RESEARCH ARTICLE Legume Research- An International Journal, Volume 49 Issue 4: 692-700 (April 2026)

Evaluation of a Convolutional Neural Network-based Model for Accurate Detection and Multi-class Classification of Economically Significant Soybean Pests using RGB Image Data

Pragalbh Sharma[1], Ashish Kulkarni[2], Shalini Sharma[3], Pooja Kulkarni[4], Shweta Karsauliya[5], Sandeep Dongre[6]

10.18805/LR-5622

ABSTRACT

Background: Soybean (*Glycine max*) is a vital legume crop cultivated worldwide for its high protein and oil content. However, its productivity is significantly affected by various insect pests that damage leaves, stems and pods during different growth stages. Pests like aphids, armyworms and bollworms can cause substantial yield losses if not detected and controlled promptly. Manual pest monitoring through field scouting remains the dominant practice but is labour-intensive, inconsistent and prone to human error, particularly in large-scale farming systems. The integration of artificial intelligence, particularly deep learning, offers a scalable solution for automating pest detection.

Methods: This study presents a convolutional neural network (CNN) model designed to detect and classify seven major insect pests commonly found in soybean crops. A dataset consisting of 2,450 RGB images was collected from online resources. The images were preprocessed, augmented and split into training, validation and test sets. The CNN was built using six convolutional layers with ReLU activations, followed by max-pooling layers, a fully connected dense layer and a softmax output layer for classification. The model was trained for 100 epochs using the Adam optimizer and sparse categorical cross-entropy loss.

Result: The final model achieved an overall accuracy of 96.88% on the test set. Class-wise evaluation showed high classification matrices across all categories. ROC-AUC values reached 1.00 for all pest classes, indicating excellent classification performance. Precision-recall curves also showed high average precision scores, confirming the model's reliability. These results demonstrate the model's potential for practical application in real-time pest monitoring systems used in precision agriculture.

Key words: Convolutional neural network, Image classification, Pest detection, Precision agriculture, Soybean pests.

INTRODUCTION

Soybean (*Glycine max*) is a globally important crop, with over 350 million metric tons produced in 2023, led by the United States, Brazil and Argentina. India ranks as the fifth-largest producer, with an average annual harvested area of around 13.5 million hectares and production of approximately 12.6-12.9 million metric tons, though yields remain below the global average (~1.15 t/ha in India vs. ~3.3 t/ha globally) (FAOSTAT, 2023; Nargund *et al*., 2024). Major soybean-growing states in India include Madhya Pradesh, Maharashtra and Rajasthan, where yield gaps persist due to insect pests, suboptimal agronomic practices and limited access to advanced technologies (ISWS, 2024).

Globally, insect pests pose a major threat to soybean production, causing an estimated 30-60% yield loss if not managed effectively (Heinrichs and Muniappan, 2019). Key pests include aphids (*Aphis glycines*), stem borers (*Etiella zinckenella*), bollworms (*Helicoverpa armigera*), armyworms, mites, grasshoppers and beetles (Gaur and Mogalapu, 2018; Natukunda and MacIntosh, 2020). In Indian soybean fields, pest detection is challenging due to dense crop canopies and overlapping pest symptoms. Mixed infestations and variable lighting further complicate detection. Background clutter under natural field conditions makes accurate visual identification particularly difficult for

[1]Institute of Business Management, GLA University, Mathura-281 406, Uttar Pradesh, India.
[2]Department of Computer Science and Applications, School of Computer Science and Engineering, Dr. Vishwanath Karad MIT World Peace University, Kothrud, Pune-411 038, Maharashtra, India.
[3]Department of Language, Culture and Society, SRM Institute of Science and Technology, Delhi NCR Campus, Modinagar, Ghaziabad-201 204, Uttar Pradesh, India.
[4]Vishwakarma University, Pune-411 048, Maharashtra, India.
[5]Department of Urban Planning, Indira Gandhi National Open University, New Delhi-110 068, India.
[6]Symbiosis Institute of Business Management, Constituent of Symbiosis International (Deemed University), Nagpur-440 008, Maharashtra India.

Corresponding Author: Pragalbh Sharma, Institute of Business Management, GLA University, Mathura-281 406, Uttar Pradesh, India. Email: pragalbh.sharma@gla.ac.in
ORCIDs: https://orcid.org/0000-0001-6861-3485, https://orcid.org/0000-0003-4771-2198, https://orcid.org/0000-0002-9117-6954, https://orcid.org/0000-0003-4596-5132, https://orcid.org/0009-0004-2227-5766, https://orcid.org/0009-0009-1014-1177

How to cite this article: Sharma, P., Kulkarni, A., Sharma, S., Kulkarni, P., Karsauliya, S. and Dongre, S. (2026). Evaluation of a Convolutional Neural Network-based Model for Accurate Detection and Multi-class Classification of Economically Significant Soybean Pests using RGB Image Data. *Legume Research.* **49(4):** 692-700. doi: 10.18805/LR-5622.

Submitted: 08-12-2025 **Accepted:** 25-03-2026 **Online:** 06-04-2026

small and marginal farmers. Traditionally, pest monitoring has depended on manual field scouting and expert consultation, which are time-consuming, labour-intensive and often error-prone (Venkatasaichandrakanth and Iyapparaja, 2024; Cho, 2024). Several studies have applied AI and deep learning techniques for disease detection in agriculture (Hai and Duong, 2024; Maltare *et al*., 2023; Bagga *et al*., 2024; AlZubi, 2023). These methods have shown promising results but often lack region-specific adaptation and practical deployability for smallholder farmers.

