



ARTICLE

An Adaptive Features Fusion Convolutional Neural Network for Multi-Class Agriculture Pest Detection

Muhammad Qasim^{1,2} , Syed M. Adnan Shah¹, Qamas Gul Khan Safi¹, Danish Mahmood² , Adeel Iqbal^{3,*} , Ali Nauman³  and Sung Won Kim^{3,*}

¹Department of Computer Science, University of Engineering and Technology, Taxila, 47050, Pakistan

²Department of Computer Science, Shaheed Zulfiqar Ali Bhutto Institute of Science and Technology, Islamabad, 44000, Pakistan

³School of Computer Science and Engineering, Yeungnam University, Gyeongsan-si, 38541, Republic of Korea

*Corresponding Authors: Adeel Iqbal. Email: adeeliqbal@yu.ac.kr; Sung Won Kim. Email: swon@yu.ac.kr

Received: 02 March 2025; Accepted: 25 April 2025; Published: 19 May 2025

ABSTRACT: Grains are the most important food consumed globally, yet their yield can be severely impacted by pest infestations. Addressing this issue, scientists and researchers strive to enhance the yield-to-seed ratio through effective pest detection methods. Traditional approaches often rely on preprocessed datasets, but there is a growing need for solutions that utilize real-time images of pests in their natural habitat. Our study introduces a novel two-step approach to tackle this challenge. Initially, raw images with complex backgrounds are captured. In the subsequent step, feature extraction is performed using both hand-crafted algorithms (Haralick, LBP, and Color Histogram) and modified deep-learning architectures. We propose two models for this purpose: PestNet-EF and PestNet-LF. PestNet-EF uses an early fusion technique to integrate handcrafted and deep learning features, followed by adaptive feature selection methods such as CFS and Recursive Feature Elimination (RFE). PestNet-LF utilizes a late fusion technique, incorporating three additional layers (fully connected, softmax, and classification) to enhance performance. These models were evaluated across 15 classes of pests, including five classes each for rice, corn, and wheat. The performance of our suggested algorithms was tested against the IP102 dataset. Simulation demonstrates that the Pestnet-EF model achieved an accuracy of 96%, and the PestNet-LF model with majority voting achieved the highest accuracy of 94%, while PestNet-LF with the average model attained an accuracy of 92%. Also, the proposed approach was compared with existing methods that rely on hand-crafted and transfer learning techniques, showcasing the effectiveness of our approach in real-time pest detection for improved agricultural yield.

KEYWORDS: Artificial neural network (ANN); support vector machine (SVM); deep neural network (DNN); transfer learning (TL)

1 Introduction

Pest management is a critical aspect of modern agriculture, essential for safeguarding crop health and ensuring sustainable food production. The success of pest management strategies hinges on the ability to accurately detect pests at an early stage, preventing infestations before they cause significant damage. Traditional pest detection methods primarily rely on manual inspection, a process that is both labor-intensive and time-consuming. Furthermore, manual methods are prone to human error and may fail to detect pests hidden within dense foliage or complex backgrounds. As a result, there is an increasing need for automated, scalable, and reliable pest detection systems that can operate efficiently across diverse environments. In recent years, image processing technology has emerged as a promising solution for automated pest detection



and recognition. However, challenges like image quality and complex backgrounds necessitate advanced techniques for reliable detection.

Our methodology leverages both early and late fusion techniques to enhance pest detection accuracy. In the early fusion approach, we first preprocess the images to standardize them and then extract features using a combination of handcrafted methods and deep learning networks. Specifically, we employ multiple pre-trained deep networks—Inception V3, VGG-16, AlexNet, ResNet18, ResNet50, ResNet101, SqueezeNet, GoogleNet, and YOLO V3—alongside handcrafted features to create a comprehensive feature pool. We then apply an adaptive feature selection process, which includes CFS to rank features based on their relevance to the target class, and RFE to iteratively remove less important features.

In the late fusion approach, we extract features from the same set of networks and handcrafted methods, but instead of combining them early, we pass them to individual classifiers. Each classifier processes its set of features independently, and the final class label is determined using majority voting or averaging of the classifiers' predictions.

Our work introduces several key contributions to the field of pest detection:

- This study combines handcrafted and deep learning features using an early fusion approach, which integrates multiple pre-trained networks to create a rich feature pool, followed by adaptive feature selection techniques like CFS and RFE to enhance classification performance.
- This study contributes an advanced feature selection technique by implementing a dual-layer feature selection process that combines correlation-based ranking and recursive elimination, significantly improving the relevance and efficiency of features used in pest classification.
- This study utilized majority voting or averaging among classifiers to get a robust late fusion strategy. We enhance the robustness of the pest detection system, ensuring that the final classification benefits from the collective wisdom of multiple models.

Through these contributions, our approach aims to advance pest detection technology, offering a more accurate and practical solution for agricultural pest management.

The structure of the paper is as follows: [Section 2](#) describes the related work, and the specifics of the suggested methodology are contained in [Section 3](#). Comprehensive experiments and a comparative analysis with several state-of-the-art methods are presented in [Section 4](#). The paper is concluded in [Section 5](#).

2 Related Work

Early detection of pest disease is a critical element in protecting our agricultural products. A rapid response to eliminate, contain, or slow the spread of pests can be successful if a newly arrived pest is detected earlier. In recent years, many pest detection and recognition systems have been presented.

2.1 Hand-Crafted

Hand-crafted features such as SIFT [1] and HOG [2] have been commonly used for feature extraction. Saliency methods have also been widely adopted in recent years and have demonstrated impressive results. A saliency model was proposed in [3] and extended in [4] using the Markov chain algorithm for object detection. A novel approach, DFN-PSAN, was introduced in [5,6] for plant leaf and pest detection and classification, while a hand-crafted feature extraction technique was employed in [7] for the classification of tea and corn pests.

2.2 Deep Learning

With the advancement of neural networks, researchers have developed architectures that can overcome the limitations of hand-crafted models. Deep Convolutional Neural Networks (CNNs) techniques have shown awe-inspiring results over time, which include GoogleNet [8] and ResNet [9].

Preprocessing is necessary because specific objects need to be filtered out in order to increase the algorithm's efficiency. He also suggested selecting the output based on the highest probability obtained object. If that probability is greater than the selected threshold (TH), the object should be selected, or else it should be ignored [10,11]. Saliency methods were used by [12,13] for the feature extraction and used 1400 images with 24 classes of pest categories from the IP102 dataset, which contains a total of 75,000 images.

