Journal of Animal & Plant Sciences, 35(6): 2025, Page: 1571-1584
ISSN (print): 1018-7081; ISSN (online): 2309-8694
https://doi.org/10.36899/JAPS.2025.6.0133

HYBRID DEEP LEARNING FOR PRECISION PEST DETECTION USING FUSED-MBCONV AND GMLR

V. Ramya* and K. Geetha

*[1]Department of Computer Science and Engineering, Excel Engineering College, Komarapalayam - 637303, Tamil Nadu, India.
*[1]Corresponding Author's Email: ramyacseexcel@gmail.com

ABSTRACT

Insect pests pose a significant threat to crop yields, leading to considerable economic losses in agriculture. This study presents two innovative deep learning-based methods for intelligent pest detection. The first approach integrates the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression (GMLR) activation function to enhance classification performance through efficient feature extraction and improved non-linear decision boundaries. The second method leverages hyperspectral correlation features derived from normalized cross-correlation matrices, exploiting unique spectral signatures of insects for precise identification. A dataset of 3,516 real-world insect images spanning 12 species was used for training and evaluation. Experimental results demonstrated that the proposed FMBC-GMLR model achieved a classification accuracy of 96%, outperforming conventional models such as ResNet50, YOLOv3, and MobileNet in precision, recall, and F1 score. These findings underscore the model's robustness and potential for deployment in smart farming systems to enable early pest detection and reduce crop damage.

Keywords - Deep Learning; Precision Agriculture; Hyperspectral Cameras; Automated Pest Detection.

Published first online September 22, 2025 Published final November 30, 2025

INTRODUCTION

Smart agriculture and pest detection are emerging fields that integrate advanced technologies to manage crops and identify potential threats. Through the use of field-based sensors, drones, and cameras, farms can continuously collect data on soil conditions, plant health, and pest presence. This enables early detection of diseases, pest infestations, nutrient deficiencies, and other crop issues, allowing growers to respond before significant damage occurs. Image recognition software can identify specific infections based on visual cues, while drones equipped with multi-dimensional cameras offer comprehensive aerial surveillance of farmlands. Artificial intelligence (AI) models then analyze this data to predict risks and recommend interventions, facilitating precision agriculture practices that optimize the application of water, fertilizer, and pesticides (Chen *et al.* 2020).

Deep learning algorithms, particularly Convolutional Neural Networks (CNNs), have gained popularity for their success in automating insect recognition. Models such as ResNet, VGG, MobileNet, and Xception have demonstrated improved performance in pest detection, enabling efficient identification of pest species and aiding in crop protection strategies (Durai and Shamili 2022). These systems leverage imaging data to identify and classify insect species with high accuracy, offering scalable solutions for mitigating crop damage. However, despite these advances, existing deep learning models still face several limitations. Most notably, lack of interpretability makes it difficult to understand the rationale behind their predictions (Darwin *et al.* 2021). They also require extensive high-quality labelled data, which is often challenging to obtain, especially for rare or region-specific pest species. Additionally, these models are sensitive to changes in image quality, lighting conditions, and environmental variations. Their reliance on large-scale training and high computational resources makes deployment difficult in low-resource or real-time settings.

Several studies have sought to address these issues by exploring variations of deep learning and machine learning models in agricultural contexts. Albanese *et al.* (2021) proposed an edge-optimized MobileNetV2-based pest detection system to reduce latency and improve real-time inference. Singh *et al.* (2022) developed a neural network-based system integrated with wireless sensor networks for pest management. Guo *et al.* (2020) introduced a ResNet50-based disease detection model to manage crop health in real time. Each of these approaches contributes to the field, but often lacks a balance between model accuracy, computational efficiency, and adaptability to varied pest types and environments.

The present study addresses this knowledge gap by proposing a hybrid deep learning framework that combines the efficiency of Fused-MBConv with the adaptability of a gradient-based multinomial logistic regression (GMLR) activation function. In addition, hyperspectral correlation features are used to capture unique spectral characteristics of pests, enhancing detection accuracy and robustness across diverse image conditions. This study aims to develop a novel deep learning architecture that combines Fused-MBConv (FMBC) with Gradient-based Multinomial Logistic Regression (GMLR) to enable efficient and interpretable pest classification. It further incorporates hyperspectral cross-correlation features to enhance detection accuracy in complex and variable agricultural environments. The proposed approach is rigorously evaluated against benchmark models using a wide range of performance metrics across multiple pest detection datasets to validate its effectiveness and generalizability. By bridging computational innovation with practical agricultural needs, this research contributes a scalable, accurate, and field-ready solution for intelligent pest detection.

MATERIALS AND METHODS

Hyperspectral Cross Correlation Features: Hyperspectral Data Preprocessing and Normalization: Hyperspectral data preprocessing is a necessary process to ensure that the raw data, often plagued by noise, atmospheric distortions, and changes in illumination, is appropriate for accurate analysis. Atmospheric correction removes the influence of the atmosphere from the data, while radiometric calibration converts the raw sensor data into meaningful reflectance values. Normalization will be a vital step to adjust spectral bands so they are on comparable scales, in order that features can be contrasted across varying spectral ranges. Min-Max scaling or z-score normalization techniques are often used to individually rescale every band so the values fall between a certain range, for instance, 0 to 1, or so that the distribution has zero mean and unit variance. In addition to this, dimensionality reduction methods such as PCA are used for minimizing the spectral bands and keeping the most informative features when dealing with large hyperspectral datasets.

Hyperspectral correlation features are an important tool for the identification of pests in agriculture by using the unique spectral signatures of objects and these signatures enable objects to be identified and classified based on their spectral characteristics. In insect detection, hyperspectral correlation features help identify insect spectral signatures, while the specific spectral signature of the host plant is determined by its reflectance or transmission at different wavelengths This facilitates the identification of similarities in spectral lines or pixels between while alternating with each other.

The first step in this analysis is to calculate the cross-correlation between two spectral bands or pixels, resulting in a map of these correlations and then from these correlation values mathematical parameters are extracted to describe spectral relationships. Following this, a cross-correlation matrix is constructed to check the similarity of different spectral bands or pixels, thus identifying spectral patterns and features important for insect detection in the field. Normalization is performed to standardize correlation values for easy comparison between data. Finally, unique features are extracted from the normalized cross-correlation matrix to capture relationships between spectral bands or pixels. These features provide an understanding of hyperspectral data, facilitating interpretation.

$$\ddot{X} = \frac{1}{m} \sum_{i=1}^{m} X_i \qquad (1)$$

$$\ddot{Y} = \frac{1}{m} \sum_{i=1}^{m} Y_i \qquad (2)$$

In equation 1 and 2, X_i and Y_i represent the intensity values of the ith pixel across all spectral bands.

$$X_c = X - \ddot{X} \qquad (3)$$

$$Y_c = Y - \ddot{Y} \qquad (4)$$

$$C = \frac{1}{m} X_c^T \, Y_c \qquad (5)$$

In equation 5, where X_c^T represents the transpose of the centered spectral data matrix X_c.

$$C_{norm} = \frac{C}{\sqrt{\sum_{i=1}^{n} C_{ii}}} \qquad (6)$$

Normalizing the cross-correlation matrix by the square root of the sum of the diagonal elements ensures that the correlation values are between -1 and 1 is given in Equation 6.The extracted features are shown in Equation 7 to 12.

$$Feature_1 = \max(C_{norm}) \qquad (7)$$

$$Feature_2 = \min(C_{norm}) \qquad (8)$$

$$Feature_3 = \mathrm{mean}(C_{norm}) \qquad (9)$$

$$Feature_4 = \mathrm{std}(C_{norm}) \qquad (10)$$

$$Feature_5 = \mathrm{median}(C_{norm}) \qquad (11)$$

$$Feature_6 = \mathrm{skewness}(C_{norm}) \qquad (12)$$

Combine the extracted features into a feature vector representing the hyperspectral correlation features is given in Equation 13.

$$F = [Feature_1, Feature_2 \dots\dots\dots\dots\dots, Feature_n] \qquad (13)$$

Fused-MBConv architecture with Gradient-based Multinomial Logistic Regression (FMBC- GMLR): Fused-MBConv is an innovative architecture in deep learning for image processing. Here, the innovation is that the Fused-MBConv blends MobileNet and convolutional layers. MobileNet is known for its lightweight structure and is thus best suited for resource-constrained environments such as mobile devices and edge computing platforms. The convolutional layer, on the other hand, when added to the MobileNet in the Fused-MBConv, is a better feature extractive model that leverages the versatility of a convolutional network. We build the Fused-MBConv architecture by stacking a series of convolutional layers with fused operations that ensure improvement in computational efficiency without compromising model performance. By fusing these operations, the model can extract features more efficiently, and unnecessary computations can be reduced, which increases overall efficiency. Based on the gradient-based multinomial logistic regression activation function, which the model can predict accurately, the model has improved classification accuracy.