To overcome these limitations, researchers began developing traditional image processing techniques that manually extract features such as colour, shape and texture to detect pests from images (Kasinathan and Uyyala, 2021). These methods have been applied specifically to soybean crops, demonstrating the potential for identifying key pests under controlled conditions (Park *et al*., 2023). The evolution of deep learning, particularly Convolutional Neural Networks (CNNs), has significantly improved pest detection accuracy by automating feature extraction and enabling robust spatial analysis (Xiang *et al*., 2023; de Melo Lima *et al*., 2024). For example, Agarwal *et al*. (2023) demonstrated the superior performance of the EfficientNetB3 model in pest image classification, outperforming traditional deep learning networks in both accuracy and efficiency.

Building on CNNs, transformer-based architectures have further enhanced pest detection by capturing long-range dependencies and global contextual relationships (Tang *et al*., 2023; Zhang and Lv, 2024). Real-time intelligent systems are also emerging. Emerging AI-based pest detection systems, such as YOLOv8 integrated with language models (Şahin *et al*., 2025) and LSTM-based classifiers (Bhoi and Sharma, 2025), demonstrate high accuracy and real-time capabilities, but most remain computationally intensive and are not yet adapted for soybean fields or resource-constrained devices. Most existing deep learning-based pest detection systems rely on computationally heavy architectures. This limits their deployment on resource-constrained devices such as smartphones, edge cameras and low-power embedded systems commonly used in Indian farming environments.

Despite these advancements, there is a noticeable research gap in the development of lightweight, custom CNN models tailored specifically for classifying multiple soybean pests using RGB images under Indian agro-ecological conditions. Existing models often lack regional adaptability, real-time deployability, or the scalability needed for practical precision farming. Therefore, there is a pressing need for efficient, field-ready and high-performing models that can support timely pest diagnosis and integrated pest management strategies in India.

This study proposes a custom CNN model trained on 2,450 labelled RGB images to classify seven economically important soybean pests: aphids, armyworms, beetles, bollworms, grasshoppers, mites and stem borers. The goal is to develop a reliable, scalable and field-deployable pest detection system that enhances precision agriculture practices, particularly in pest-affected and yield-gap regions across India.

MATERIALS AND METHODS

Dataset collection and classes

This study focuses on classifying common insect pests that significantly affect soybean crops. The dataset used contains 2,450 RGB images, categorized into seven distinct pest classes: *aphids, armyworm, beetle, bollworm, grasshopper, mites* and *stem borer* (Fig 1). The images were organized in a directory structure where each subfolder represented a pest class. The dataset was obtained from the Mendeley database (Shinde and Attar, 2024), which

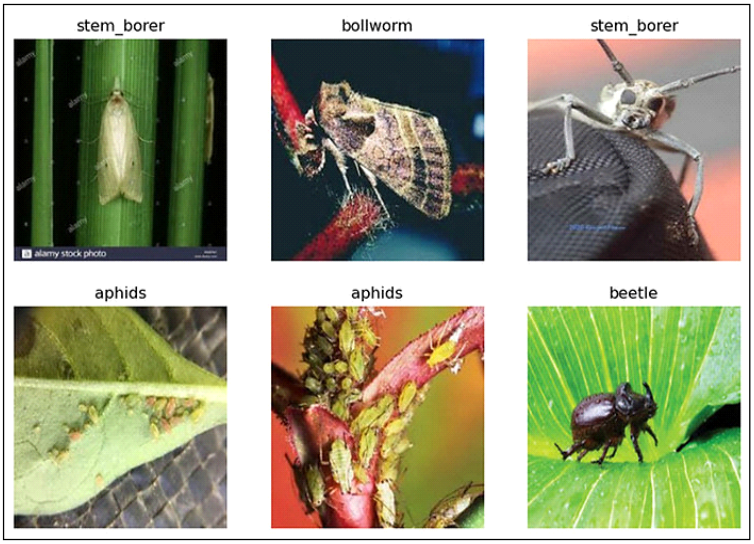


Fig 1: RGB images from the dataset.

provides high-resolution images of soybean leaves with pest attacks from the Maharashtra region in India.

The dataset was loaded using the image_dataset_ from_directory() function from the TensorFlow Keras API. The dataset was loaded with parameters: image size of 256×256 pixels, batch size of 32 and shuffling enabled with a random seed of 123. The shape of each image tensor is:

$$I \in R^{256 \times 256 \times 3}$$

TensorFlow detected the seven classes successfully and mapped each image to a corresponding numeric label (from 0 to 6).

Data partitioning and preprocessing

The dataset was divided into three subsets: training, validation and testing, following an 80:10:10 split ratio. Given the total number of images N=2450, the training set comprised 80% of the data, resulting in N_{train}= 0.8× 2450=1960 images. The validation set accounted for 10% of the dataset, yielding N_{val}= 0.1×2450=245 images. Similarly, the test set also contained 10%, with N_{test}=0.1×2450=245 images. This partitioning ensured that the model had sufficient data for learning while also providing separate sets for tuning and evaluating performance. A custom function using the take() and skip() methods splits the tf.data.Dataset object accordingly. Each partition was then prepared using TensorFlow's cache(), shuffle() and prefetch() methods to enhance performance and minimize I/O bottlenecks. The training set was shuffled with a buffer size of 1000. Prefetching allowed parallel loading and processing, which improved training efficiency.