Improved CNN was used to classify apple pests [14–16]. Deep learning approaches like VGG, ResNet18, and ResNet-50 were used to detect various crops to predict the occurrence of pests [17–20]. A method for pest localization and classification to address inefficiencies in manual counting and current CNN-based algorithms was developed and used in various flavor YOLOv3, YOLOv5, YOLO-GBS in various studies [21–25]. These deep neural networks are designed to identify and categorize agricultural pests from photos, tackling concerns related to minute variations in appearance and size between pests. In order to improve representation capacity, Ren et al. [26] created a feature reuse residual block (FR-ResNet) that combines input residual signals tested on the IP102 dataset. In order to classify insect pests, Kasinathan et al. [27] experimented on the Wang and Xie dataset using ANN, SVM, KNN, NB, and CNN models.

Some of the most recent work under a highly complex background is mentioned next.

Jiao et al. [28] and Nanni et al. [29] used a modified CNN with an ensemble to successfully detect pests in complex scenes. In order to improve pest detection performance and accuracy, and minimize parameter size by addressing scale variability and optimizing feature extraction, Efficient Scale-Aware Network (ESA-Net) and YOLOv5 were used [30,31]. A simple MobileNet-based pest classification technique was improved by a dual-branch feature fusion module and the ASGE attention mechanism. It improves classification accuracy and efficiency by optimizing the model architecture and activation functions [32,33]. Table 1 shows the most recent literature work containing strengths and weaknesses.

Table 1: Comparative analysis and limitations of existing state-of-the-art techniques

Techniques	Strength	Weakness	Dataset	Accuracy
GAN-ResNet [10]	The GAN-MSDB-ResNet model achieves high training and test recognition accuracy for pest identification.	Relies on enhanced datasets, which do not represent the variability and complexity.	Aug-Dataset	99.34%
YOLOv5 [20,26]	The YOLO-GBS model achieves a significant performance in complex scenes.	The research may lack extensive real-world validation and scalability.	Pady Pest, IP102	79.8%
ResNet-based model [27]	Good for aphid detection.	Limited by data dependency and computational needs.	Agricultural Pest	75.77%

(Continued)

Table 1 (continued)

Techniques	Strength	Weakness	Dataset	Accuracy
R-CNN [28]	Method achieves high accuracy in a two-stage R-CNN.	Method's complexity and limited applicability in limited resources or smaller datasets.	AgriPest21	77%
Ensemble CNNs [29]	The ensemble of CNNs performed well on multiple pest datasets.	Method's complexity and optimization variants are required.	IP102	99.81%
Yolo v5 [31]	Improve accuracy using PestLite model.	Limited applicability to other agricultural scenarios without further adaptation.	IP102	90.7%
MobileNet-V2 [32]	PestNet(Lightweight) enhances classification accuracy with lower complexity.	Improvements like attention mechanisms and multi-scale fusion for task adaptation.	IP102	87.62%

3 Proposed Methodology

This research addresses the plant pest classification problem by applying transfer learning to ten state-of-the-art networks, utilizing two innovative methods: PestNet-EF (Early Fusion) and PestNet-LF (Late Fusion with Averaging and Majority Voting).

3.1 PestNet-EF (Early Fusion)

In this study, we implemented an Early Fusion methodology structured into four distinct blocks to enhance the accuracy and efficiency of pest detection. The first block, Image Acquisition and Preprocessing, involves capturing high-quality pest images and applying various preprocessing techniques such as resizing, normalization, and background removal to prepare the images for feature extraction. This ensures that the images are standardized, reducing noise and variations that could negatively impact the subsequent analysis.

The second block, Feature Extraction, combines both handcrafted and deep learning features to create a comprehensive feature pool. Handcrafted features are derived from traditional image processing techniques, while deep features are extracted from multiple pre-trained networks, including Inception V3, VGG-16, AlexNet, ResNet variants, SqueezeNet, GoogleNet, and YOLO V3. These features capture a wide range of visual characteristics, from simple textures to complex patterns, ensuring a rich and diverse set of features for classification. In the third block, Adaptive Feature Selection, a two-step process is employed: first, CFS ranks the features based on their correlation with the target class, identifying the most relevant features. Next, RFE systematically removes less important features, leaving only the most discriminative ones. The reason for choosing feature selection was to enhance classification performance by identifying and retaining the most relevant features, which improves model accuracy and reduces overfitting. Unlike dimensionality reduction techniques, which can obscure interpretability by transforming feature spaces, our approach of using methods like CFS and RFE directly highlights and selects the best features, preserving their original meaning. Additionally, global pooling methods may lead to loss of critical information by averaging features, while our method retains the most informative attributes for robust pest detection.

In the final block, the selected features are then concatenated, forming a final feature vector used in the classification process as shown in Fig. 1 (Blocks 1, 2, 3, 4). We chose simple concatenation for the Early Fusion (EF) type to prioritize interpretability, computational efficiency, and establish a clear baseline for performance. This straightforward method allows us to effectively integrate handcrafted and deep learning features without unnecessary complexity, ensuring clarity in demonstrating our system’s efficacy for pest detection.

