated with the equation (7):

$$Accuracy = \frac{\text{Total number of correct prediction Accuracy (TN+TP)}}{\text{Total no.of prediction(TP+TN+FP+FN)}} \quad (7)$$

- Precision: It is an indicator that shows the quality of the correct prediction made by any model. It returns a fraction of the true positive value from the total correct prediction. It is indicated by the equation (8):

$$Precision = \frac{Correctly\ identified\ samples(TP)}{Total\ number\ of\ correct\ prediction(TP+FP)} \quad (8)$$

- Recall: It indicates the true positive ratio of any model. It returns a fraction of correctly identified values and is represented by equation (9):

$$\text{Recall} = \frac{correct\ identified\ values(TP)}{total\ no.of\ correct\ prediction(TP+FN)} \quad (9)$$

- F1-Score: It takes recall and precision for consideration to calculate performance measurements. It represents the harmonic mean of recall and precision stated by equation (10):

$$\text{F1-Score}=2\times \frac{precision\times recall}{precision+recall} \quad (10)$$

- Confusion Matrix: It is one of the most descriptive matrices used to obtain category-wise statistical measurements of classifiers. It demonstrates correctly identified as well as incorrectly identified categories. It reports TP (true positive- total number of actual categories identified correct class), TN(true negative: total number of negative class correctly identified as negative), FP(false positive- total number of the negative class identified in positive class), FN(false negative represented incorrectly identified negative class). It is generally utilized for classification problems having multiple classes. Figure 3 shows the basic structure of the confusion matrix.

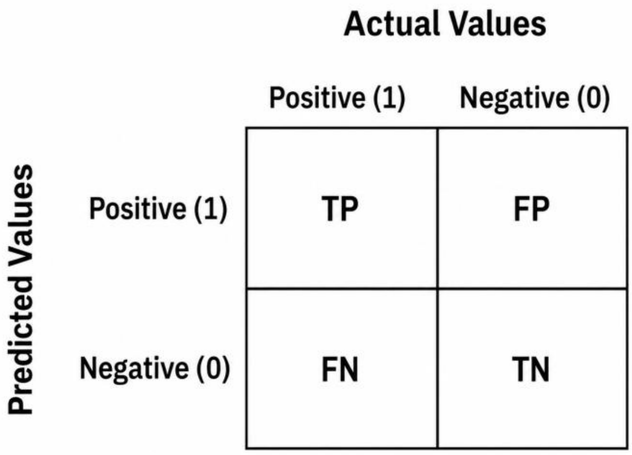

Figure 3. Basic structure of the Confusion matrix

To corroborate the effectiveness of the proposed model for pest classification the targeted dataset was divided into training, validation, and testing set in a proportion of 7:2:1. Secondly, six different state-of-the-art pre-trained models such as ResNet18, ResNet50, ResNet101, VGG16, VGG19, and inceptionV3 also experiment to classify the insect pests on the CASTIPest dataset, which means these models are also tested on the targeted dataset for comparison. In this regard, Table 3 shows validation and training accuracy with associated loss reflected by the experiment. The accuracy on the training dataset is increased after every epoch till it reaches 96.91%, 96.50%, 97.21%, 83.19%, 83.49%, 61.5%, and 99.19% for ResNet18, ResNet50, ResNet101, VGG16, VGG19, InceptionV3, and CASTIPestViT. Along with this, the accuracy of the validation set is also increased after every epoch till it reaches 94.76%, 92.55%, 89.73%, 79.46%, 77.34%, 79.76%, and 97.60% for ResNet18, ResNet50, ResNet101, VGG16, VGG19, InceptionV3, and CASTIPestViT respectively.