Data augmentation techniques

To avoid overfitting due to the limited size of the dataset, data augmentation was applied using the tf. keras. Sequential API. Random transformations were added to the training data pipeline to increase image diversity synthetically. The operations included: Random horizontal and vertical flipping, Random rotation within 0.2 radians. The image augmentation pipeline is defined as:

data_augmentation= tf.keras. Sequential ([layers. Random Flip ("horizontal_and_vertical"), layers. Random Rotation (0.2),])

These transformations were only applied to the training dataset. Validation and test sets remained untouched to ensure unbiased evaluation.

Model architecture

A deep convolutional neural network (CNN) was developed using the TensorFlow Keras Sequential API (Fig 2). This architecture was designed to extract spatial features from pest images and classify them into the appropriate categories. The input images were first resized and normalized using a rescaling operation. Each pixel value (I) was divided by 255 to bring it within the range [0, 1]. Normalization ensures consistent pixel scaling, which improves the numerical stability and convergence speed during training (Faye *et al*., 2024).

The model consisted of six convolutional layers. The number of filters increased progressively through the network: 64, 64, 64, 256, 256 and 512. The selection of the number of layers and filters was based on prior experimentation to balance feature extraction capability with computational efficiency for resource-constrained devices. Each convolutional layer used a kernel of size 3×3, followed by the Rectified Linear Unit (ReLU) activation function. The ReLU activation is defined as:

$$y = \max(0, w * x + b)$$

Where,

w= The convolution filter.

x= The input feature map.

b= The bias term.

This non-linearity introduces sparsity and reduces the vanishing gradient problem (Teuwen and Moriakov, 2019). Each convolutional layer was followed by a max pooling layer to reduce the spatial dimensions of the feature maps. Max pooling selects the maximum value from each region R in the feature map, as defined by:

$$y = \underbrace{\max}_{(i,j) \in R} x_{i,j}$$

This downsampling operation reduces computation and controls overfitting by providing translational invariance (Suárez-Paniagua and Segura-Bedmar, 2018). Following the convolutional and pooling layers, a flattening layer was applied to convert the 3D output into a 1D vector. This flattened vector was then passed to a fully connected dense layer with 64 neurons and ReLU activation. The final output layer used a softmax activation function to classify the image into one of the seven pest classes. The softmax function outputs a probability distribution across the classes and is defined as:

$$\hat{y}_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}}, \quad \text{for } i = 1, 2, \ldots, C$$

Here,

C=7= The number of pest classes.

z_i= The logit (raw score) for class i.

The class with the highest probability is selected as the predicted label.

Model compilation and training

The model was compiled with the Adam optimizer and sparse categorical cross-entropy loss function, which is suitable for multi-class classification problems where labels are provided as integers. The loss function is defined as:

$$L = -\log(\hat{y}_y)$$

Here,

Y= The true class label.

$\hat{y}$= The predicted probability for class (y).

The model was trained for 100 epochs on the training dataset. The number of epochs was selected based on

early experimentation to ensure convergence while preventing overfitting. Each epoch involved forward propagation, loss calculation, backpropagation and weight updates. The training history was recorded, including both training and validation accuracy and loss per epoch. This allowed for monitoring the convergence behaviour and detecting overfitting or underfitting.

Justification of hyperparameters

The CNN architecture was designed with six convolutional layers, progressively increasing filter sizes (64, 64, 64, 256, 256, 512) to capture low- to high-level features effectively from soybean pest images. This design balances model complexity and computational efficiency, ensuring sufficient representational capacity without overfitting. The choice of 64 neurons in the dense layer and ReLU activation was based on common best practices for multi-class image classification, providing non-linearity and mitigating vanishing gradient issues. A training duration of 100 epochs was selected after preliminary experiments, which showed stable convergence of both training and validation accuracy, ensuring robust learning while avoiding unnecessary computational overhead.

Hardware and software environment

The experiment was carried out on a personal computer configured with an Intel® Core™ i5-11320H processor running at 3.20 GHz and 16 GB of RAM. The operating system used was Windows 10. All development and model training tasks were performed using Python 3.11. TensorFlow version 2.15.0 served as the primary deep learning library. The implementation was done in Jupyter Notebook, accessed through the Anaconda distribution. Although GPU acceleration was not utilized in this setup, the model and code are compatible with GPU-enabled environments, which can significantly reduce training time for larger datasets or more complex models. The available hardware was sufficient for training and evaluating the model within a reasonable duration. Although GPU acceleration was not utilized in this setup, the model and code are compatible with GPU-enabled environments, which can significantly reduce training time for larger datasets or more complex models.

Evaluation metrics

The model's performance was evaluated using standard classification metrics.

$$\text{Precision (P)} = \frac{TP}{TP + FP}$$

$$\text{Recall (R)} = \frac{TP}{TP + FN}$$

$$F1 = 2 \cdot \frac{P \,.\, R}{P + R}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Here,
TP= True positive.
FP= False positive.
TN= True negative.
FN= False negative.