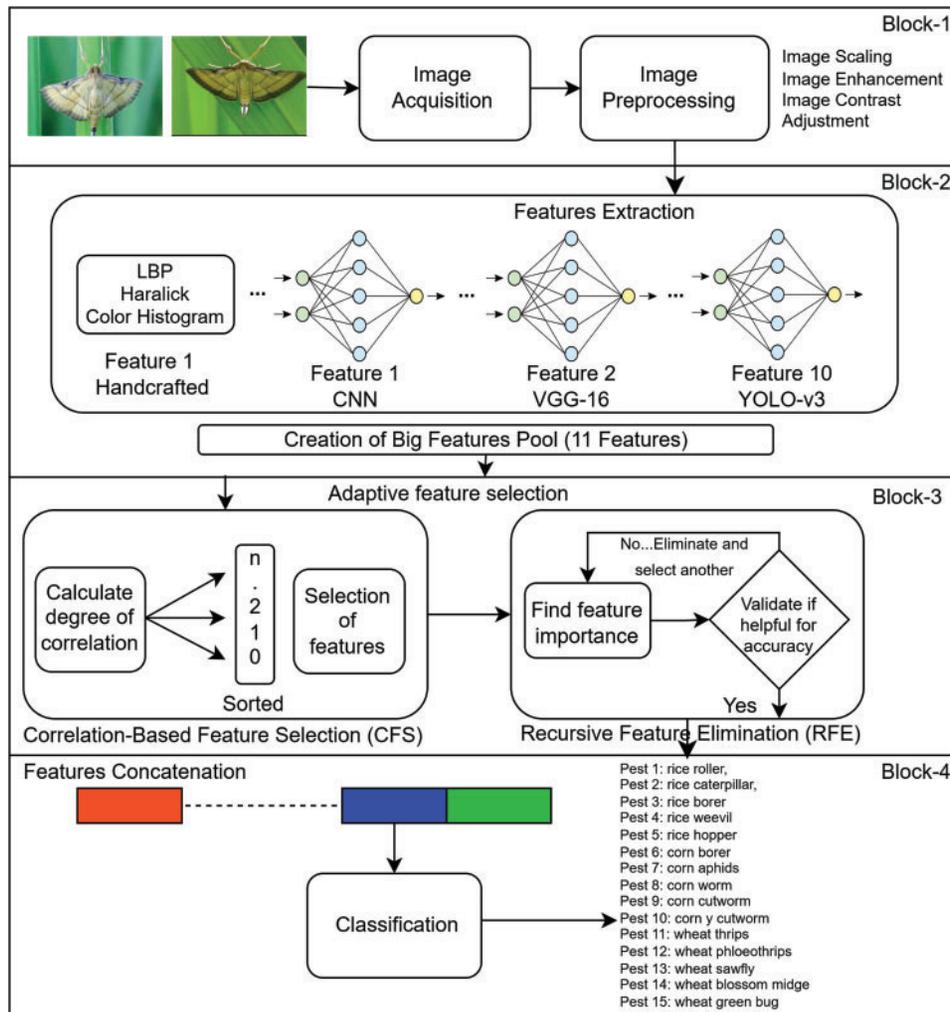


Figure 1: PestNet EF model

3.1.1 Correlation-Based Feature Selection (CFS)

In our Early Fusion methodology, we implemented CFS [34] to enhance feature relevance before fusion. CFS evaluates the correlation between each feature and the target class, selecting features that are highly correlated with the class but uncorrelated with each other. The selected features maximize the merit function.

$$M_s = \frac{k \cdot \overline{r_{cf}}}{\sqrt{k + k(k-1) \cdot \overline{r_{ff}}}} \tag{1}$$

where k is the number of features, r_{cf} is the average feature-class correlation, and r_{ff} is the average feature-feature intercorrelation, as shown in Eq. (1).

3.1.2 Recursive Feature Elimination (RFE)

In our Early Fusion methodology, we applied RFE to refine feature selection [35]. RFE iteratively removes the least important features based on their impact on the classifier's performance. The importance of each feature is determined by the weights w_i in the linear model, and the model's performance is evaluated using a metric such as accuracy or F1-score. At each iteration, the features with the smallest weights are removed, and this process continues until the optimal subset of features is identified, maximizing the classifier's performance using the function minimize $\text{Loss}(\mathcal{F}^*)$. where \mathcal{F} is the feature set and \mathcal{F}^* is the selected subset of features and Loss evaluates the classifier's performance.

3.2 PestNet-LF (Late Fusion)

In our approach, we employed a Late Fusion methodology, organized into four key blocks to optimize the pest detection process. Image Acquisition and Preprocessing involves collecting pest images and applying essential preprocessing steps, such as resizing, normalization, and background removal, to standardize the images. These preprocessing techniques are crucial for minimizing noise and ensuring that the images are in the optimal format for feature extraction, thus laying a strong foundation for the subsequent analysis.

The next section focuses on Feature Extraction, where both handcrafted and various deep-learning features were extracted to form a comprehensive feature pool. Handcrafted features derived from traditional image processing techniques capture specific characteristics relevant to pest identification. Simultaneously, deep features are obtained from a variety of pre-trained networks, including Inception V3, VGG-16, and several ResNet variants, among others. Each network contributes unique representations, capturing different levels of abstraction and detail. These features are passed to individual classifiers, each operating independently. The classifiers then produce their predictions based on the features they receive. Finally, majority voting and averaging mechanisms are applied to these predictions, where the class label with the most votes across all classifiers is selected as the final decision. This late fusion strategy ensures that the strengths of multiple classifiers are combined, leading to a more robust and accurate classification outcome.

Fig. 2 represents the main flow diagram of PestNet-LF. This dual approach in PestNet-LF allows us to combine the strengths of multiple network architectures and handcrafted features, resulting in a versatile and highly accurate pest classification system.

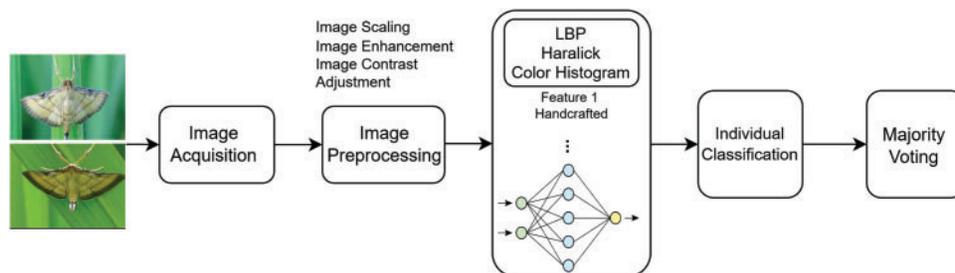


Figure 2: PestNet LF model

Averaging and Majority Voting Method

Averaging and majority voting techniques were employed, where the confidence scores from each classifier were averaged and voted to determine the final class label. This approach allows for a more balanced decision-making process, as it considers the contribution of each classifier proportionally, leading to a smoother and potentially more accurate classification outcome.

3.3 Handcrafted Features

Haralick for texture, Hu Moments for Shape, and Color Histogram for color were used for handcrafted feature extraction. The Haralick texture method is mainly used for image texture because of its simplicity and intuitiveness. Haralick uses a grey-level co-occurrence matrix (GLCM) that uses neighboring pixels to find the image information.

Hu Moments uses seven other numbers by using central moments [36]. Some moments include translation, scale, rotation, and reflection. The reason for using Hu Moments is its property of translation invariance. A moment is shown in Eq. (2).

$$M = \sum_x \sum_y I(x, y) \quad (2)$$

A central moment is shown in Eq. (4).

$$\mu_{i,j} = \sum_x \sum_y (x, \bar{x})^i (y, \bar{y})^j I(x, y) \quad (3)$$

Color Histogram was used to extract the color features of images. Color histogram uses frequency distribution to bin different colors in an image and creates separate histograms for the three other channels, RGB.

3.4 Deep Learning Architectures

The identification and recognition of objects make extensive use of deep learning models. We have used the following deep learning architectures in this work: Inception V3, VGG-16, AlexNet, ResNet18, ResNet50, ResNet101, SqueezeNet, GoogleNet, YOLO V3, EfficientNet, Vision, and Swin Transform model. The characteristics of these networks are presented in Table 2.

Table 2: Deep learning and transfer learning architectures with number of parameters

DL architectures	No. of parameters	No. of layers	Size in MB
AlexNet [10]	60 M	8	227
ResNet18 [12]	25 M	18	44
VGG-16 [18]	138 M	23	528
ResNet50 [18,27]	23 M	50	98
GoogleNet [21]	7 M	22	27
YOLO V3 [16,24,25]	62 M	106	30
Inception V3	27 M	42	93
EfficientNet	3.5 M	337	22
Vision transform	86 M	12	330
Swin transform	88 M	12	340

An input layer, hidden layers, and an output layer make up a convolutional neural network. One or more convolution-performing layers are included in the hidden layers of a convolutional neural network. This usually consists of a layer that uses the layer's input matrix to perform a dot product of the convolution kernel.