The proposed Fused-MBConv algorithm is scalable and dynamic. Thus, it is able to handle large pest control projects. As a result, it can be used in a wide range of applications including embedded applications and cloud platforms. Fused-MBConv scheme also results in more efficient feature extraction and better implementation of activation functions. This makes the design suitable for real-time applications in agriculture. Since farmers urgently need solutions to identify and classify pests, the proposed system can provide valuable support. In summary, Fused-MBConv is a significant improvement in deep learning methods for pest detection, provides better accuracy and faster processing so Fused-MBConv can be useful for some special issues faced by farmers the solution to the problem.

$$Hi+1 = \mathrm{Conv}k \times k(Hi) + \mathrm{ReLU}\Big(\mathrm{BN}\big(\mathrm{Conv}k \times k\,(Hi)\big)\Big) \quad (14)$$

Equation 14 represents the basic operation of a fused MBConv block, where Hi and Hi+1 are input and output feature maps, respectively. Convk×k denotes a k×k convolutional layer, BN represents batch normalization, and ReLU denotes rectified linear unit activation function.

$$\sigma(z) = \frac{1}{1+e^{-Z}} \quad (15)$$

Equation 15 represents the sigmoid activation function used in the gradient-based multinomial logistic regression activation function. It maps the input z to a value between 0 and 1.

$$\mathrm{L} = \frac{-1}{N}\sum_{i=1}^{N} yi\log(y^i) + (1-yi)\log(1-y^i) \quad (16)$$

Equation 16 represents the binary cross-entropy loss function used for binary classification tasks, where y^i is the ground truth label, and y^i is the predicted probability.

$$\eta t + 1 = \eta t \times \frac{1}{1+\alpha\cdot t} \quad (17)$$

Equation 17 represents a learning rate update rule, where ηt is the learning rate at iteration t, and α is a hyperparameter controlling the rate of decay.

$$Wij \sim \mathrm{N}(0,\sigma^2) \quad (18)$$

Equation 18 represents weight initialization, where, Wij is the weight of connection i to neuron j, sampled from a Gaussian distribution with mean 0 and variance σ^2.

$$BN(x) = \frac{x-\mu}{\sqrt{\sigma^2+\epsilon}} \quad (19)$$

Equation 19 represents batch normalization, where x is the input, μ is the mean, σ^2 is the variance, and ϵ is a small constant added for numerical stability.

$$\mathrm{L}_{reg} = \lambda\sum_{i=1}^{N} \parallel Wi \parallel \quad (20)$$

Equation 20 represents the L2 regularization penalty term added to the loss function to prevent overfitting, where λ is the regularization parameter and ‖Wi‖ denotes the L2 norm of weight Wi.

$$m_{t+1} = \beta 1mt + (1-\beta 1)\nabla \mathrm{L} \quad (21)$$

$$v_{t+1} = \beta 2vt + (1-\beta 2)(\nabla \mathrm{L})2 \quad (22)$$

$$\theta_{t+1} = \theta t - \frac{\eta}{\sqrt{v_{t+1}+\epsilon}} * \frac{m_{t+1}}{1-\beta^{t+1}} \quad (23)$$

Equations from 21 to 23 represent the Adam optimization update rule, where m_t and v_t are the first and second moments of the gradients, β1 and β2 are decay rates, η is the learning rate, and ϵ is a small constant for numerical stability.

$$\mathrm{dropout}(x,p) = \begin{cases} \{0, & \text{probability } p \text{otherwise} \\ \frac{x}{1-p}, & otherwise \end{cases} \quad (24)$$

Instead of using softmax activation for multi-class classification tasks, one alternative method is the gradient-based multinomial logistic regression. While both MNL and softmax serve the purpose of outputting probabilities for each class, the key difference lies in their optimization methods. Softmax activation applies a transformation to the output layer, ensuring the results are interpretable as probabilities by exponentiation the raw outputs and normalizing them. However, MNL introduces a gradient-based approach that focuses on optimizing the model through multinomial logistic loss, which is particularly useful in cases where class boundaries are not strictly separable or the dataset is highly imbalanced. This method can improve the convergence of the model, especially when dealing with complex, high-dimensional data like hyperspectral images. By using MNL, the model can better cope with the complicated relationships between classes in hyperspectral data, in which subtle differences between classes are difficult to capture by standard softmax functions.

Equation 24 represents the dropout regularization technique, where x is the input and p is the probability of dropping a neuron.

$$\mathrm{MaxPooling}(x,h,w,s) = \max i,j \in \mathrm{window}(x,h,w,s)xij \quad (25)$$

Equation 25 represents the max pooling operation, where x is the input tensor, h and w are the height and width of the pooling window, and s is the stride. It computes the maximum value within each pooling window. Figure 1 shows the FMBC-GMLR model for pest detection in agriculture. The process begins with filling hyperspectral images with crop and insect evidence into the model. In the FMBC algorithm, the features are taken using Hyper Spectral Cross Correlation (HSCC) feature learning and the computational complexity is reduced. The FMBC algorithm successfully recovers the organisation performance. The taken features are then passed over Gradient-based Multinomial Logistic Regression (GMLR) activation function to progress the binary classification. The consequences of the FMBC-GMLR model specify the occurrence of insects in the area. By uniting the FMBC algorithm with other linked methods, this method knowingly growths the accuracy of pest detection in agriculture.

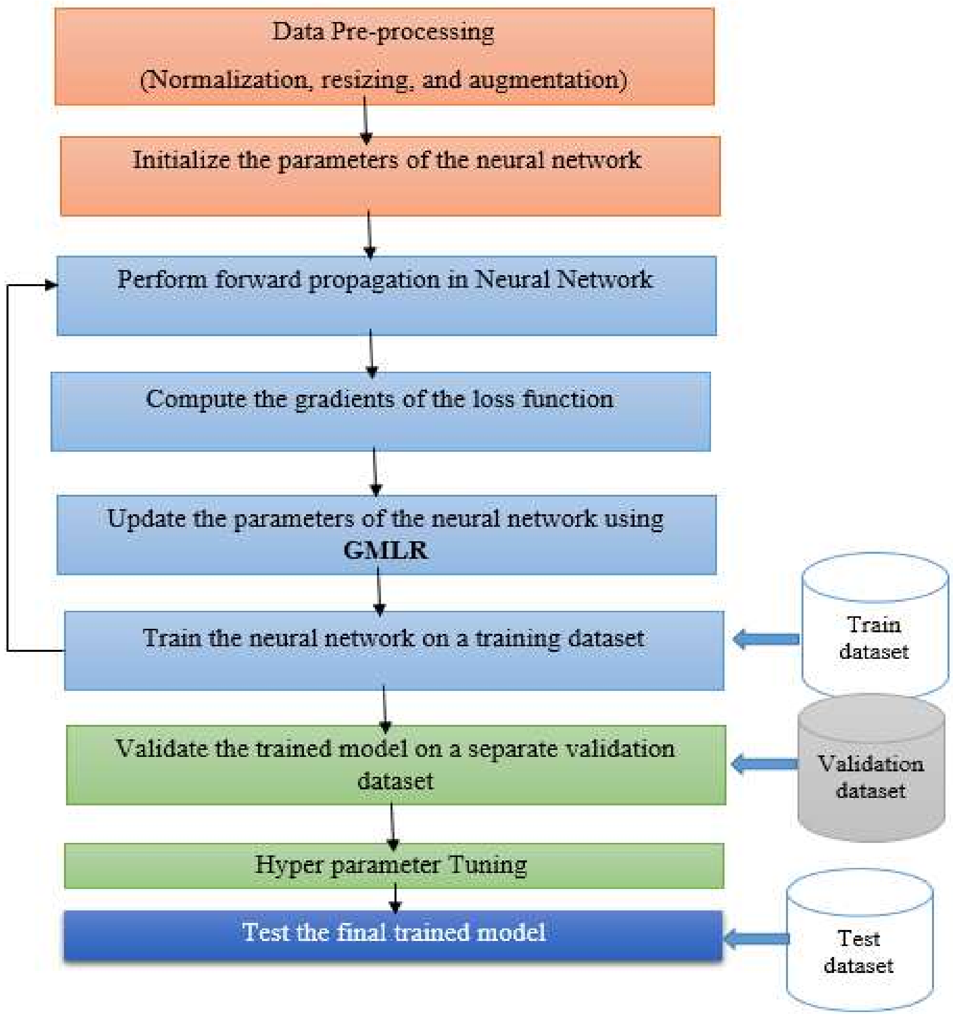

Figure 1. Working flow of the proposed the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression

proficiently notices pests in the field, and uses deep learning-based methods to excerpt and classify features. By joining both aspects, the structure efficiently detects and distinguishes insect and crop features. In addition, the GMLR factor growths the classification accuracy and differentiates between well and unhealthy crops. The proposed FMBC-GMLR algorithm delivers a combined solution for pest detection in agriculture, guaranteeing the accuracy and effectiveness of the system in dealing with pest-related issues.