A confusion matrix was generated to examine misclassifications between classes. Receiver Operating Characteristic (ROC) curves were plotted for all seven classes using a One-vs-Rest strategy. For this, the true labels were binarized and compared against the model's predicted probabilities. The Area Under the Curve (AUC) was calculated for each class.

$$AUC = \int_0^1 TPR\,(f)\,df$$

Here,
TPR= The true positive rate.
F= The false positive rate.

Precision-recall curves were also generated to examine the trade-off between precision and recall (r) at various thresholds. The area under the precision-recall curve, known as average precision (AP), was computed for each class.

$$AP = \int_0^1 \text{Precision}\,(r)\,dr$$

Finally, a prediction function was used to compute the class label and confidence score for each test image.

RESULTS AND DISCUSSION

The model was trained for 100 epochs (Fig 3). During the first epoch, the training accuracy was 14.38% and the validation accuracy was 11.16%, indicating that the model had not yet learned meaningful features. By epoch 25, the training accuracy increased to 81.80% and validation accuracy reached 84.38%, showing substantial learning progress.

At epoch 50, training accuracy rose to 97.60%, while validation accuracy improved to 92.86%, with a significant reduction in both training and validation loss. The model achieved its highest validation performance at epoch 75, where validation accuracy reached 100% and validation loss dropped to 0.0087. The final epoch (epoch 100) showed a training accuracy of 96.37% and validation accuracy of 98.21%, with low loss values of 0.1038 (training) and 0.0681 (validation). The model demonstrated consistent improvement in learning and generalization across epochs. The gap between training and validation metrics remained small, suggesting minimal overfitting. These results indicate that the CNN effectively learned to classify pest images and performed reliably on unseen data.

Fig 4 shows the confusion matrix for pest classification on the test dataset. The model classified aphids correctly in 48 out of 49 cases, with one image misclassified as stem borer. For armyworm, 39 images were correctly identified and one was misclassified as grasshopper. All beetle images were classified correctly, showing perfect recognition. In the case of bollworm, 34 were predicted

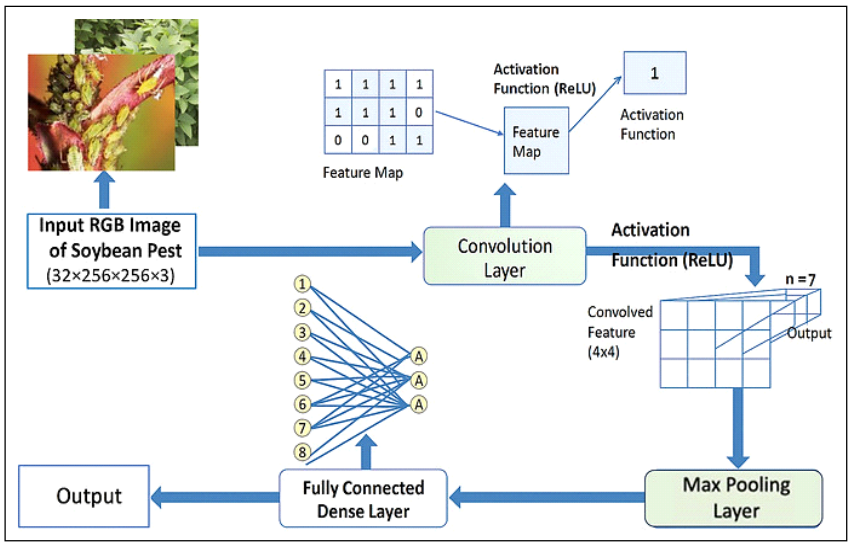


Fig 2: Workflow of the custom CNN architecture for soybean pest classification using RGB images.

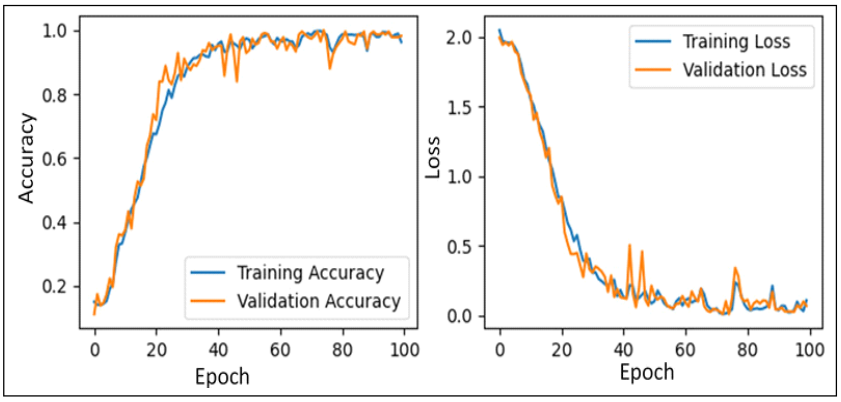


Fig 3: Training and validation accuracy and loss over 100 epochs.