4 Experimental Results

This section presents the experimental results and evaluates the model's performance based on the conducted assessments. The research utilized a system with an Intel Quad-Core i7 2820QM processor (2.30 GHz), 16 GB of RAM, an NVIDIA Quadro graphics card, a Samsung 850 Pro 256 GB SSD, and a 1 TB hard drive. The software environment included Windows 7 Professional and Python 3.8.

4.1 IP102 Dataset

Insect pest classification is getting more important day by day. This research utilized the IP102 dataset exclusively, leveraging its extensive collection of over 75,000 images across 102 pest classes, specifically designed for detailed insect pest classification. We chose IP102 over datasets like AgriPest21 and AgriPest because it serves as a recognized benchmark, facilitating direct performance comparisons with state-of-the-art methods and offering a focused analysis within this domain. Furthermore, its fine-grained curation and diverse range of agricultural pests allowed for targeted model development. This dataset's depth ensured robust training and evaluation, enhancing the model's generalization capabilities across diverse crop types, while also alleviating the computational demands associated with processing multiple large datasets. The details from the dataset of the IP102 are shown in Table 3, and Fig. 3 shows the sample of images from IP102.

Table 3: IP102 dataset classes detail

	Super-Class	Class	Train	Val	Test	IR	
FC		Rice	14	5043	843	2531	6.4
		Corn	13	8404	1399	4212	27.9
		Wheat	9	2048	340	1030	5.2
		Beet	8	2649	441	1330	15.4
		Alf alfa	13	6230	1037	3123	10.7
EC		Vitis	16	10,525	1752	5274	74.8
		Citrus	19	4356	725	2192	17.6
		Mango	10	5840	971	2927	61.7
IP102		FC	57	24,602	4098	12,341	39.4
		EC	45	20,721	3448	10,393	80.8
	IP102	102	45,095	7508	22,619	80.8	

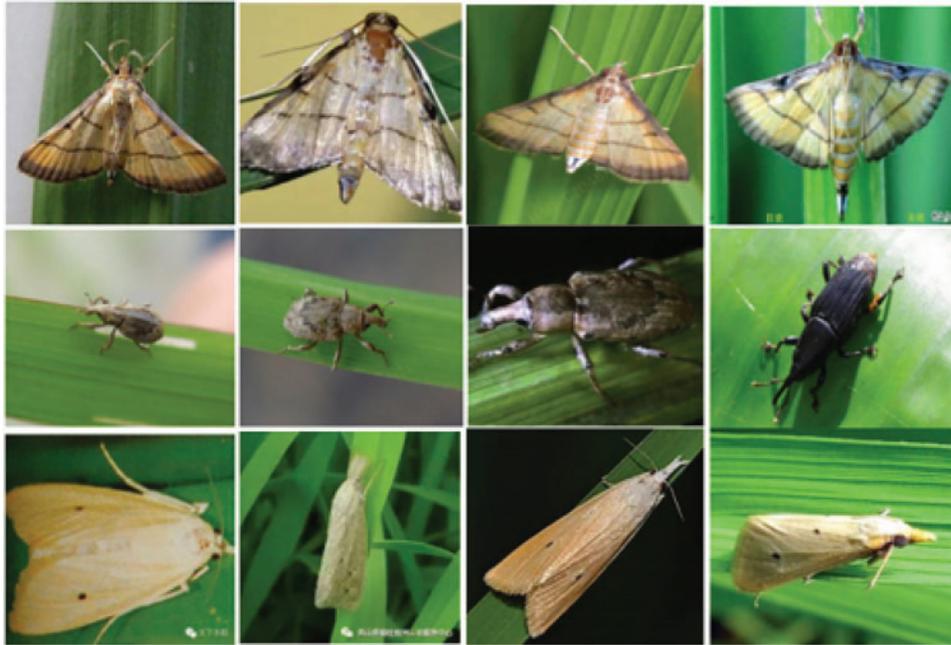


Figure 3: Image samples from IP102

We narrowed our focus to 15 pest classes from IP102 to prioritize those most impactful to staple crops like rice, corn, and wheat, ensuring immediate practical relevance. This subset provided a diverse visual spectrum, challenging the model with varying shapes and appearances, while also reflecting common and rare pest occurrences as documented in agricultural research. This approach allowed for concentrated model optimization, maximizing its efficacy in real-world pest management scenarios. Each crop contains 500 images of rice, corn, and wheat pests respectively, whose names are rice roller, rice caterpillar, rice borer, rice weevil, rice hopper, corn borer, corn aphids, corn worm, corn cutworm, corn yellow cutworm, wheat thrips, wheat phloeothrips, wheat sawfly, wheat blossom midge and wheat green bug.

4.2 Data Augmentation

During training, we augmented our dataset with random translations ($\pm 10\%$), horizontal flips, and scaling (0.8–1.2x) to enhance model robustness to pest position and size variations [37,38]. Crucially, augmentation was exclusive to training; testing was performed on unaltered images to accurately evaluate model performance on genuine data. This approach ensured the model's effectiveness was measured against real-world scenarios, maintaining the integrity of our evaluation. For the labeling, one of the free labeling tools, LabelMe, was used to label images.

4.3 Performance Measure

Object-level testing was used, which involves measuring the accuracy and efficiency of objects detected from an image.

$$Accuracy = \frac{TPR + TNR}{TPR + TNR + FPR + FNR} \quad (4)$$

$$\text{Precision } (P) = \frac{TPR}{TPR + FPR} \quad (5)$$

$$\text{Recall } (R) = \frac{TPR}{TPR + FNR} \quad (6)$$

$$F_1\text{Score } (F1) = \frac{P + R}{2} \quad (7)$$

Eqs. (4)–(7) are used to determine accuracy, precision, recall, and F-score. In this case, the numbers for true positives (TPR), true negatives (TNR), false positives (FPR), and false negatives (FNR) are represented.

4.4 Results

Results were calculated using early fusion PestNet-EF and late fusion PestNet-LF techniques and their different variants. Each model is evaluated based on inference time, FLOPs (floating-point operations), memory usage, and accuracy. We optimized inference times to remain under 100 ms, ensuring our models can operate in real-time applications effectively. The FLOPs, ranging from 120 to 160 million, reflect the computational demands typical of advanced deep learning architectures. We also maintained memory usage between 900 to 1000 MB to accommodate the complexity of the models. The PestNet-EF model demonstrated exceptional performance with an accuracy of 96%, highlighting its effectiveness in accurately detecting and classifying pests as shown in Table 4.