Table 3. Comparison in terms of Training Accuracy, Validation Accuracy, Training Loss, and Validation Loss

Model	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
ResNet18	96.91%	94.76%	0.1011	0.179
ResNet50	96.50%	92.55%	0.1026	0.2139
ResNet101	97.21%	89.73%	0.0856	0.3137
VGG16	83.19%	79.46%	0.4925	0.6755
VGG19	83.49%	77.34%	0.4834	0.6817
InceptionV3	61.5%	79.76%	1.0915	0.6124
CASTIPestViT	99.19%	97.60%	0.0229	0.0775

In Figure 4, the subfigures (a, c, e, g, i, k, m) show the accuracy curve of the training and validation dataset achieved by the ResNet18, ResNet50, ResNet101, VGG16, VGG19, InceptionV3, and CASTIPestViT in 10 epochs. Whereas the subfigures (b, d, f, h, j, l, n) show the loss curve achieved by these models in 10 epochs. A value of 0.1011 and 0.179 achieved by ResNet18, 0.1026 and 0.2139 achieved by ResNet50, 0.0856 and 0.3137 achieved by ResNet101, 0.4925 and 0.6755 achieved by VGG16, 0.4834 and 0.6817 achieved by VGG19, 1.0915 and 0.6124 achieved by InceptionV3, and 0.0775 and 0.0229 achieved by CASTIPestViT for training and validation dataset. The comparison of accuracy and loss curves based on fine-tuned models demonstrates that the aforementioned models achieve good accuracy and loss ratio for castor insect pest classification. In this, RestNet18, and RestNet50 achieved more than 90% validation accuracy ratio but suffered from curve fluctuation. Like the same, RestNet101, VGG16, and VGG19 achieved moderate accuracies such as 89.73%, 79.46%, and 77.34% but suffered with the same curve fluctuation problem. The training of InceptionV3 shows that it achieves the lowest validation accuracy as well as the highest training loss. But it also performs smooth training with the lowest curve fluctuation. On the other hand, the proposed CASTIPestViT achieves the best validation accuracy among all, it is obtaining the highest validation accuracy of 97.60%, and the lowest validation loss of 0.0775 with a smooth training curve.

Furthermore, Table 4 performs a category-wise comparison between the state-of-the-art and proposed model in term of precision, recall, and f1-score. The results demonstrated that the proposed vision transformer for castor insect pest classification achieves superior performance among all state-of-the-art models. We observed that the pest category of castor semilooper, Thrips, and whitefly achieves a value of higher than 90% in terms of precision, f1 score, and recall for proposed, ResNet101, ResNet50, and ResNet18.

Table 4. Performance comparison of the CASTIPest dataset in terms of precision, recall, and f1-score between fine-tuned models and CASTIPestViT

Pest Category	**Criterion**	**Model**						
		ResNet18	ResNet50	ResNet101	VGG16	VGG19	InceptionV3	Proposed
Castor Semilooper	Precision	0.98 %	0.94%	0.96%	0.96	0.87	0.90	1.00
	Recall	0.98%	0.96%	0.93%	0.85	0.92	0.87	0.99
	F1-Score	0.98%	0.95%	0.95%	0.90	0.89	0.89	0.99
Leaf Hopper	Precision	0.91%	0.92%	0.81%	0.56	0.66	0.93	0.96
	Recall	0.83%	0.68%	0.81%	0.56	0.55	0.52	0.93
	F1-Score	0.87%	0.78%	0.81%	0.56	0.60	0.66	0.94
Leaf Miner	Precision	0.87%	0.83%	0.94%	0.70	0.76	0.93	0.95
	Recall	0.91%	0.98%	0.86%	0.69	0.82	0.63	0.99
	F1-Score	0.89%	0.90%	0.90%	0.70	0.79	0.75	0.97
Thrips	Precision	0.95%	0.95%	0.93%	0.74	0.70	0.66	0.95
	Recall	0.90%	0.90%	0.91%	0.75	0.81	0.96	0.96
	F1-Score	0.92%	0.93%	0.92%	0.75	0.75	0.78	0.96
Whitefly	Precision	0.93%	0.94%	0.97%	0.90	1.00	0.88	1.00
	Recall	1.00%	1.00%	1.00%	0.93	0.85	0.89	0.98
	F1-Score	0.96%	0.97%	0.98%	0.92	0.92	0.89	0.99
Biharhairy	Precision	0.97%	0.96%	0.87%	0.75	0.76	0.89	0.98
	Recall	0.94%	0.92%	0.96%	0.82	0.75	0.91	0.98
	F1-Score	0.96%	0.94%	0.92%	0.78	0.76	0.90	0.98