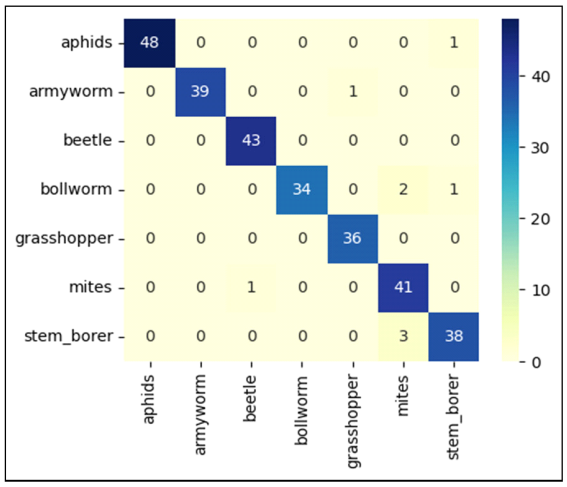


Fig 4: Confusion matrix for pest classification.

correctly, while two were confused with mites and one with stem borer. All grasshopper samples were correctly identified with no misclassification. The model predicted mites correctly for 41 out of 42 images, with one confused as beetle. For stem borer, 38 images were correctly classified and three were wrongly labelled as mites. Overall, the confusion matrix shows that the model performed well across all seven classes with few misclassifications.

Table 1 presents the classification metrics for each pest class. The CNN model showed strong performance in identifying aphids, with a precision of 1.0000 and recall of 0.9796. The F1-score reached 0.9897, indicating high accuracy. These results show the model could distinguish aphids from other pests with minimal false negatives. Armyworm detection was also reliable, achieving a precision of 1.0000 and recall of 0.9750. The F1-score of 0.9873 suggests that the model handled this class with few classification errors. For beetles, the recall reached 1.0000, while precision was slightly lower at 0.9773. The model successfully identified all beetle samples, giving an F1-score of 0.9885. This implies clear feature representation for this class.

The bollworm class had perfect precision (1.0000) but slightly lower recall at 0.9189, with an F1-score of

0.9577. A few bollworms were misclassified, possibly due to similarity with stem borers. Grasshopper identification was nearly flawless. With a precision of 0.9730 and recall of 1.0000, the F1-score stood at 0.9863. The model recognized all test samples from this class correctly. The mites class showed the lowest precision (0.8913) among all. However, its recall was high at 0.9762, giving an F1-score of 0.9318. This suggests the model confused mites with visually similar classes like beetles. In the case of stem borers, the model achieved a precision of 0.9500 and recall of 0.9268, resulting in an F1-score of 0.9383. A few samples may have been classified as mites, but performance remained acceptable. The overall accuracy across all pest classes was 96.88%. Both macro and weighted averages remained high, indicating balanced model performance without favouring any single class.

Fig 5 (a) shows the ROC curves for each pest class. All classes achieved an Area Under the Curve (AUC) of 1.00, which indicates perfect discrimination ability. While all classes achieved an Area Under the Curve (AUC) of 1.00, indicating perfect discrimination, this result should be interpreted cautiously. Such perfect scores may reflect a high model capacity relative to the dataset size and variability, potentially indicating overfitting or limited diversity in the test set. Future work with larger and more diverse datasets is recommended to confirm the generalizability of the model under varied field conditions.

The model was able to distinguish between each pest and all others with zero false positives. ROC curves near the top-left corner suggest strong classification performance. This result confirms that the model learned highly separable features for all seven pest categories. Fig 5(b) presents the PR curves for all pest classes. The precision and recall values remained consistently high across the classes. Aphids, armyworm, beetle, bollworm and grasshopper showed average precision (AP) values of 1.00, indicating both low false positives and low false negatives. Mites and stem borer each achieved an AP of 0.99, slightly below perfect due to a few misclassifications. High AP values indicate that the model is highly confident and accurate, even with class imbalance.

Fig 6 presents predictions made by the trained CNN model on test images from each pest class. The model correctly identified grasshopper with a confidence of 90.37%. Although slightly lower than other predictions, this score still indicates a strong match. The stem borer and beetle samples were classified with 100% confidence, suggesting the model has learned distinct features for these categories. For armyworm, the confidence reached 99.77%, reflecting the high-quality visual features in the test sample. Mites were detected with 99.29% confidence, which supports the model's ability to recognize even small and visually complex pests. The bollworm sample was identified with 99.48% confidence, showing that the model

Table 1: Performance metrics of the CNN model for soybean pest detection.

Class	Precision	Recall	F1-score	Support
Aphids	1	0.9796	0.9897	49
Armyworm	1	0.975	0.9873	40
Beetle	0.9773	1	0.9885	43
Bollworm	1	0.9189	0.9577	37
Grasshopper	0.973	1	0.9863	36
Mites	0.8913	0.9762	0.9318	42
Stem borer	0.95	0.9268	0.9383	41
Accuracy			0.9688	
Macro avg	0.9702	0.9681	0.9685	288
Weighted avg	0.9703	0.9688	0.9689	288