Table 4: Accuracy of Deep Learning techniques with and without enhancement

Model variant	Inference time (ms)	FLOPs (Millions)	Memory usage (MB)	Accuracy (%)
PestNet-EF (No Augmentation)	76	160	1000	87
Baseline model	75	120	900	88
CFS Only Model	80	135	850	84
RFE Only Model	78	130	870	86
PestNet-LF (Average)	82	140	950	92
PestNet-LF (Majority)	81	145	1000	94
PestNet-EF	77	160	1000	96

The enhanced deep learning techniques were used in the PestNet-EF and PestNet-LF modules. The training accuracy and validation accuracy of PestNet-EF for 100 epochs are shown in Figs. 4 and 5.

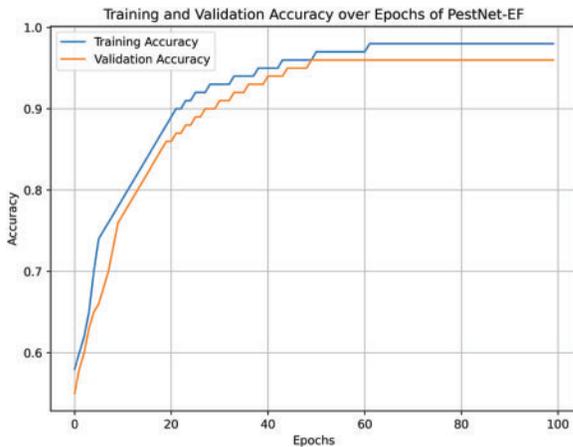


Figure 4: Training accuracy and validation accuracy of PestNet-EF

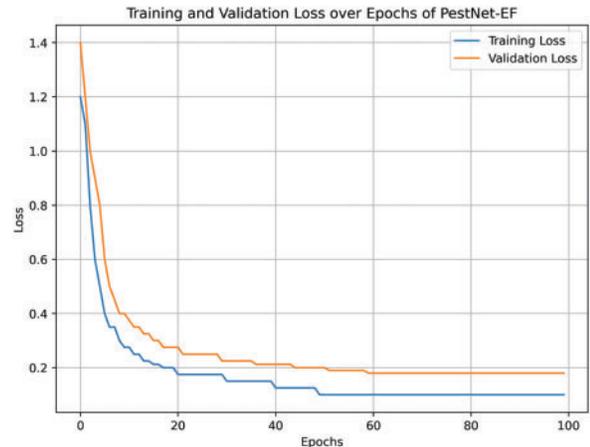


Figure 5: Training loss and validation loss of PestNet-EF

The PestNet-EF performance was evaluated using confusion matrices and their normalized counterparts for each of the 15 crop disease classes (five each for rice, corn, and wheat). The model achieved an overall accuracy of 96%, demonstrating robust performance in distinguishing among the diverse set of plant diseases. Fig. 6 shows the Confusion matrix of PestNet-EF.

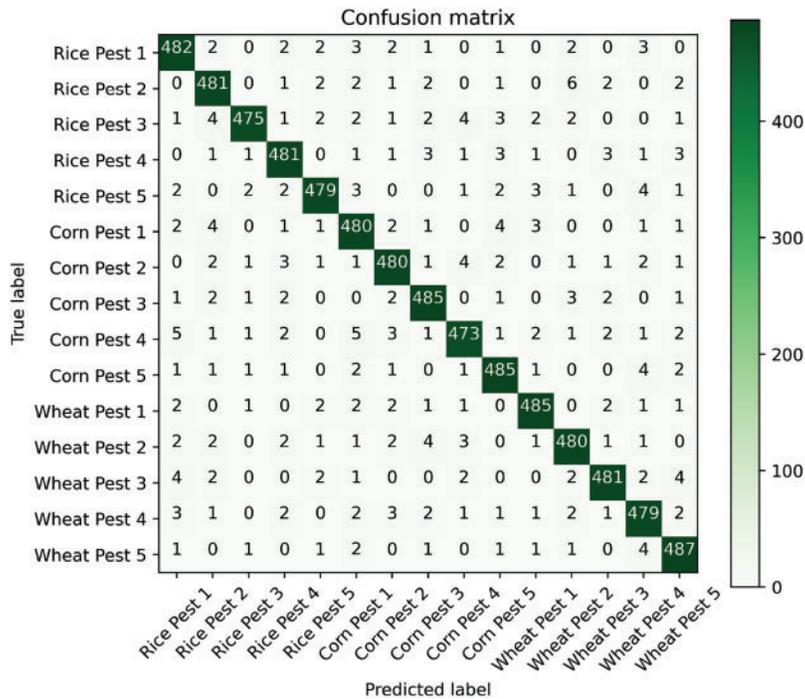


Figure 6: Confusion matrix of PestNet-EF

Early Fusion (EF) performed better than Late Fusion (LF) primarily due to its ability to integrate handcrafted and deep-learning features at an earlier stage, leading to a richer and more comprehensive

representation of data. This integration aligns with the principles of feature-level fusion, as described in studies on multimodal learning [39]. This integration allows the model to capitalize on the complementary strengths of both feature types, enhancing the training process by providing a unified feature set that captures intricate relationships essential for accurate classification. In contrast, LF processes features independently across classifiers, which can dilute the information synergy and lead to a less effective overall model performance. The combined strength of features in EF facilitates better decision-making, ultimately resulting in higher accuracy.

PestNet-EF and PestNet-LF were also tested in combination with different techniques, instead of all 10 of the models going into the early fusion or late fusion. Table 5 shows that the combination of the PestNet-EF model, GoogleNet, YOLOv3, and ResNet50 got a precision of 94%, a Recall of 95%, and an F-score of 96%. PestNet-LF with Average got a precision of 91%, a Recall of 89% and an F-Score of 90%, and with Majority voting got a precision of 92%, a Recall of 90% and an F-Score of 91%. Details of the rest of the combination results are given below in Table 5.

Table 5: Results of PestNet-EF and PestNet-LF models

Combination techniques	PestNet-EF model			PestNet-LF model		
	Precision (%)	Recall (%)	F-1 Score (%)	Precision (%)	Recall (%)	F-1 Score (%)
Handcrafted + YOLOv3 + ResNet50	89.00	94.70	92.23	90.00	89.00	91.00
CNN + YOLOv3 + ResNet50	92.50	95.35	94.15	92.00	90.18	90.00
InceptionV3 + YOLOv3 + ResNet50	90.00	94.34	92.31	91.00	90.79	89.86
GoogleNet + YOLOv3 + ResNet50	94.00	95.37	96.26	92.00	89	92.37%
Alexnet + YOLOv3 + VGG16	91.92	95.29	93.57	92.00	90.43	93.38
ResNet101 + YOLOv3 + ResNet50	93.00	94.90	93.94	92.00	89.84	90.36
SqueezeNet + YOLOv3 + ResNet50	92.00	93.34	94.12	93.00	89.88	91.42
Average	–	–	–	91.43	89.03	90.20%
Majority Voting	–	–	–	92.00	90.46	91.37%

Fig. 7 shows the performance comparisons of Precision, Recall, Accuracy, and F1-Score between the Handcrafted features extraction techniques and CNN, Inception V3, VGG-16, AlexNet, ResNet18, ResNet50, ResNet101, SqueezeNet, GoogleNet, and YOLO V3 and their combinations using PestNet-LF and PestNet-EF.