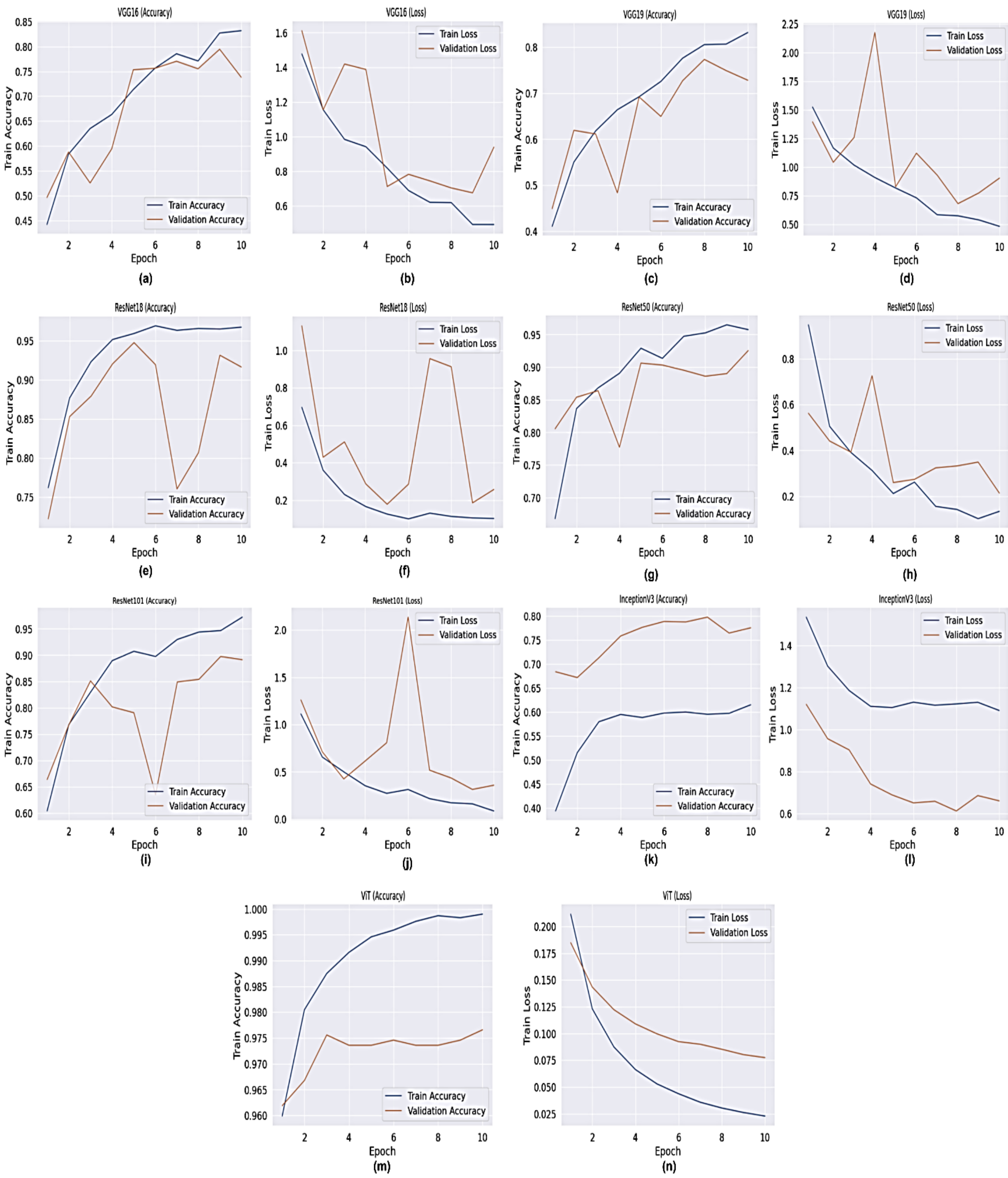

Figure 4. The Accuracy (a, c, e, g, i, k, m) and Loss(b, d, f, h, j, l, n) curve of ResNet18, ResNet50, ResNet101, VGG16, VGG19, InceptionV3 and CASTIPestViT