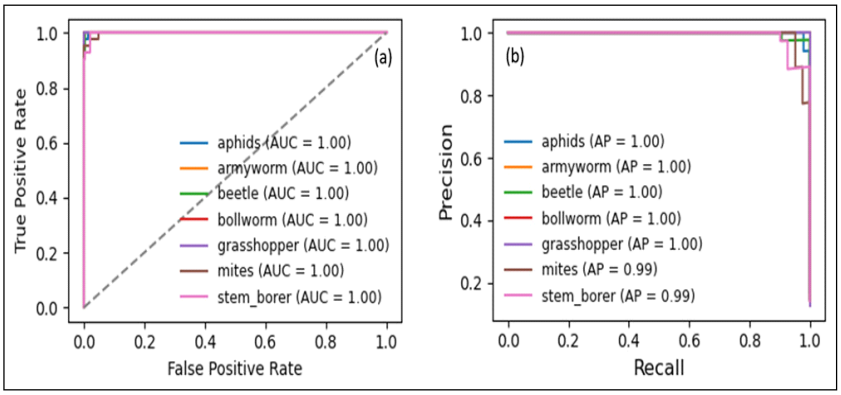


Fig 5: (a) ROC-AUC curves (b) PR curves for all classes.

performs well even for fine-grained insect classes. These results validate the CNN model's robustness across diverse pest types and confirm its practical utility in soybean pest detection scenarios. The high confidence levels across predictions indicate strong generalization on unseen data.

The custom CNN model demonstrated strong performance in classifying seven soybean pest classes, achieving an overall accuracy of 96.88% and high precision, recall and F1-scores across classes. The ROC-AUC values of 1.00 indicate excellent discrimination; however, such perfect scores suggest careful interpretation, as they may reflect the limited variability in the dataset. Future studies with larger and more diverse datasets are necessary to fully assess generalizability under field conditions.

The confusion matrix and class-wise metrics reveal that some visually similar pests, such as mites and stem borers, were occasionally misclassified, highlighting the challenges of fine-grained pest differentiation. Despite these minor misclassifications, the model showed robust performance and high confidence in predictions, suggesting strong feature learning.

Several recent studies have explored deep learning and hyperspectral techniques for soybean pest detection. Table 2 compares existing studies on soybean pest detection. Tetila *et al*. (2020) evaluated five deep learning models, Inception-v3, ResNet-50, VGG-16, VGG-19 and Xception, on UAV images, achieving a maximum accuracy of 93.82%. Gui *et al*. (2023) proposed an Attention-ResNet meta-learning model using hyperspectral data for detecting *Leguminivora glycinivorella*, reaching an accuracy of 94.57%.

Similarly, Tailanián *et al*. (2015) used SVM and spectral signatures for early caterpillar detection, reporting a 95%

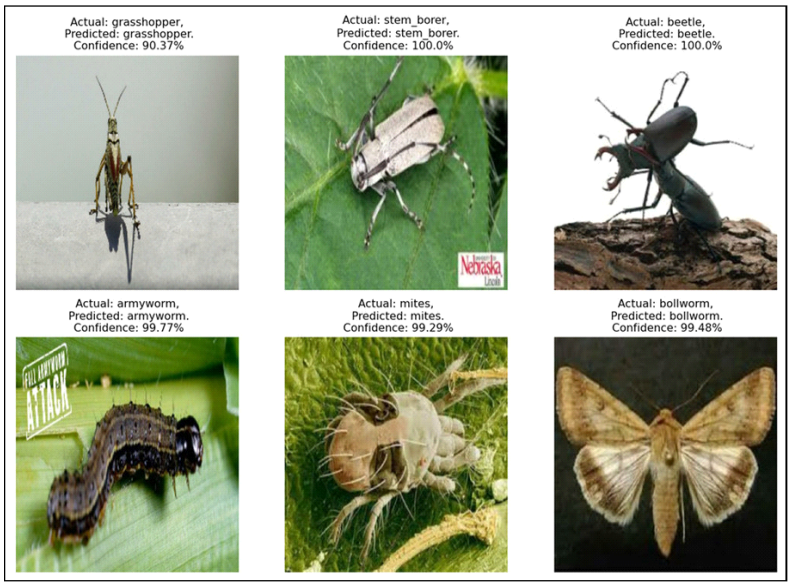


Fig 6: Sample predictions from the CNN model on test images.

Table 2: Comparative summary of soybean pest detection studies using deep learning and hyperspectral imaging approaches.

Author(s)	Method/Model	Dataset	Accuracy
Tetila *et al.* (2020)	Inception-v3, ResNet-50, VGG-16/19, Xception	5,000 UAV images in field conditions	93.82%
Gui *et al.* (2023)	A-ResNet (Attention-ResNet) + SVM	Hyperspectral images for *L. glycinivorella*	94.57%±0.19
Tailanián *et al.* (2015)	SVM with hyperspectral feature extraction	Spectral data from 4,000 ha soybean farm	95%
Ma *et al.* (2014)	FRSTCA + SVDD	362 hyperspectral soybean samples	98.80%
Huang *et al.* (2012)	Hyperspectral transmittance + SVDD	100 pods (225 beans)	95.60%
Shah *et al.* (2022)	ResNet-50 (DCNN)	Augmented image dataset (4 pest classes)	96.25%
Your Study (2025)	Custom CNN (6 Conv layers, 7 pest classes)	2,450 RGB images (labeled and preprocessed)	96.88%

classification rate. Ma *et al*. (2014) combined hyperspectral imaging with fuzzy-rough set-based wavelength selection and SVDD, achieving 98.8% accuracy on insect-damaged soybeans. Huang *et al*. (2012) applied hyperspectral transmittance imaging and SVDD, yielding 95.6% accuracy. In a more recent study, Shah *et al*. (2022) used a ResNet-50 model on an augmented image dataset, achieving 96.25% accuracy. In comparison, the present study developed a custom CNN trained on 2,450 RGB images across seven pest classes, achieving a competitive accuracy of 96.88%, demonstrating its effectiveness and scalability for real-time field deployment.