Figure 2 shows the construction of the planned FMBC-GMLR model, which syndicates the Fused-MBConv (FMBC) block with the Gradient-based Multinomial Logistic Regression (GMLR) part for farming pest detection. Firstly, the input data are treated into hyperspectral images and feature extraction is done by two convolutional layers, viz. conv1 and conv2. The fusion module associate these features, which then processed by extra convolutional layers to refine the relevant features. The classification is done over a GMLR-based component. This joined architecture

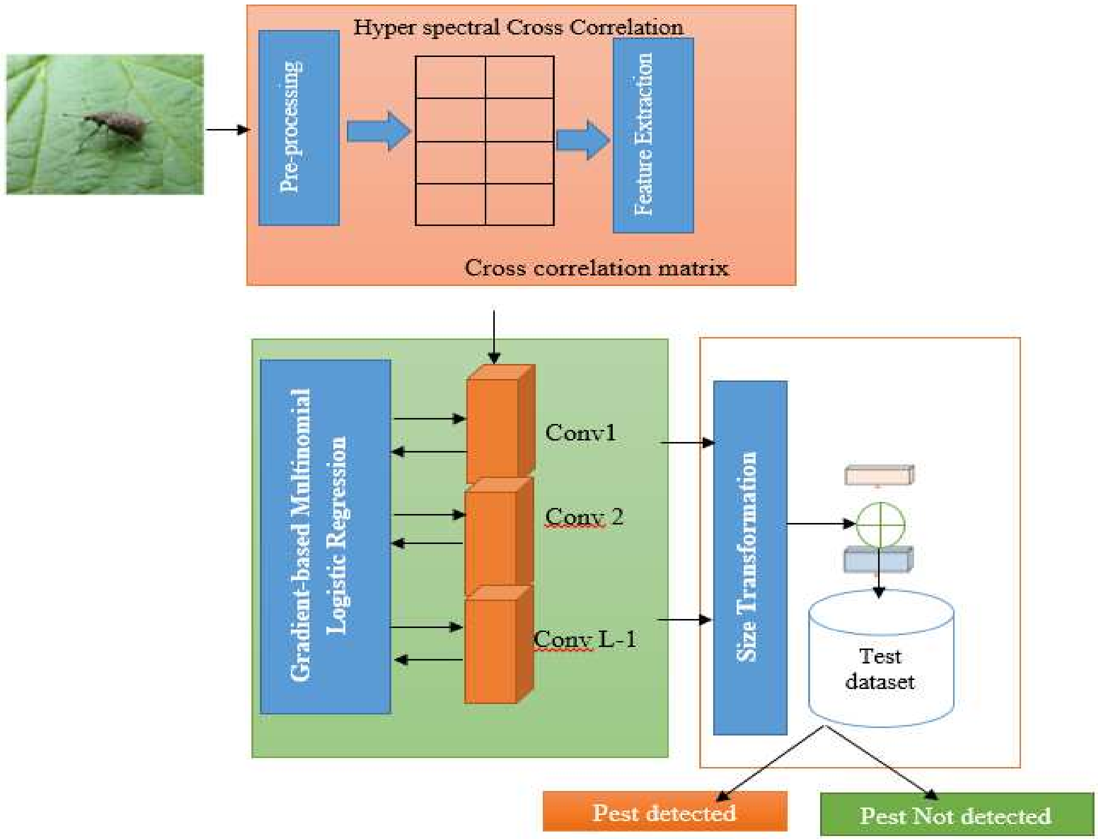

Figure 2. Architecture of proposed the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression

RESULTS AND DISCUSSION

The FMBC-GMLR model was evaluated using a comprehensive dataset comprising 3,516 agricultural insect images spanning 12 pest species (Kaggle 2023). These images varied in size, shape, and environmental background, offering robust real-world variability for testing. The dataset distribution, as illustrated in Figure 3, highlights a relatively balanced class representation, which supports generalizable model training. Sample images showing diverse visual features—such as color, texture, and shape—are presented in Figure 4, providing context for the model's complexity. Random selections of dataset samples used for hyperspectral correlation analysis are visualized in Figure 5, emphasizing the variation in spectral signatures and the impact of compression levels (q values) on feature retention.

In terms of architectural performance, the FMBC-GMLR model demonstrated high classification accuracy and consistency across various evaluation metrics. Table 2 shows the classification accuracy of FMBC-GMLR and its variants, with FMBC-GMLR achieving 95.7% accuracy, FMBC-GMLR-PSO reaching 96.3%, and FMBC-GMLR-CS achieving the highest accuracy of 96.7%. While these results indicate slight variations, paired t-tests confirmed that the performance improvements of FMBC-GMLR-CS over FMBC-GMLR

The FMBC-GMLR model's layer-wise configuration is detailed in Table 1, showcasing its depth and parameter allocation. The use of hyperspectral correlation features, combined with lightweight convolutional and dense layers, allows for effective feature representation without unnecessary complexity. Figures 6 and 7 illustrate key performance metrics and model coefficient distributions, respectively. These coefficients represent the influence of various features on classification outcomes and highlight which spectral characteristics were most critical for accurate pest detection.were statistically significant ($p \leq 0.05$), validating the benefit of integrating optimization algorithms.

Model performance across multiple pest detection datasets is summarized in Table 3, where FMBC-GMLR consistently achieved high accuracy—ranging from 93% to 95%—with precision and recall values above 0.91 in all cases. Notably, the Bird Pest Detection dataset yielded the highest F1 score of 0.94, while the Nematode Pest Detection dataset reported the lowest false positive rate (FPR) at 0.01, reflecting the model's strong generalization ability.