The category of leaf miner, leaf hopper, and thrips showed poorer performance matrices due to their similar features. These categories have lots of similar features in terms of color and structures that make it difficult for the classifiers to classify accurately. In experiment we also observe that Leafhoppers are small-sized pests with high pose variation and often appear partially occluded. Leaf miner damage patterns are primarily internal leaf texture distortions rather than distinct pest body features, making visual discrimination more challenging. In this regard, Figure 5 demonstrated the confusion matrix of the aforementioned models that are experimented on 6 different pest categories.

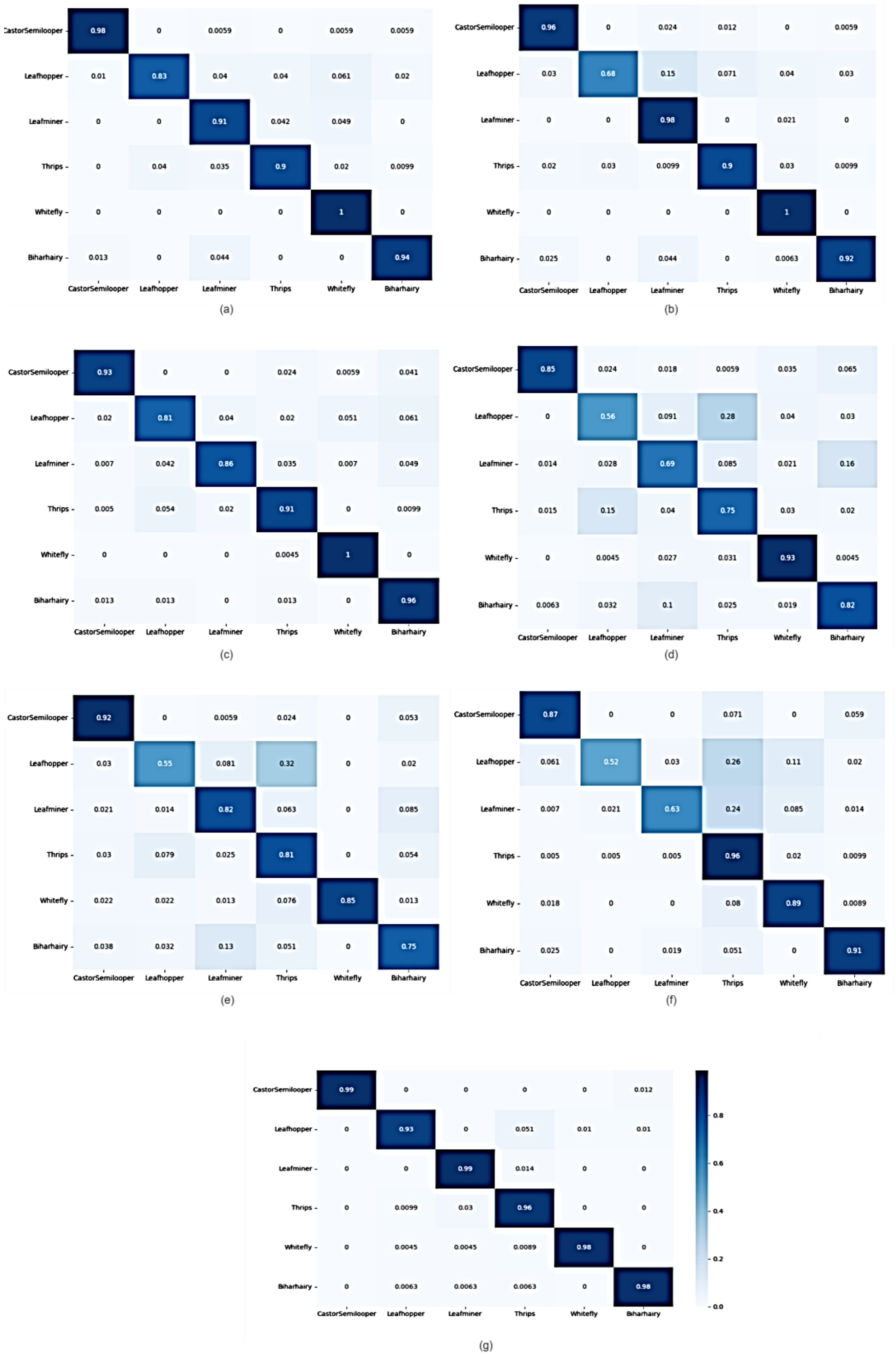

Figure 5. The confusion matrix of (a) ResNet18, (b) ResNet50, (c) ResNet101, (d) VGG16, (e) VGG19, (f) InceptionV3 and (g) Proposed(CASTIPestViT)

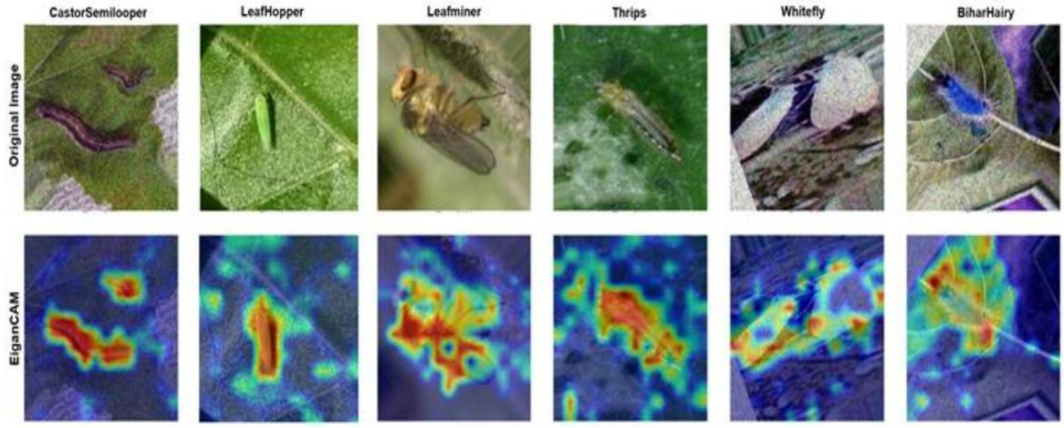

Figure 6. Original Pest Images and Interpretation using EigenCAM

We also apply EigenCAM to explain the focus of model's working on image information with respect to each type of insect pests. This gave us a more holistic summary of the main and secondary features that led to our model predictions in heatmaps. As shown in Figure 6, the generated heatmaps clearly demonstrated that the model focused on biological relevant of features of each pests. In the regard, the class of Castorsemilooper highlighted larval body structure including segmented form and color variations. In case of Leafhopper, the attention was concentrated on the body and wing contours. Similarly, for Whitefly and Biharhairy Caterpillar, the model emphasized distinct structural characteristics such as wing regions, and body texture. In the case of Leafminer or Thrips, the model highlighted the surrounding leaf damage patterns and pest's body. These visual explanations confirmed that the proposed Vision Transformer based model effectively learn discriminative spatial features and focuses on meaningful regions rather than irrelevant background information. Overall, EigenCAM enhances the interpretability of the ViT model with clearly providing of localized attention maps. It promote transparency and trust in model's decision-making process for insect pest classification.

In this experiment, several CNN models are used by the authors for castor insect pest classification and recognition. Several scholars have utilized this CNN architecture for different crop datasets. According to the results achieved by the aforementioned models, the proposed fine-tuned vision transformer outperforms the different CNN models. The models follow two topologies of feature extraction. The initial features of the models are achieved by transfer learning and the second level of features for the targeted dataset is extracted with the help of fine-tuning training. By analyzing and comparing the results above, the proposed model achieved the highest precision, recall, and f1-score. The comparison between training and validation accuracy also demonstrated that the proposed vision transformer model enhances accuracy for castor insect pests classification.