4.5 Comparison with State of the Art

Table 6 encompasses precision, recall, F1-score, and accuracy for PestNetLF (Average), PestNetLF (Majority Voting), and PestNetEF, and different deep learning Inception V3, VGG-16, ResNet50, ResNet101, SqueezeNet, GoogleNet, and YOLO V3, and transfer learning algorithms EfficientNet, Vision Transform, and Swin Transform models. Results indicate that PestNetLF with Majority Voting achieved an accuracy of 94%, outperforming the average variant. Notably, PestNetEF demonstrated superior overall performance with an accuracy of 96%, highlighting the effectiveness of the proposed enhancements.

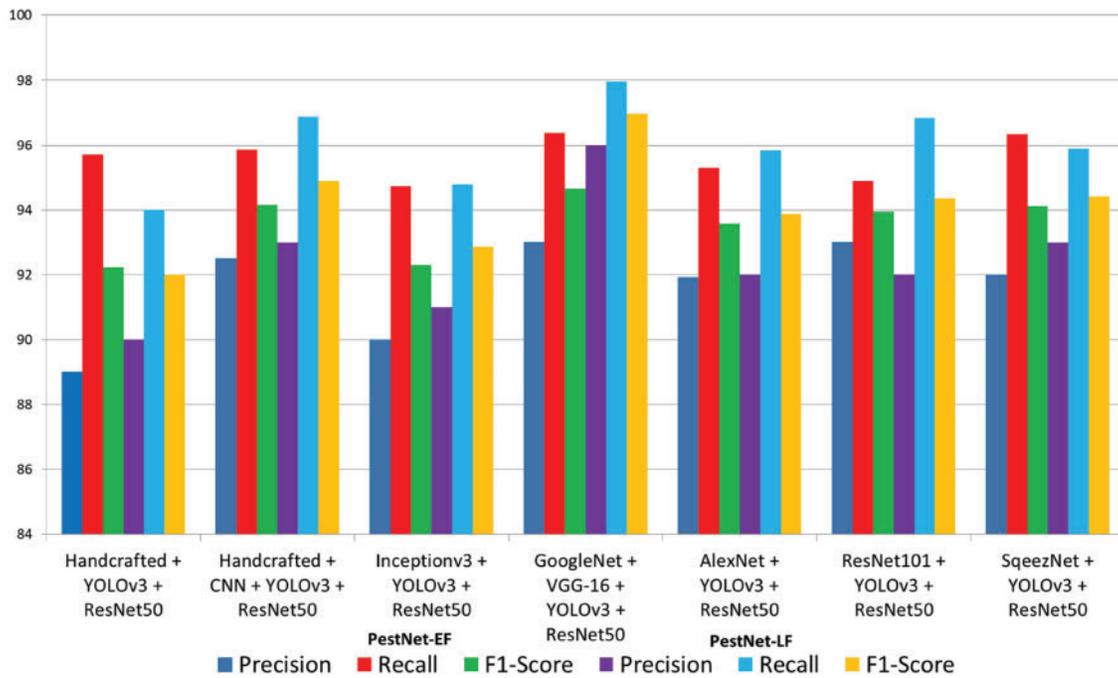


Figure 7: Performance comparison of precision, recall and F1-Score of early and late fusion

Table 6: Results comparisons of PestNet-EF and PestNet-LF with state-of-the-art models

	Type	Comp. complexity	Precision (%)	Recall (%)	F-1 Score (%)	Accuracy (%)
Handcrafted	Manual	Low	76	79	78	75
Inception V3	Deep learning	Moderate	91	91	93	89
VGG-16	Deep learning	Moderate	89	92	90	86
ResNet50	Deep learning	Moderate	91	90	93	88
ResNet101	Deep learning	Moderate	90	89	92	89
SqueezeNet	Deep learning	Moderate	92	92	92	87
GoogleNet	Deep learning	Moderate	94	91	94	91
YOLO V3	Deep learning	Moderate	93	93	92	90
EfficientNet	Deep learning	Moderate	91	89	90	92
Vision Transformer (ViT)	Transformer-based	High	90	91	92	92
Swin Transformer	Hierarchical transformer	High	93	94	94	95
PestNet-LF (Average)	Hybrid (Deep learning)	Moderate	91	89	90	92
PestNet-LF (Majority)	Hybrid (Deep learning)	Moderate	92	90	93	94
PestNet-EF	Hybrid (Deep learning)	Moderate	94	95	96	96

Figs. 8–11 compare the performance of our suggested method with state-of-the-art work. This illustrates the growth of F1-Scores, Precision, Recall, and Accuracy over time.