Practical deployment in Indian soybean fields presents challenges such as dense crop canopies, overlapping pest symptoms, mixed infestations and variable illumination. The model's performance suggests it can handle these conditions effectively, particularly for smallholder farmers using smartphones or edge devices. Nonetheless, potential overfitting indicated by perfect ROC-AUC values should be addressed in future work by expanding dataset diversity and incorporating UAV-assisted imagery or cross-regional data.

Overall, the proposed CNN model is highly effective, scalable and field-ready for integrated pest management in soybean farming. Its lightweight architecture enables real-time applications, supporting timely decision-making and contributing to improved crop health monitoring and yield optimization.

CONCLUSION

A deep learning-based approach was proposed in this study to identify seven key insect pests affecting soybean crops. Using a custom CNN model trained on 2,450 RGB images, the system achieved 96.88% accuracy along with strong classification matrices and ROC-AUC values. These results demonstrate the model's reliability and its potential application in precision pest management. There are some limitations. The dataset was relatively small and may not represent the full variability of real-world field conditions. In addition, the exclusive use of RGB images may reduce performance under complex lighting or environmental scenarios. Future improvements will include expanding the dataset with images from different regions and crop stages. The use of additional imaging types, such as thermal or hyperspectral, could further improve accuracy. Deploying the model on mobile or edge devices and integrating transformer-based architectures may also enhance performance in dynamic, real-time agricultural environments.

ACKNOWLEDGEMENT

Authors' contributions

All authors contributed toward data analysis, drafting and revising the paper and agreed to be responsible for all the aspects of this work.

Funding details

This research received no external funding.

Data availability

The data analysed/generated in the present study will be made available from corresponding authors upon reasonable request.

Availability of data and materials

Not applicable.

Use of artificial intelligence

Not applicable

Declarations

Authors declare that all works are original and this manuscript has not been published in any other journal.

Conflict of interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

REFERENCES

Agarwal, A., Vats, S., Agarwal, R., Ratra, A., Sharma, V. and Jain, A. (2023). EfficientNetB3 for Automated Pest Detection in Agriculture. In Proceedings of the 2023 10th International Conference on Computing for Sustainable Global Development (INDIACom). IEEE. (pp. 1408-1413).

AlZubi, A.A. (2023). Artificial Intelligence and its application in the prediction and diagnosis of animal diseases: A review. *Indian Journal of Animal Research*. **57(10):** 1265-1271. doi: 10.18805/IJAR.BF-1684.

Bagga, T., Ansari, A.H., Akhter, S., Mittal, A. and Mittal, A. (2024). Understanding indian consumers' propensity to purchase electric vehicles: An analysis of determining factors in environmentally sustainable transportation. *International Journal of Environmental Sciences*. **10(1):** 1-13.

Bhoi, M. and Sharma, P. (2025). An advanced LSTM framework for pest detection and classification in agricultural settings. *SHS Web of Conferences*. **216:** 01032. https://doi.org/10.1051/shsconf/202521601032.

Cho, O.H. (2024). An evaluation of various machine learning approaches for detecting leaf diseases in agriculture. *Legume Research*. **47(4):** 619-627. doi: 10.18805/LRF-787.

De Melo L.B.P., De Araújo, B.B.L., Hirose, E. and Borges, D.L. (2024). A lightweight and enhanced model for detecting the neotropical brown stink bug, *Euschistus heros* (Hemiptera: Pentatomidae) based on YOLOv8 for soybean fields. *Ecological Informatics*. **80:** 102543. https://doi.org/10.1016/j.ecoinf.2024.102543.

Faye, B., Azzag, H., Lebbah, M. and Feng, F. (2024). Context normalization: A new approach for the stability and improvement of neural network performance. *Data and Knowledge Engineering*. **155:** 102371. https://doi.org/10.1016/j.datak.2024.102371.

FAOSTAT. (2023). Soybeans: Yields by Country (tonnes per hectare), 2023. Food and Agriculture Organization of the United Nations. https://www.fao.org/faostat/en/#data/QC.

Gaur, N. and Mogalapu, S. (2018). Pests of Soybean. In Springer eBooks (pp. 137-162). https://doi.org/10.1007/978-981-10-8687-8_6.

Gui, J., Xu, H. and Fei, J. (2023). Non-destructive detection of soybean pest based on hyperspectral image and attention-resnet meta-learning model. *Sensors*. **23(2):** 678. https://doi.org/10.3390/s23020678.

Hai, N.T. and Duong, N.T. (2024). An improved environmental management model for assuring energy and economic prosperity. *Acta Innovations*. **52:** 9-18. https://doi.org/10.62441/ActaInnovations.52.2.

Heinrichs, E.A. and Muniappan, R. (2019). Integrated pest management for tropical crops: Soyabeans. *CABI Reviews*. p 1-44. https://doi.org/10.1079/pavsnnr201813055.