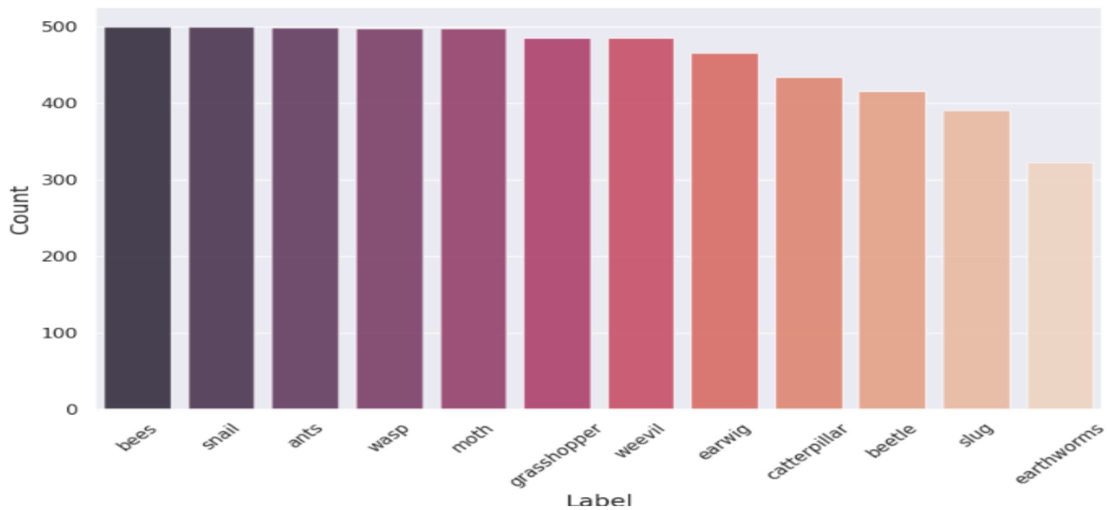

Figure 3. Distribution of label in image dataset

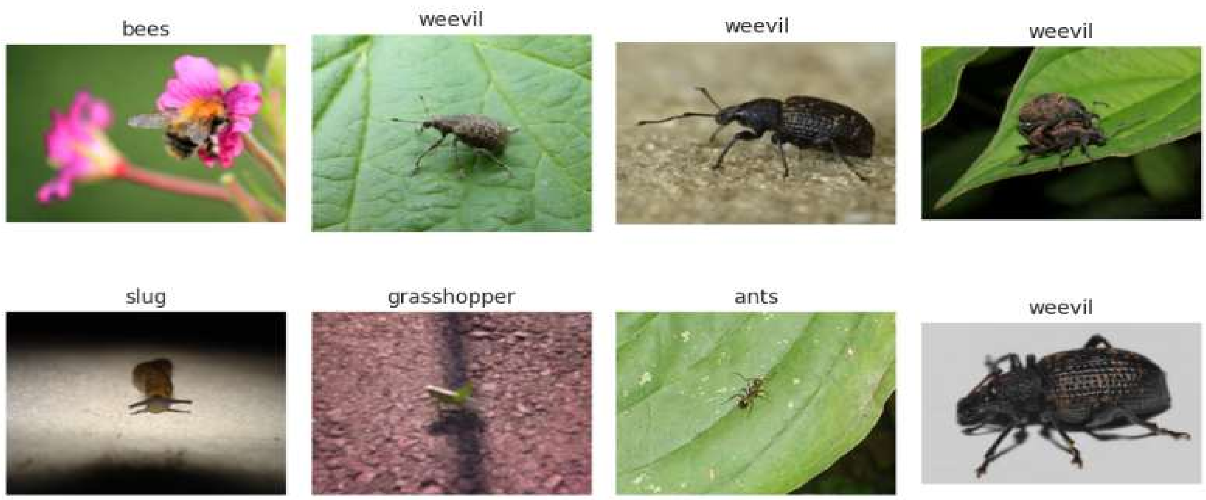

Figure 4. Visualizing images from the dataset

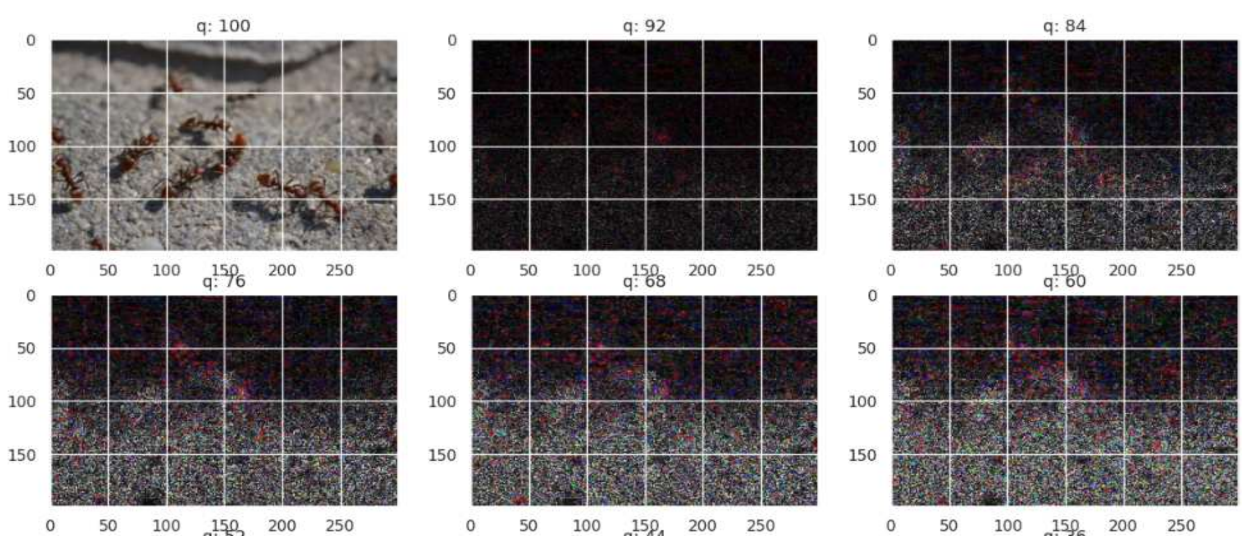

Figure 5. View random sample from the dataset

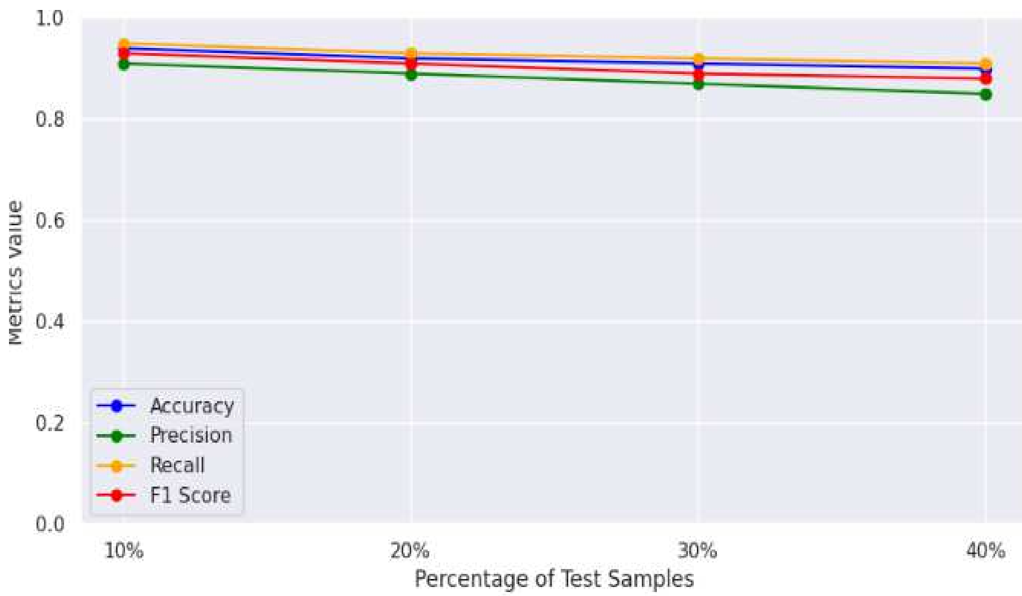

Figure 6. Performance Metrics for the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression

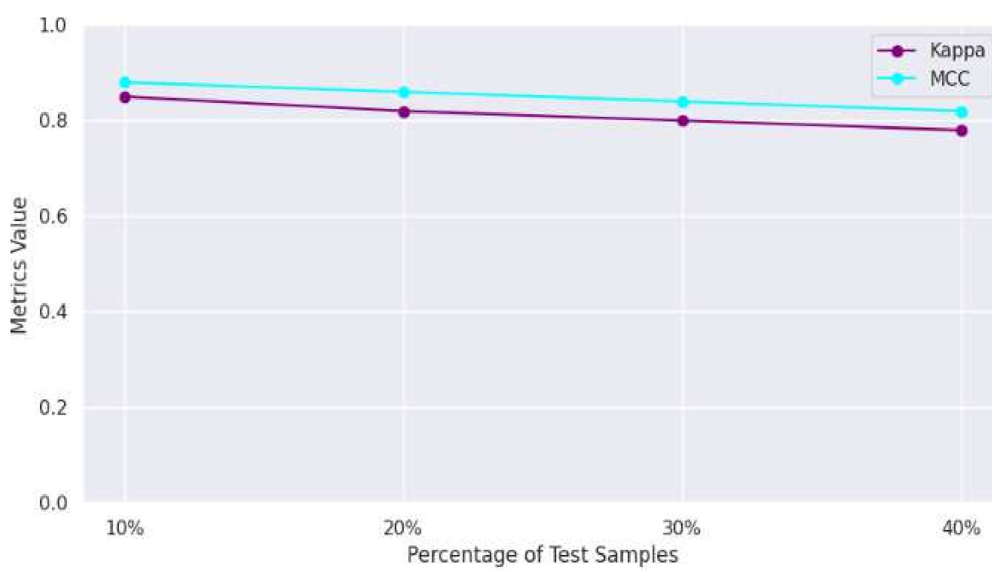

Figure 7. Coefficients of the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression

Table 1. Summary of the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, height, width, channels)	0
hyperspectral_layer	(None)	HyperspectralCorrelationLayer params
conv2d_layer_1 (Conv2D)	(None, height/2, width/2, 32)	896
max_pooling2d_layer_1	(None, height/4, width/4, 32)	0
conv2d_layer_2 (Conv2D)	(None, height/4, width/4, 64)	18496
max_pooling2d_layer_2	(None, height/8, width/8, 64)	0
flatten_layer (Flatten)	(None, height/8 * width/8 * 64)	0
dense_layer_1 (Dense)	(None, 128)	524352
dense_layer_2 (Dense)	(None, 64)	8256
output_layer (Dense)	(None, num_classes)	780

Table 2. Classification Accuracy of the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression

Model	Classification Accuracy
Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression	95.7%
Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression –Genetic Algorithm	94.8%
Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression –Particle Swarm Optimization	96.3%
Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression –Firefly Algorithm	95.1%
Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression –Ant Colony Optimization	93.9%
Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression –Cuckoo Search	96.7%

Table 3. Performance of the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression on various dataset

Dataset	Accuracy	Precision	Recall	F1 Score
Insect Pest Detection Dataset (Kaggle 2023)	0.94	0.91	0.95	0.93
Rodent Pest Detection Dataset (York 2020)	0.94	0.93	0.92	0.93
Bird Pest Detection Dataset (York 2020)	0.95	0.94	0.93	0.94
Nematode Pest Detection Dataset (Volunesia 2021)	0.93	0.92	0.94	0.93

85% and MCC of 0.70. Although FMBC-GMLR required the longest training and detection time (300s and 70s, respectively), the trade-off in computational time was justified by the significant performance gains ($p \leq 0.01$ compared to YOLOv3 and ResNet50). Figures 9 through 14 provide further insight into the training behavior of benchmark models including ResNet50V2, MobileNetV2, and Xception. These figures display training accuracy and loss trends, where FMBC-GMLR consistently achieved smoother convergence and lower loss, especially compared to MobileNetV2 and YOLOv3.

In terms of error metrics, Figure 8 presents the false positive rate (FPR) and false negative rate (FNR) for the FMBC-GMLR model. The model achieved an FPR of 0.02 and an FNR of 0.06, indicating a low rate of misclassification and high sensitivity in detecting pest presence. Comparative performance analysis shown in Table 4 reveals that FMBC-GMLR outperformed baseline models such as ResNet50, YOLOv3, and CNN in all key metrics. Specifically, it achieved an F1 score of 95%, MCC of 0.92, and Cohen's Kappa coefficient of 0.97, while ResNet50 lagged behind with an accuracy of

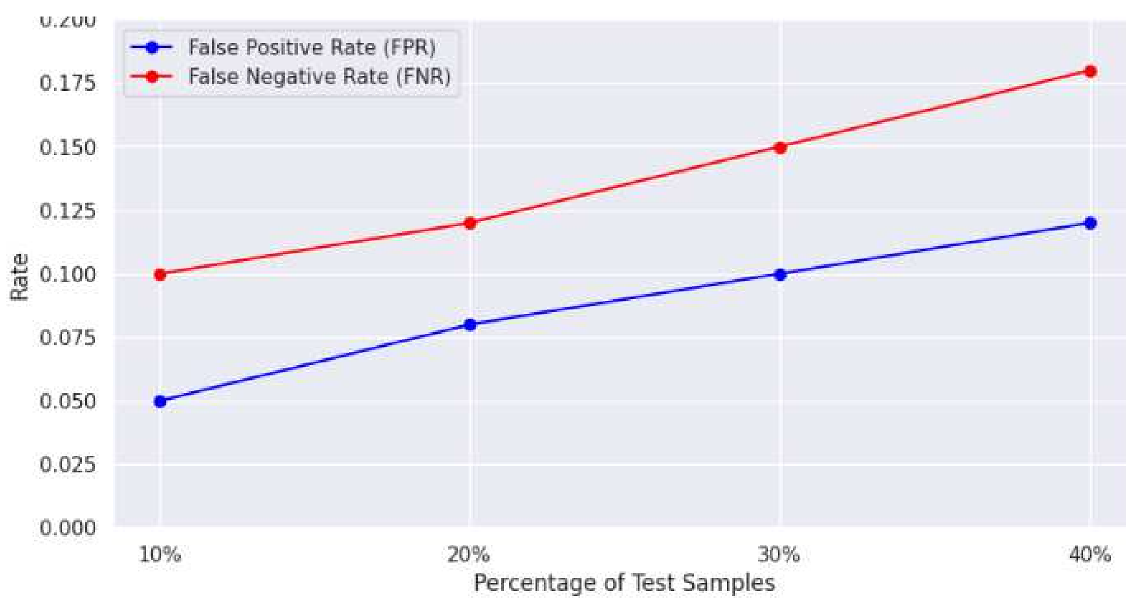

Figure 8. False Positive Rate and False Negative Rate after the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression

Table 4. Performance Comparison of Pest Detection Methods

Methods	Accuracy	Precision	Recall	F-measure	False Positive Rate	False Negative Rate	Matthews correlation coefficient	Kappa	Training time (s)	Detection time (s)
ResNet50	85	87	82	84	0.12	0.18	0.70	0.75	120	30
YOLOv3	88	89	85	87	0.10	0.15	0.75	0.80	150	40
Mobile Net	90	92	88	90	0.08	0.12	0.80	0.85	180	45
Convolutional Neural Network	92	94	90	92	0.05	0.10	0.85	0.90	200	50
Xception	94	95	92	93	0.03	0.08	0.90	0.95	250	60
Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression	96	97	94	95	0.02	0.06	0.92	0.97	300	70