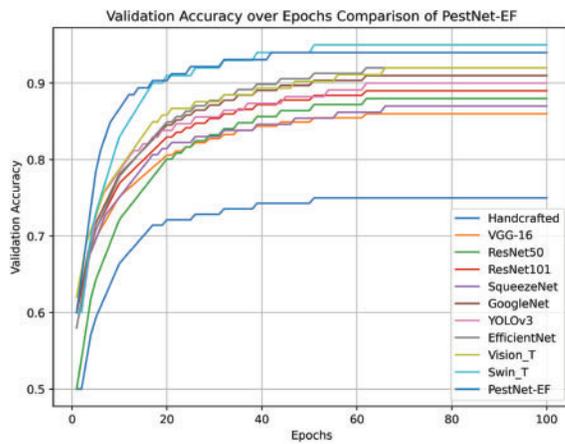


Figure 8: Validation accuracy over epochs comparison of IP102 dataset

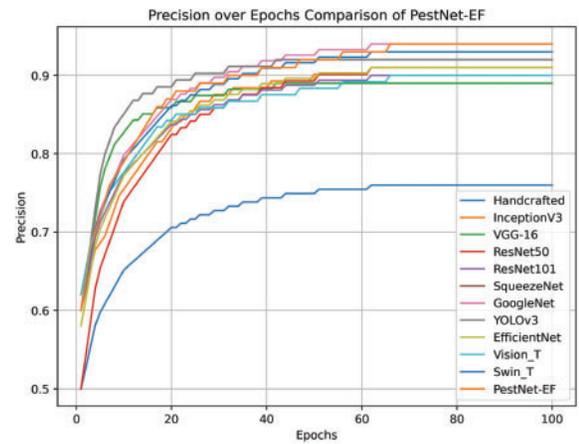


Figure 9: Precision over epochs comparison of IP102 dataset

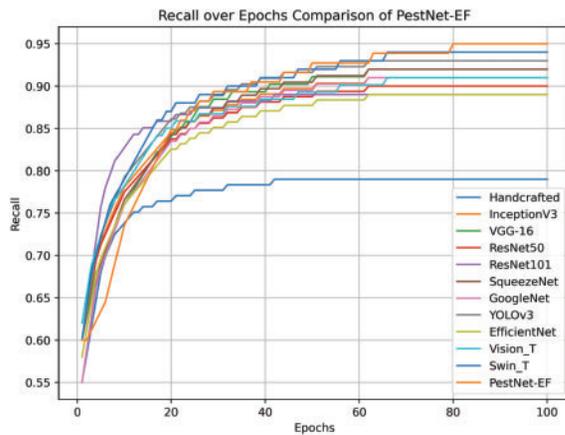


Figure 10: Recall over epochs comparison of IP102 dataset

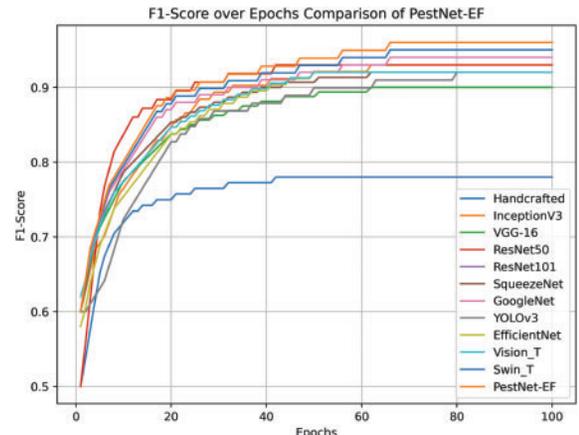


Figure 11: F1-Score over epochs comparison of IP102 dataset

Table 7 compares the proposed PestNet-EF methodology with latest work, showing a clear progression in accuracy from traditional handcrafted features to advanced neural network techniques. This shows only Kasinathan's results reached 90% rest of them are 86%, 77%, and even 74% on the same dataset. The PestNet-EF methodology represents the latest advancement, combining state-of-the-art techniques with potentially novel contributions in model architecture, feature extraction, and training. The substantial improvement in accuracy to 96% indicates that PestNet-EF effectively addresses the limitations of previous approaches and leverages advanced techniques to achieve superior performance.

Table 7: Comparison with state of the art work

Source	Approaches used	Dataset	Classes	Accuracy (%)
Kasinathan et al. [27]	CNN	IP102	24	90
Nanni et al. [29]	CNN	IP102	10	74.11
Jiao et al. [28]	Fused CNN	AgriPest21	11	77
Dong et al. [31]	YOLO V5	IP102	9	86
Peng et al. [32]	MobileNet V2	IP102	12	86
Proposed methodology	PestNet-EF	IP102	15	96

5 Conclusion

This paper proposes an original, profound learning approach for pest detection in the agriculture field by the hybridization of hand-crafted and automatically extracted features. The proposed system uses two models, PestNet-EF and PestNet-LF, which use early and late fusion approaches to process images with complex backgrounds to highlight pests accurately by eliminating everything in the image. In early fusion, features extracted from handcrafted approaches (Hu moment, Haralick, and Color Histogram) are fused with 10 deep learning architectures, and additionally, adaptive feature selection methodology using CFS and RFE, and then features are classified using SVM. On the other hand, in the late fusion approach, handcrafted features are combined with different combinations of modified deep learning architectures (containing three extra layers) to extract features. The proposed fusion methodology outperforms individual hand-crafted techniques. Given the tests of different image datasets and the datasets obtained from various greenhouses, the PestNet-EF algorithm achieved an accuracy of 96%, and PestNet-LF achieved an accuracy of 94% using majority voting.

Acknowledgement: The authors declare that there are no acknowledgments to report.

Funding Statement: This research was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2021R1A6A1A03039493) and in part by the NRF grant funded by the Korean government (MSIT) (NRF-2022R1A2C1004401).

Author Contributions: Conceptualization, Muhammad Qasim, Danish Mahmood, Adeel Iqbal; methodology, Syed M. Adnan Shah, Qamas Gul Khan Safi, Ali Nauman; validation, Muhammad Qasim, Adeel Iqbal, Sung Won Kim; investigation, Muhammad Qasim, Qamas Gul Khan Safi; writing—original draft preparation, Muhammad Qasim, Qamas Gul Khan Safi, Ali Nauman; writing—review and editing, Danish Mahmood, Syed M. Adnan Shah; supervision, Syed M. Adnan Shah, Sung Won Kim, Danish Mahmood; project administration, Sung Won Kim; funding acquisition, Sung Won Kim, Adeel Iqbal. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Data available on request from the authors.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Lowe DG. Distinctive image features from scale-invariant keypoints. *Int J Comput Vis.* 2004;60(2):91–110. doi:10.1023/B:VISI.0000029664.99615.94.

2. Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05); 2005 Jun 20–25; San Diego, CA, USA. p. 886–93. doi:10.1109/CVPR.2005.177.
3. Itti L, Koch C, Niebur E. A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans Pattern Anal Mach Intell.* 1998;20(11):1254–9. doi:10.1109/34.730558.
4. Harel J, Koch C, Perona P. Graph-based visual saliency. In: *Advances in neural information processing systems.* Vol. 19; 2006. p. 545–52. doi:10.7551/mitpress/7503.001.0001.
5. Dai G, Fan J, Dewi C. ITF-WPI: image and text based cross-modal feature fusion model for wolfberry pest recognition. *Comput Electron Agric.* 2023;212(2):108129. doi:10.1016/j.compag.2023.108129.
6. Dai G, Tian Z, Fan J, Sunil CK, Dewi C. DFN-PSAN: multi-level deep information feature fusion extraction network for interpretable plant disease classification. *Comput Electron Agric.* 2024;216(1):108481. doi:10.1016/j.compag.2023.108481.
7. Xie C, Zhang J, Li R, Li J, Hong P, Xia J, et al. Automatic classification for field crop insects via multiple-task sparse representation and multiple-kernel learning. *Comput Electron Agric.* 2015;119(2):123–32. doi:10.1016/j.compag.2015.10.015.
8. Yulita IN, Rambe MFR, Sholahuddin A, Prabuwono AS. A convolutional neural network algorithm for pest detection using GoogleNet. *AgriEngineering.* 2023;5(4):2366–80. doi:10.3390/agriengineering5040145.
9. Zhang X, Li H, Sun S, Zhang W, Shi F, Zhang R, et al. Classification and identification of apple leaf diseases and insect pests based on improved ResNet-50 model. *Horticulturae.* 2023;9(9):1046. doi:10.3390/horticulturae9091046.
10. Rustia DJA, Lin CE, Chung J-Y, Zhuang Y-J, Hsu J-C, Lin T-T. Application of an image and environmental sensor network for automated greenhouse insect pest monitoring. *J Asia Pac Entomol.* 2020;23(1):17–28. doi:10.1016/j.aspen.2019.11.006.
11. Geng C, Huang S-J, Chen S. Recent advances in open set recognition: a survey. *IEEE Trans Pattern Anal Mach Intell.* 2020;43(10):3614–31. doi:10.1109/TPAMI.2020.2981604.
12. Nanni L, Maguolo G, Pancino F. Insect pest image detection and recognition based on bio-inspired methods. *Ecol Inform.* 2020;57(2):101089. doi:10.1016/j.ecoinf.2020.101089.
13. Xia W, Han D, Li D, Wu Z, Han B, Wang J. An ensemble learning integration of multiple CNN with improved vision transformer models for pest classification. *Ann Appl Biol.* 2023;182(2):144–58. doi:10.1111/aab.12804.
14. Jiao L, Dong S, Zhang S, Xie C, Wang H. AF-RCNN: an anchor-free convolutional neural network for multi-categories agricultural pest detection. *Comput Electron Agric.* 2020;174(11):105522. doi:10.1016/j.compag.2020.105522.
15. Jiang P, Chen Y, Liu B, He D, Liang C. Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access.* 2019;7:59069–80. doi:10.1109/ACCESS.2019.2914929.
16. Tian Y, Yang G, Wang Z, Li E, Liang Z. Detection of apple lesions in orchards based on deep learning methods of CycleGAN and YOLOV3-dense. *J Sens.* 2019;2019:1–13. doi:10.1155/2019/7630926.
17. Fuentes A, Yoon S, Park DS. Deep learning-based phenotyping system with glocal description of plant anomalies and symptoms. *Front Plant Sci.* 2019;10:1321. doi:10.3389/fpls.2019.01321.
18. Hu K, Liu Y, Nie J, Zheng X, Zhang W, Liu Y, et al. Rice pest identification based on multi-scale double-branch GAN-ResNet. *Front Plant Sci.* 2023;14:1167121. doi:10.3389/fpls.2023.1167121.
19. Dewi C, Christanto HJ, Dai GW. Automated identification of insect pests: a deep transfer learning approach using ResNet. *Acadlore Trans Mach Learn.* 2023;2(4):194–203. doi:10.56578/ataiml020402.
20. Amrani A, Diepeveen D, Murray D, Jones MG, Sohel F. Multi-task learning model for agricultural pest detection from crop-plant imagery: a Bayesian approach. *Comput Electron Agric.* 2023;218:108719. doi:10.1016/j.compag.2024.108719.
21. Chen C-J, Huang Y-Y, Li Y-S, Chang C-Y, Huang Y-M. An AIoT based smart agricultural system for pests detection. *IEEE Access.* 2020;8(9):180750–61. doi:10.1109/ACCESS.2020.3024891.
22. Arsenovic M, Karanovic M, Sladojevic S, Anderla A, Stefanovic D. Solving current limitations of deep learning based approaches for plant disease detection. *Symmetry.* 2019;11(7):939. doi:10.3390/sym11070939.

23. Zheng Y-Y, Kong J-L, Jin X-B, Wang X-Y, Su T-L, Zuo M. CropDeep: the crop vision dataset for deep-learning-based classification and detection in precision agriculture. *Sensors*. 2019;19(5):1058. doi:10.3390/s19051058.
24. Amin J, Anjum MA, Zahra R, Sharif MI, Kadry S, Sevcik L. Pest localization using YOLOv5 and classification based on quantum convolutional network. *Agriculture*. 2023;13(3):662. doi:10.3390/agriculture13030662.
25. Hu Y, Deng X, Lan Y, Chen X, Long Y, Liu C. Detection of rice pests based on self-attention mechanism and multi-scale feature fusion. *Insects*. 2023;14(3):280. doi:10.3390/insects14030280.
26. Ren F, Liu W, Wu G. Feature reuse residual networks for insect pest recognition. *IEEE Access*. 2019;7:122758–68. doi:10.1109/ACCESS.2019.2938194.
27. Kasinathan T, Singaraju D, Uyyala SR. Insect classification and detection in field crops using modern machine learning techniques. *Inf Process Agric*. 2021;8(3):446–57. doi:10.1016/j.inpa.2020.09.006.
28. Jiao L, Xie C, Chen P, Du J, Li R, Zhang J. Adaptive feature fusion pyramid network for multi-classes agricultural pest detection. *Comput Electron Agric*. 2022;195:106827. doi:10.1016/j.compag.2022.106827.
29. Nanni L, Manfè A, Maguolo G, Lumini A, Brahnam S. High performing ensemble of convolutional neural networks for insect pest image detection. *Ecol Inform*. 2022;67(2):101515. doi:10.1016/j.ecoinf.2021.101515.
30. Dong S, Teng Y, Jiao L, Du J, Liu K, Wang R. ESA-Net: an efficient scale-aware network for small crop pest detection. *Expert Syst Appl*. 2024;236(3):121308. doi:10.1016/j.eswa.2023.121308.
31. Dong Q, Sun L, Han T, Cai M, Gao C. PestLite: a novel YOLO-based deep learning technique for crop pest detection. *Agriculture*. 2024;14(2):228. doi:10.3390/agriculture14020228.
32. Peng H, Xu H, Shen G, Liu H, Guan X, Li M. A lightweight crop pest classification method based on improved MobileNet-V2 model. *Agronomy*. 2024;14(6):1334. doi:10.3390/agronomy14061334.
33. Qasim M, Mahmood D, Bibi A, Masud M, Ahmed G, Khan S, et al. PCA-based advanced local octa-directional pattern (ALODP-PCA): a texture feature descriptor for image retrieval. *Electronics*. 2022;11(2):202. doi:10.3390/electronics11020202.
34. Liu H, Yu L. Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans Knowl Data Eng*. 2005;17(4):491–502. doi:10.1109/TKDE.2005.66.
35. Guyon I, Weston J, Barnhill S, Vapnik V. Gene selection for cancer classification using support vector machines. *Mach Learn*. 2002;46:389–422. doi:10.1023/A:1012487302797.
36. Hu MK. Visual pattern recognition by moment invariants. *IRE Trans Inf Theory*. 1962;8(2):179–87. doi:10.1109/TIT.1962.1057692.
37. Wang J, Perez L. The effectiveness of data augmentation in image classification using deep learning. *Convolut Neural Netw Vis Recognit*. 2017;11:1–8. doi:10.48550/arXiv.1712.04621.
38. Shorten C, Khoshgoftaar TM. A survey on image data augmentation for deep learning. *J Big Data*. 2019;6(1):1–48. doi:10.1186/s40537-019-0197-0.
39. Baltrusaitis T, Ahuja C, Morency L-P. Multimodal machine learning: a survey and taxonomy. *IEEE Trans Pattern Anal Mach Intell*. 2018;41(2):423–43. doi:10.1109/TPAMI.2018.2798607.