Furthermore, we know that validation on external datasets is necessary to evaluate cross-domain generalization capability. The real agricultural conditions are dynamic, with variations in lighting conditions, camera devices, growth stages, and pest density distributions. In our future work, we will focus on evaluating the model generalization and robustness using external benchmark pest datasets and cross location agricultural imagery. Additionally, the future research will explore deployment of proposed model in mobile based application for pest detection in different conditions.

6. CONCLUSION

This research demonstrated a transfer learning and fine-tuning-based vision transformer for classifying insect pests in castor crops. However, the preparation of insect pests dataset was one of the essential steps of this work to achieve optimal results. To do so, the authors used a hybrid data augmentation framework to enrich the targeted dataset. The automatic extraction of features in deep learning is an undeniable advantage. This task is also enhanced by the attention mechanism in training procedure for accurate classification of insect pests. The transfer learning approach also help to increase the applicability of this model. The training phase involves two stages: initial feature extraction from the pre-trained ImageNet1k dataset and fine-tuning using a vision transformer. The performance of the suggested deep learning model demonstrated superior results as compared to other benchmarking models. It also highlights the potential of vision transformers in agricultural pest detection, with future efforts focused on enhancing model generalization through hybridization techniques.

REFERENCES

[1] K. L. Manjunatha, P. C. Ganiger, and C. R. J. Basha, **Population dynamics of pests infesting castor and their natural enemies in Southern Karnataka**, *J. Entomol. Zool. Stud.*, vol. 7, no. 1, pp. 238–243, 2019.

[2] B. Dan, X. Sun, and L. Liu, **Diseases and pests identification of Lycium barbarum using SE-MobileNet V2 algorithm**, *12th International Symposium on Computational Intelligence and Design (ISCID)*, Hangzhou, China, pp. 121–125, 2019. https://doi.org/10.1109/ISCID.2019.00034

[3] T. Stackhouse, S. L. Boggess, D. Hadziabdic, R. N. Trigiano, M. D. Ginzel, and W. E. Klingeman, **Conventional gel electrophoresis and TaqMan probes enable rapid confirmation of thousand cankers disease from diagnostic samples**, *Plant Dis.*, vol. 105, no. 10, pp. 3171–3180, 2021. https://doi.org/10.1094/PDIS-10-20-2258-RE

[4] A. Fauzi, I. Syarif, and T. Badriyah, **Development of a mobile application for plant disease detection using parameter optimization method in convolutional neural networks algorithm**, *EMITTER International Journal of Engineering Technology*, vol. 11, no. 2, pp. 192–213, 2023. https://doi.org/10.24003/emitter.v11i2.808