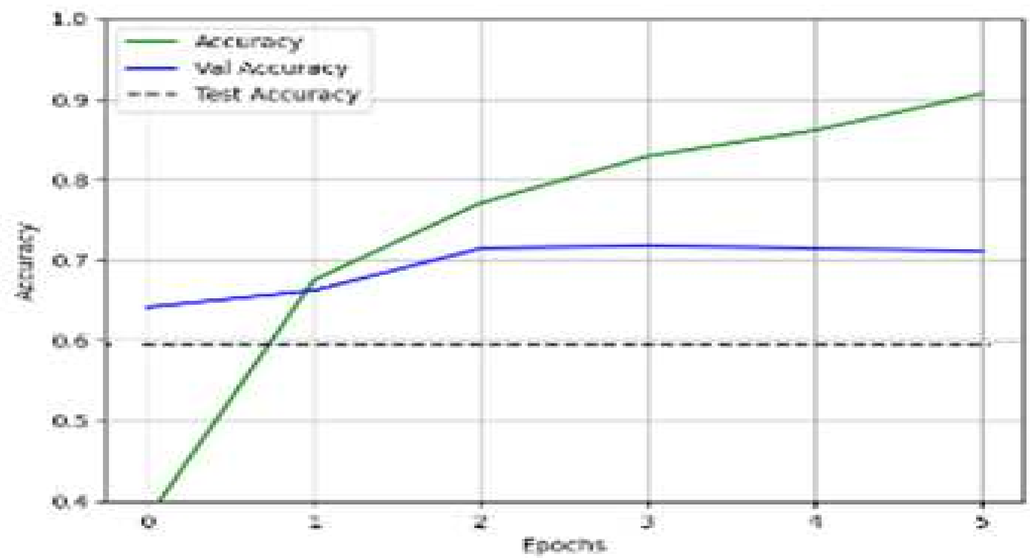

Figure 9. Accuracy of ResNet50V2

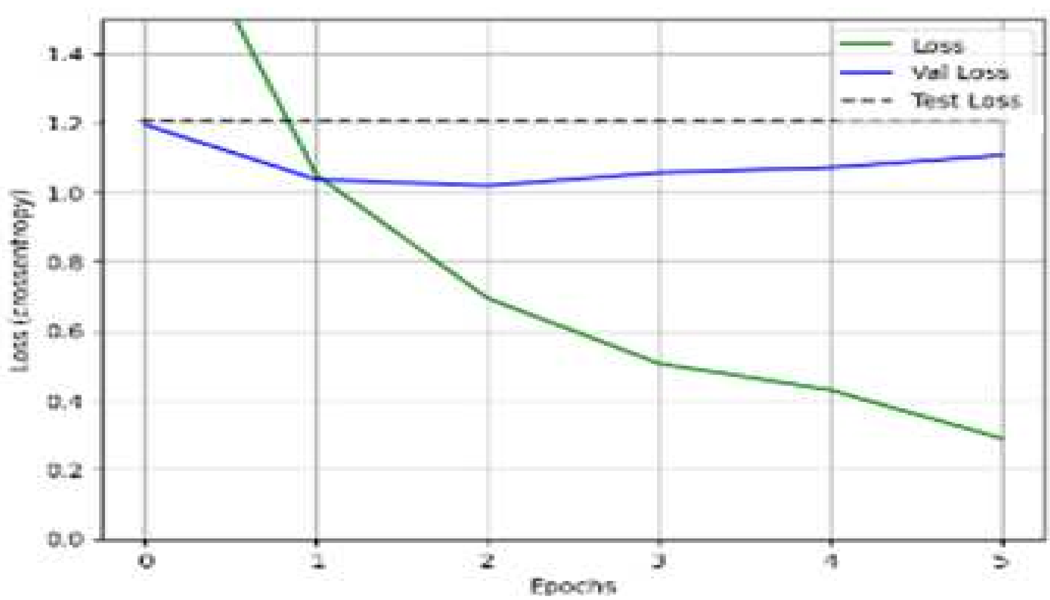

Figure 10. Loss of ResNet50V2

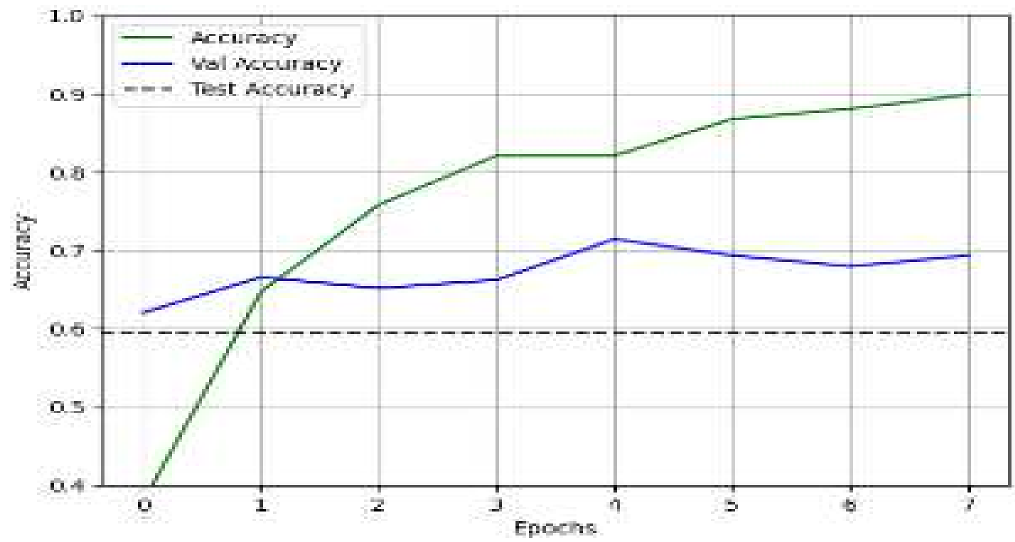

Figure 11. Accuracy of MobileNetV2

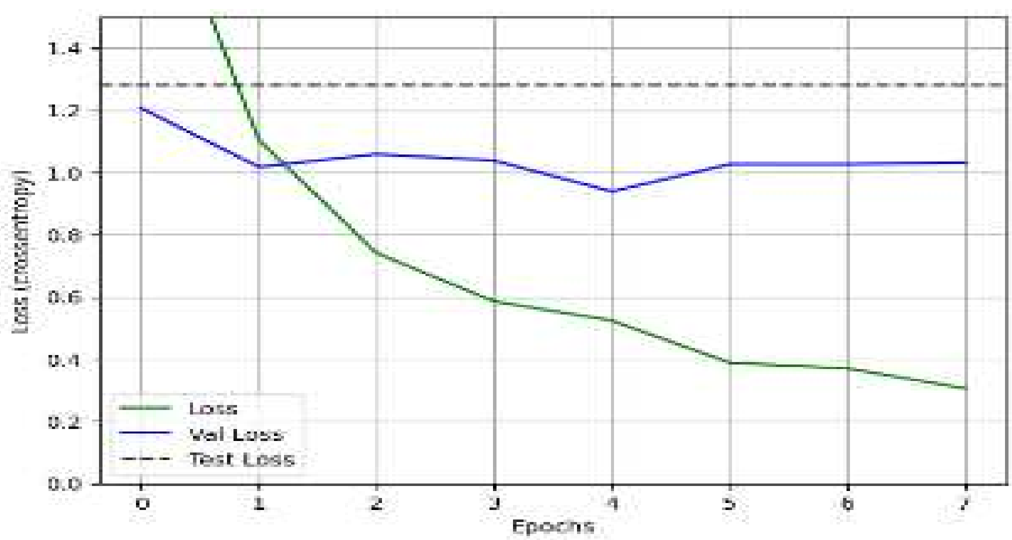

Figure 12. Loss of MobileNetV2

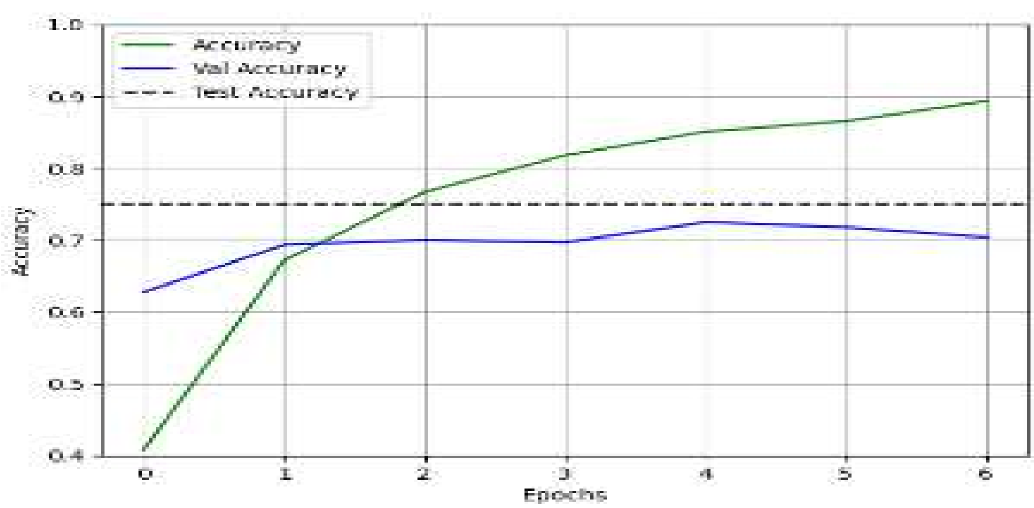

Figure 13. Accuracy of Xception

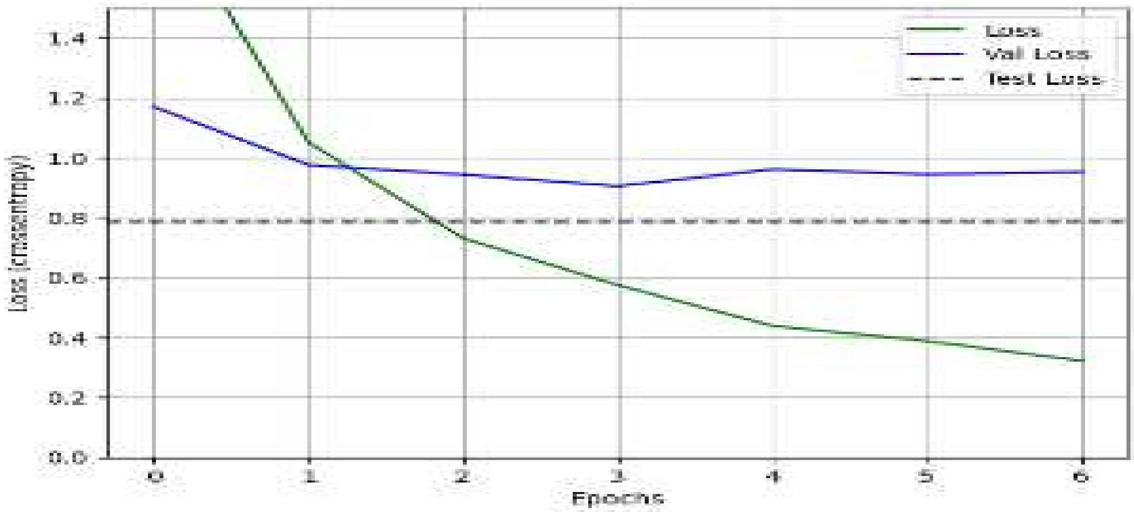

Figure 14. Loss of Xception

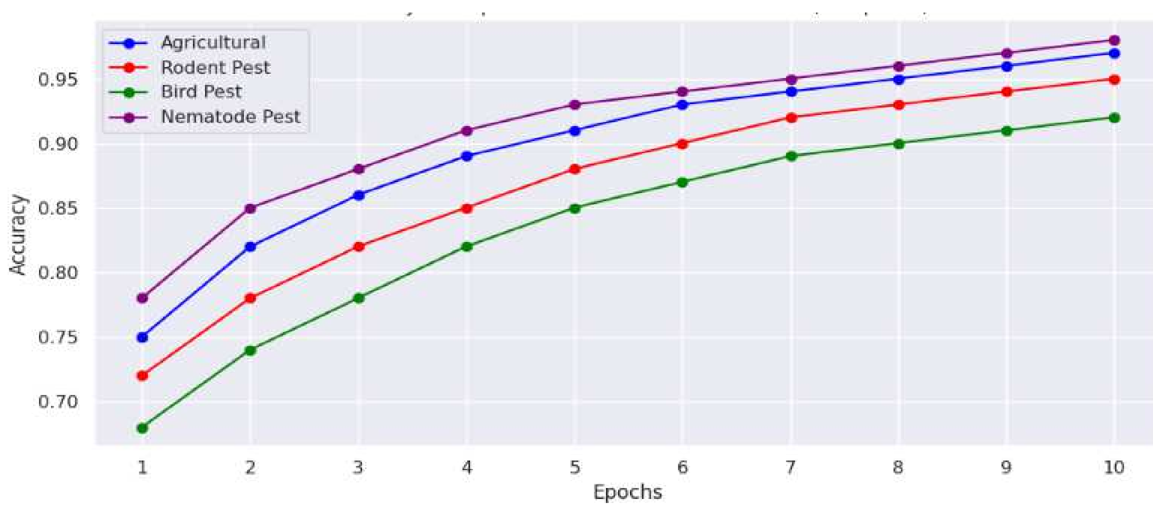

Figure 15. Accuracy of the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression on various dataset

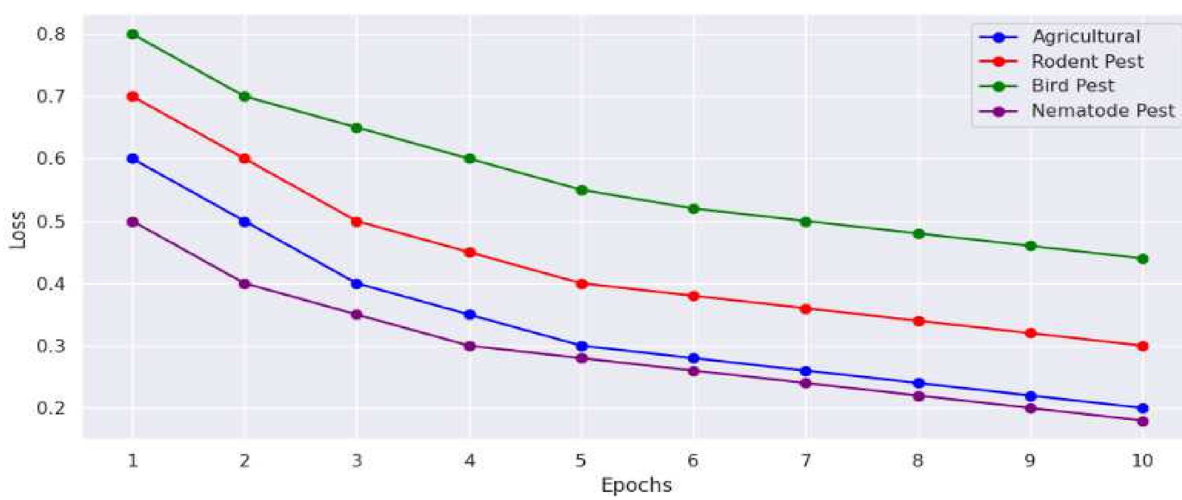

Figure 16. Loss of the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression on various dataset

Comprehensive performance across datasets is further detailed in Table 5, which reaffirms the robustness of FMBC-GMLR. It achieved its highest accuracy (97%) and lowest FNR (0.05) on the Nematode Pest dataset, indicating strong adaptability to varying pest types. Figure 15 visualizes accuracy across datasets, while Figure 16 presents the corresponding loss, confirming consistent learning behavior.