Huang, M., Wan, X., Zhang, M. and Zhu, Q. (2012). Detection of insect-damaged vegetable soybeans using hyperspectral transmittance image. *Journal of Food Engineering*. **116(1):** 45-49. https://doi.org/10.1016/j.jfoodeng.2012.11.014.

ISWS. (2024). Addressing yield gaps in Indian soybean cultivation through integrated weed and pest management. *Indian Journal of Weed Science*. **56(4):** 417-425.

Kasinathan, T. and Uyyala, S.R. (2021). Machine learning ensemble with image processing for pest identification and classification in field crops. *Neural Computing and Applications*. **33(13):** 7491-7504. https://doi.org/10.1007/s00521-020-05497-z.

Ma, Y., Huang, M., Yang, B. and Zhu, Q. (2014). Automatic threshold method and optimal wavelength selection for insect-damaged vegetable soybean detection using hyperspectral images. *Computers and Electronics in Agriculture*. **106:** 102-110. https://doi.org/10.1016/j.compag.2014.05.014.

Maltare, N.N., Sharma, D. and Patel, S. (2023). An exploration and prediction of rainfall and groundwater level for the district of Banaskantha, Gujarat, India. *International Journal of Environmental Sciences*. **9(1):** 1-17.

Nargund, R., Bhatia, V.S., Sinha, N.K., Mohanty, M., Jayaraman, S., Dang, Y.P., Nataraj, V., Drewry, D. and Dalal, R.C. (2024). Assessing soybean yield potential and yield gap in different agroecological regions of India using the DSSAT model. *Agronomy*. **14(9):** 1929. https://doi.org/10.3390/agronomy14091929.

Natukunda, M.I. and MacIntosh, G.C. (2020). The resistant soybean-aphis glycines interaction: Current knowledge and prospects. *Frontiers in Plant Science*. **11.** https://doi.org/10.3389/fpls.2020.01223.

Park, Y., Choi, S. H., Kwon, Y., Kwon, S., Kang, Y.J. and Jun, T. (2023). Detection of soybean insect pest and a forecasting platform using deep learning with unmanned ground vehicles. *Agronomy*. **13(2):** 477. https://doi.org/10.3390/agronomy13020477.

Şahin, Y.S., Gençer, N.S. and Şahin, H. (2025). Integrating AI detection and language models for real-time pest management in tomato cultivation. *Frontiers in Plant Science*. **15.** https://doi.org/10.3389/fpls.2024.1468676.

Shah, D., Gupta, R., Patel, K., Jariwala, D. and Kanani, J. (2022). Deep Learning Based Pest Classification in Soybean Crop using Residual Network 50. In: Proceedings of the 2022 IEEE 2nd International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC). IEEE. (pp.1-5). https://doi.org/10.1109/iSSSC56467.2022.10051424.

Shinde, S. and Attar, V. (2024). MH-SoyaHealthVision: An Indian UAV and leaf image dataset for integrated crop health assessment (Version 1) [Data set]. Mendeley Data. https://doi.org/10.17632/hkbgh5s3b7.1.

Suárez-Paniagua, V. and Segura-Bedmar, I. (2018). Evaluation of pooling operations in convolutional architectures for drug-drug interaction extraction. *BMC Bioinformatics*. **19(S8).** https://doi.org/10.1186/s12859-018-2195-1.

Tailanián, M., Castiglioni, E., Musé, P., Flores, G.F., Lema, G., Mastrángelo, P., Almansa, M., Liñares, I.F. and Liñares, G.F. (2015). Early pest detection in soy plantations from hyperspectral measurements: A case study for caterpillar detection. *Proceedings of SPIE, the International Society for Optical Engineering/Proceedings of SPIE*. **9637:** 96372I. https://doi.org/10.1117/12.2195083.

Tang, Z., Lu, J., Chen, Z., Qi, F. and Zhang, L. (2023). Improved pest-YOLO: Real-time pest detection based on efficient channel attention mechanism and transformer encoder. *Ecological Informatics*. **78:** 102340. https://doi.org/10.1016/j.ecoinf.2023.102340.

Tetila, E.C., Machado, B.B., Astolfi, G., De Souza, B.N.A., Amorim, W.P., Roel, A.R. and Pistori, H. (2020). Detection and classification of soybean pests using deep learning with UAV images. *Computers and Electronics in Agriculture*. **179:** 105836. https://doi.org/10.1016/j.compag.2020.105836.

Teuwen, J. and Moriakov, N. (2019). Convolutional Neural Networks. In Elsevier eBooks (pp. 481–501). https://doi.org/10.1016/b978-0-12-816176-0.00025-9.

Venkatasaichandrakanth, P. and Iyapparaja, M. (2024). A survey on pest detection and classification in field crops using artificial intelligence techniques. *International Journal of Intelligent Robotics and Applications*. **8(3):** 709-734. https://doi.org/10.1007/s41315-024-00347-w.

Xiang, Q., Huang, X., Huang, Z., Chen, X., Cheng, J. and Tang, X. (2023). Yolo-pest: An insect pest object detection algorithm *via* CAC3 module. *Sensors*. **23(6):** 3221. https://doi.org/10.3390/s23063221.

Zhang, Y. and Lv, C. (2024). TinySegformer: A lightweight visual segmentation model for real-time agricultural pest detection. *Computers and Electronics in Agriculture*. **218:** 108740. https://doi.org/10.1016/j.compag.2024.108740.