[5] M. Chithambarathanu and M. K. Jeyakumar, **Survey on crop pest detection using deep learning and machine learning approaches**, *Multimed. Tools Appl.*, 2023. https://doi.org/10.1007/s11042-023-15221-3
[6] L. M. Tassis, J. E. de-Souza, and R. A. Krohling, **A deep learning approach combining instance and semantic segmentation to identify diseases and pests of coffee leaves from in-field images**, *Comput. Electron. Agric.*, vol. 186, p. 106191, 2021. https://doi.org/10.1016/j.compag.2021.106191
[7] N. Dong, Q. Feng, J. Chang *et al.*, **White blood cell classification based on a novel ensemble convolutional neural network framework**, *J. Supercomput.*, 2023. https://doi.org/10.1007/s11227-023-05490-y
[8] F. Deng, W. Mao, Z. Zeng, H. Zeng, and B. Wei, **Multiple diseases and pests detection based on federated learning and improved Faster R-CNN**, *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–11, 2022. https://doi.org/10.1109/TIM.2022.3201937
[9] S. CK, J. CD, and N. Patil, **Cardamom plant disease detection approach using EfficientNetV2**, *IEEE Access*, vol. 10, pp. 789–804, 2022. https://doi.org/10.1109/ACCESS.2021.3138920
[10] L. Alzubaidi, J. Zhang, A. J. Humaidi *et al.*, **Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions**, *J. Big Data*, vol. 8, p. 53, 2021. https://doi.org/10.1186/s40537-021-00444-8
[11] P. Venkatasaichandrakanth and M. Iyapparaja, **Pest detection and classification in peanut crops using CNN, MFO, and EViTA algorithms**, *IEEE Access*, vol. 11, pp. 54045–54057, 2023. https://doi.org/10.1109/ACCESS.2023.3281508
[12] A. Chougui, A. Moussaoui, and A. Moussaoui, **Plant-leaf diseases classification using CNN, CBAM and Vision Transformer**, *5th International Symposium on Informatics and its Applications (ISIA)*, M'sila, Algeria, pp. 1–6, 2022. https://doi.org/10.1109/ISIA55826.2022.9993601
[13] Y. Peng and Y. Wang, **CNN and transformer framework for insect pest classification**, *Ecol. Inform.*, vol. 72, p. 101846, 2022. https://doi.org/10.1016/j.ecoinf.2022.101846
[14] L. Wu, Z. Liu, T. Bera *et al.*, **A deep learning model to recognize food contaminating beetle species based on elytra fragments**, *Comput. Electron. Agric.*, vol. 166, p. 105002, 2019. https://doi.org/10.1016/j.compag.2019.105002
[15] Y. Qing, F. Jin, T. Jian *et al.*, **Development of an automatic monitoring system for rice light-trap pests based on machine vision**, *J. Integr. Agric.*, vol. 19, no. 10, pp. 2500–2513, 2020. https://doi.org/10.1016/S2095-3119(20)63168-9
[16] C. Dong, Z. Zhang, J. Yue, and L. Zhou, **Automatic recognition of strawberry diseases and pests using convolutional neural network**,

Smart Agric. Technol., vol. 1, p. 100009, 2021. https://doi.org/10.1016/j.atech.2021.100009

[17] C. Muppala and V. Guruviah, **Detection of leaf folder and yellow stemborer moths in the paddy field using deep neural network with search and rescue optimization**, *Inf. Process. Agric.*, vol. 8, no. 2, pp. 350–358, 2021. https://doi.org/10.1016/j.inpa.2020.09.002

[18] D. Zhu, L. Z. Xie, B. X. Chen *et al.*, **Knowledge graph and deep learning based pest detection and identification system for fruit quality**, *Internet Things*, vol. 21, p. 100649, 2023. https://doi.org/10.1016/j.iot.2022.100649

[19] Z. Dongmei, W. Ke, G. Hongbo *et al.*, **Classification and identification of citrus pests based on InceptionV3 convolutional neural network and migration learning**, *International Conference on Internet of Things and Intelligent Applications (ITIA)*, Zhenjiang, China, pp. 1–7, 2020. https://doi.org/10.1109/ITIA50152.2020.9312359

[20] S. Li, H. Wang, C. Zhang, and J. Liu, **A self-attention feature fusion model for rice pest detection**, *IEEE Access*, vol. 10, pp. 84063–84077, 2022. https://doi.org/10.1109/ACCESS.2022.3194925

[21] Y. Zhang and Y. P. Liu, **Identification of navel orange diseases and pests based on the fusion of DenseNet and self-attention mechanism**, *Comput. Intell. Neurosci.*, vol. 2021, p. 5436729, 2021. https://doi.org/10.1155/2021/5436729

[22] A. Berka, A. Hafiane, Y. Es-Saady *et al.*, **CactiViT: Image-based smartphone application and transformer network for diagnosis of cactus cochineal**, *Artif. Intell. Agric.*, vol. 9, pp. 12–21, 2023. https://doi.org/10.1016/j.aiia.2023.07.002

[23] Nitin, S. B. Gupta, R. Yadav, F. Bovand, and P. K. Tyagi, **Developing precision agriculture using data augmentation framework for automatic identification of castor insect pests**, *Front. Plant Sci.*, vol. 14, p. 1101943, 2023. https://doi.org/10.3389/fpls.2023.1101943