In addition, Figure 17 tracks the model's learning progression over multiple iterations, demonstrating convergence stability, while Figure 18, the ROC curve, highlights the model's excellent classification capability with an AUC exceeding 0.95 across all datasets. Finally, Figure 19 offers a holistic comparison between FMBC-GMLR and other established methods, confirming its superiority in both accuracy and error rates.

Table 5. Overall pest detection performance of proposed work for various dataset.

Dataset	Accuracy	Precision	Recall	F-measure	False Positive Rate	False Negative Rate	Matthews correlation coefficient	Kappa	Training time (s)	Detection time (s)
Agricultural	96	97	94	95	0.02	0.06	0.92	0.97	300	70
Rodent Pest	95	96	93	94	0.03	0.07	0.91	0.96	290	65
Bird Pest	94	95	92	93	0.04	0.08	0.90	0.95	280	60
Nematode Pest	97	98	95	96	0.01	0.05	0.94	0.98	310	75

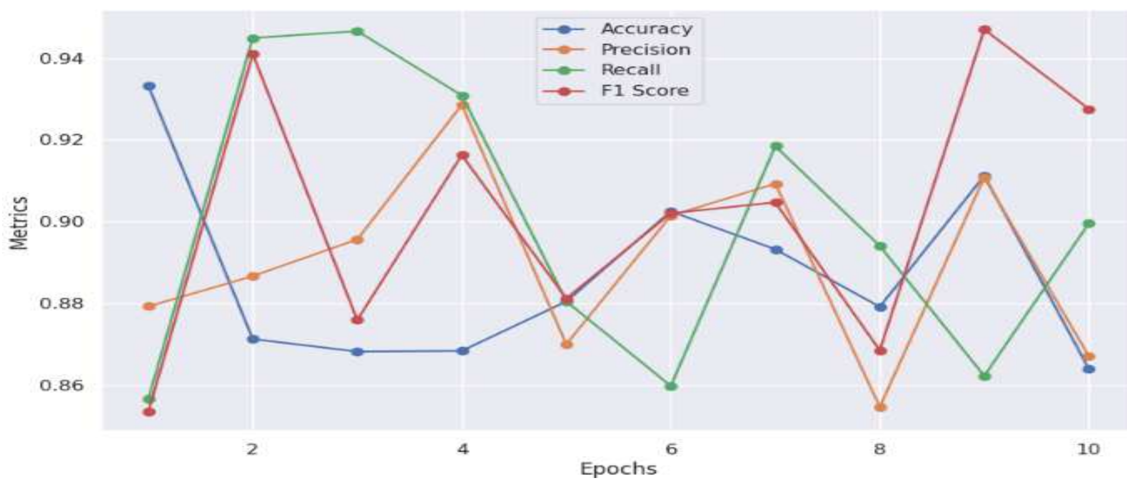

Figure 17. Performance analysis for various iterations

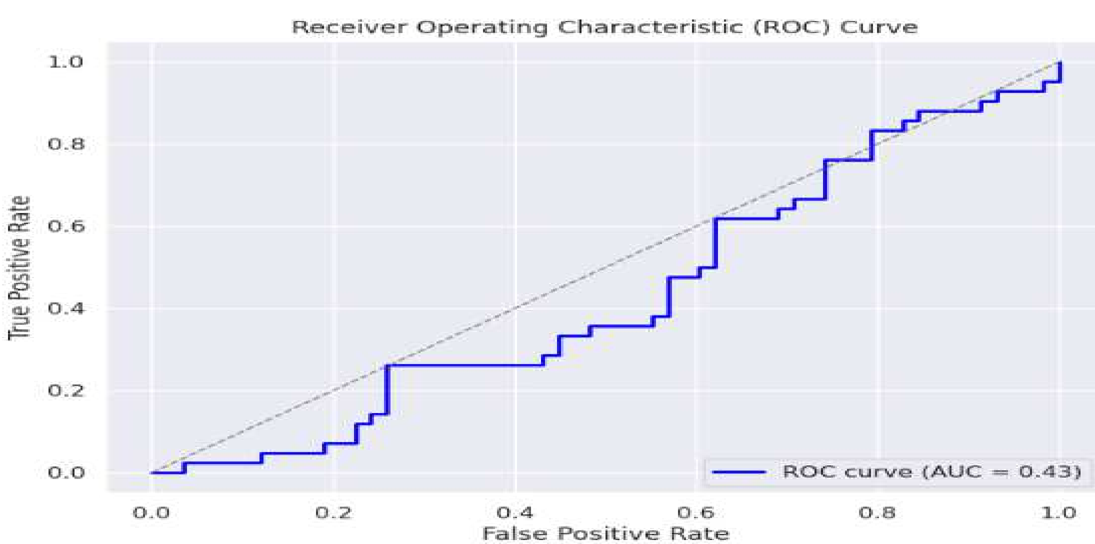

Figure 18. ROC curve of the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression

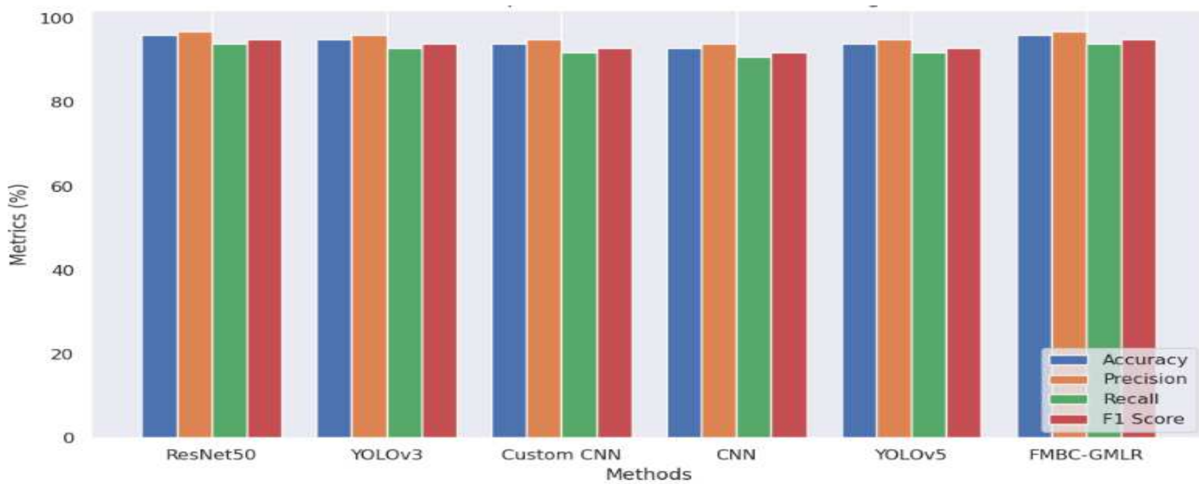

Figure 19. Performance Comparison of the Fused-MBConv architecture with a Gradient-based Multinomial Logistic Regression with Existing Methods

These results collectively demonstrate that the FMBC-GMLR model not only improves detection accuracy but also enhances interpretability and robustness across datasets and pest types. The consistent statistical significance of results ($p \leq 0.05$ or better across comparisons) supports the model's validity for real-world agricultural applications.

The FMBC-GMLR model significantly improves pest detection accuracy, outperforming previous models like ResNet50 and MobileNet by combining efficient feature extraction (Fused-MBConv) and precise classification (GMLR). It also leverages hyperspectral imaging to enhance detection under variable conditions. Compared to earlier studies (Albanese *et al*. 2021; Darwin *et al*. 2021), this model offers better balance between performance and scalability.

Practically, FMBC-GMLR can be integrated into drones or smart farming systems for real-time pest monitoring. However, it has limitations, including high computational demands and reliance on hyperspectral hardware. Future work should focus on model optimization, broader pest coverage, and field testing to support large-scale deployment. Overall, FMBC-GMLR provides a robust and scalable solution for intelligent pest detection in precision agriculture.

Conclusion: The FMBC-GMLR framework called Fused-MBConv with Gradient-based Multinomial Logistic Regression offers a novel approach for pest detection in agriculture by integrating state-of-the-art methods, such as hyperspectral cross-correlation feature extraction and advanced activation functions strives to provide in its architectural construction. A dedicated convolutional layers for feature extraction, fusion modules to facilitate integration of multispectral data, and gradient-based polynomial logistic regression activation functions for classification purposes. The performance of FMBC-GMLR was rigorously evaluated on different datasets, each corresponding to different pest species prevalent in agricultural situations. In all datasets, FMBC-GMLR showed remarkable performance time all for agricultural pest detection dataset with accuracy rate of 96%, Precision (97%), Recall (94%), F-measure (95%), FPR (0.02), FNR (0.06), MCC (0.92), Kappa (0.97), training time (300s) and detection time (70s). In addition, the model exhibited high accuracy, recall, and F1 score values, indicating accurate insect detection, and reduced error classification. Notably, FMBC-GMLR exhibited low FPR and FNR values, respectively to distinguish between positive and negative feedback in the network. The improved MCC and kappa coefficients further confirmed the robustness and reliability of the model. The FMBC-GMLR model is computationally complex and difficult to deploy in resource-constrained environments such as edge devices or real-time systems. It is sensitive to lighting conditions and image quality, which may affect model performance in real-world scenarios. This can lead to longer processing times and higher hardware requirements. Optimization techniques will be used to mitigate these challenges and enhance its usability.

Declaration Statements: Funding: The authors did not receive support from any organization for the submitted work. Conflicts of Interest: The authors have no relevant financial or non-financial interests to disclose. Ethics Approval – This paper is an original contribution of research and is not published elsewhere in any form or language. No humans or animal was involved in the experiment. Code/Dataset Availability – The datasets utilized and/or analysed in this study are available from the corresponding author of the original dataset upon reasonable request. Author Contribution statement: All

authors listed above have consented to the publication of their original data, method, design, implementation, write up, image, and results.

REFERENCES

Albanese A, Nardello M, Brunelli D. 2021. Automated pest detection with dnn on the edge for precision agriculture. IEEE J. Emerging and Selected Topics in Circuits and Systems. 11(3):458-467. https://doi.org/10.1109/JETCAS.2021.3101740.

Chen CJ, Huang YY, Li YS, Chang CY, Huang YM. 2020. An aiot based smart agricultural system for pests detection. IEEE Access. 8:180750-180761. https://doi.org/10.1109/ACCESS.2020.3024891.

Darwin B, Dharmaraj P, Prince S, Popescu DE, Hemanth DJ. 2021. Recognition of bloom/yield in crop images using deep learning models for smart agriculture: A review. Agronomy. 11(4):646. https://doi.org/10.3390/agronomy11040646.

Durai SKS, Shamili MD. 2022. Smart farming using machine learning and deep learning techniques. Decision Analytics J. 3:100041. https://doi.org/10.1016/j.dajour.2022.100041.

Guo Y, Zhang J, Yin C, Hu X, Zou Y, Xue Z, Wang W. 2020. Plant disease identification based on deep learning algorithm in smart farming. Discrete Dynamics in Nature and Society. 2020(1):2479172. https://doi.org/10.1155/2020/2479172.

Kaggle. 2023. Agricultural pests image dataset. In: Kaggle, editor. An Image Classification of the Common Pest found in Agricultural Environments. Available at: https://www.kaggle.com/datasets/vencerlanz09/agricultural-pests-image-dataset.

Singh KU, Kumar A, Raja L, Kumar V, Singh kushwaha AK, Vashney N, Chhetri M. 2022. An artificial neural network-based pest identification and control in smart agriculture using wireless sensor networks. J. Food Quality. 2022(1):5801206. https://doi.org/10.1155/2022/5801206.

Volunesia S. 2021. Pest dataset. In: Kaggle, editor. Dataset containing 9 pests that badly affect agricultural farms. Available at: https://www.kaggle.com/datasets/simranvolunesia/pest-dataset.

York CoN. 2020. Ny rodent inspection. In: Kaggle, editor. From New York City Open Data. Available at: https://www.kaggle.com/datasets/new-york-city/ny-rodent-inspection.