[24] A. Dosovitskiy, L. Beyer, A. Kolesnikov *et al.*, **An image is worth 16×16 words: Transformers for image recognition at scale**, *arXiv preprint arXiv:2010.11929*, 2020. https://doi.org/10.48550/arXiv.2010.11929

[25] A. Alnafjan and M. Aldayel, **Visual similarity detection for intellectual property using deep transfer learning**, *EMITTER International Journal of Engineering Technology*, vol. 13, no. 2, pp. 173–197, 2025. https://doi.org/10.24003/emitter.v13i2.852

[26] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, **ImageNet: A large-scale hierarchical image database**, *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, pp. 248–255, 2009. https://doi.org/10.1109/CVPR.2009.5206848

[27] S. F. Stefenon, L. O. Seman, A. C. R. Klaar, R. G. Ovejero, and V. R. Q. Leithardt, **Hypertuned-YOLO for interpretable distribution power grid fault location based on EigenCAM**, *Ain Shams Eng. J.*, vol. 15, no. 6, p. 102722, 2024. https://doi.org/10.1016/j.asej.2024.102722

[28] N. Yadav, S. B. Gupta, and R. K. Yadav, **Comparative analysis of first-order optimization algorithms**, in *Artificial Intelligence and Speech Technology*, CRC Press (Boca Raton), pp. 279–289, 2021. https://doi.org/10.1201/9781003150664-32

[29] M. L. Huang, T. C. Chuang, and Y. C. Liao, **Application of transfer learning and image augmentation technology for tomato pest identification**, *Sustain. Comput. Inform. Syst.*, vol. 33, p. 100646, 2022. https://doi.org/10.1016/j.suscom.2021.100646

[30] K. Simonyan and A. Zisserman, **Very deep convolutional networks for large-scale image recognition**, *3rd International Conference on Learning Representations (ICLR)*, pp. 1–14, 2015. https://doi.org/10.48550/arXiv.1409.1556

[31] X. Cheng, Y. Zhang, Y. Chen, Y. Wu, and Y. Yue, **Pest identification via deep residual learning in complex background**, *Comput. Electron. Agric.*, vol. 141, pp. 351–356, 2017. https://doi.org/10.1016/j.compag.2017.08.005

[32] M. M. Leonardo, T. J. Carvalho, E. Rezende, R. Zucchi, and F. A. Faria, **Deep feature-based classifiers for fruit fly identification (Diptera: Tephritidae)**, *31st Conference on Graphics, Patterns and Images (SIBGRAPI)*, pp. 41–47, 2018. https://doi.org/10.1109/SIBGRAPI.2018.00012

[33] S. Xing and M. Lee, **A study of tangerine pest recognition using advanced deep learning methods**, *Preprints*, 2018. https://doi.org/10.20944/preprints201811.0161.v1

[34] K. He, X. Zhang, S. Ren, and J. Sun, **Deep residual learning for image recognition**, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. https://doi.org/10.1109/CVPR.2016.90

[35] J. Almryad and H. Kutucu, **Automatic identification for field butterflies by convolutional neural networks**, *Eng. Sci. Technol. Int. J.*, vol. 23, no. 1, pp. 189–195, 2020. https://doi.org/10.1016/j.jestch.2020.01.006

[36] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, **Rethinking the inception architecture for computer vision**, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, 2016. https://doi.org/10.1109/CVPR.2016.308

[37] Y. Li, H. Wang, L. M. Dang, A. Sadeghi-Niaraki, and H. Moon, **Crop pest recognition in natural scenes using convolutional neural networks**, *Comput. Electron. Agric.*, vol. 169, p. 105174, 2020. https://doi.org/10.1016/j.compag.2019.105174

[38] D. Ozdemir and M. S. Kunduraci, **Comparison of deep learning techniques for classification of the insects in order level with mobile software application**, *IEEE Access*, vol. 10, pp. 35675–35684, 2022. https://doi.org/10.1109/ACCESS.2022.3163380