eSystems Engineering Society EST. 2007

International Journal of Data Science and Advanced Analytics

Journal homepage: www.ijdsaa.com

Insect Pest Classification Using Vision Transformers

Remen Khullar[1], Luke K. Topham[1] and Channa Basava Chola[2]
[1]Liverpool John Moores University, Liverpool, UK [2]University of Mysore, India

[1]remenkhullar@gmail.com, l.k.topham@LJMU.ac.uk, [2]Channabasavac7@compsci.uni-mysore.ac.in

Abstract- Transformer-based neural networks have emerged as the dominant approach for natural language processing tasks due to their strong performance in text understanding and generation. Motivated by this success, recent research has explored the application of transformer architectures to computer vision, an area traditionally dominated by Convolutional Neural Networks (CNNs). Although the adoption of transformers in vision tasks is relatively recent, early studies have demonstrated promising results, particularly in image classification. However, limited research has systematically evaluated Vision Transformer models for fine-grained insect pest classification using large-scale agricultural datasets. In this paper, a Vision Transformer (ViT)-based computer vision model is proposed for the automated identification and classification of insect pests that damage agricultural crops. Insect pests destroy nearly one-third of global agricultural production and pose significant risks to food security and public health. Automating pest identification using deep learning can significantly reduce the time and effort required compared to manual inspection. The proposed model is trained and fine-tuned on the IP102 dataset, which contains more than 75,000 images spanning 102 categories of common crop-damaging insect pests. Experimental results demonstrate that the 8×8 ViT achieved 41.75% test accuracy, 67.28% top 5 accuracy, and 85.56% ROC-AUC, compared with 47.61% accuracy, 73.76% top 5 accuracy, and 91.21% ROC-AUC achieved by the best-performing CNN, InceptionV3. These results demonstrate the potential of transformer-based architectures for fine-grained insect pest classification in the IP102 benchmark.

Keywords- Transformer networks; vision transformers; image classification; insect pest classification

1. Introduction

Transformer architectures, introduced by [1], have become the dominant modelling paradigm for natural language processing due to their ability to capture long-range dependencies using self-attention mechanisms. Compared to earlier sequence modelling approaches, transformers offer improved parallelisation and computational efficiency, enabling training on extremely large datasets with billions of parameters. This success has motivated an increasing interest in extending transformer-based architectures to computer vision tasks. Computer vision has historically been dominated by Convolutional Neural Networks (CNNs), which exploit inductive biases such as locality and translation invariance. Architectures such as ResNet continue to serve as the foundation for many state-of-the-art vision systems, as demonstrated by [2, 3, 4]. Although CNNs remain highly effective, their architectural constraints have prompted investigation into alternative designs that

https://doi.org/10.69511/ijdsaa.v7i2.337
Received 28 July 2025; Received in revised form 03 September 2025; Accepted 15 October 2025.
Available online 23 October 2025.

may offer improved scalability and representational capacity.

The application of transformers to vision tasks is relatively recent. Vision Transformers [5] replace convolutional operations with self-attention applied to sequences of image patches. While these models are theoretically more scalable than CNNs, they lack strong inductive biases and therefore require substantially larger training datasets to generalise effectively. When trained on sufficiently large datasets, however, vision transformers have been shown to match or surpass the performance of state-of-the-art CNNs.

In parallel with these architectural developments, insect pests remain a major threat to global agricultural productivity. They are responsible for the destruction of nearly one-third of global crop yields and contribute to the spread of plant-related diseases, with significant implications for food security and agricultural economics, as reported by [6] and [7]. Accurate identification of insect pests typically requires expert knowledge and extensive manual effort, making large-scale monitoring costly and time-consuming. Automated pest classification using computer vision models offers a practical solution to address these challenges.

This paper investigates the use of a transformer-based computer vision model for the identification and classification of insect pests. The proposed approach is based on the Vision Transformer (ViT) architecture [5] is trained and fine-tuned on the IP102 dataset [8], which contains over 75,000 images across 102 insect pest categories. The performance of the transformer-based model is evaluated and compared against widely used CNN architectures, including VGG16, Xception, InceptionV3, and EfficientNetV2B3, trained under identical conditions. The results provide insight into the suitability of transformer-based models for fine-grained insect pest classification in agricultural applications.

2. Related Work

Transformer architectures, originally proposed by Vaswani et al. [1], have recently been adapted to computer vision tasks following their success in natural language processing. Dosovitskiy et al. introduced the ViT [5], demonstrating that a pure transformer-based architecture can achieve competitive or superior performance compared to CNNs when trained on sufficiently large datasets. Their work showed that vision transformers are scalable and computationally efficient in theory, particularly when leveraging large-scale pretraining followed by fine-tuning on mid-sized datasets. Despite these advances, CNNs remain the dominant approach for most computer vision applications. Large-scale architectures such as ResNet continue to serve as the backbone of many state-of-the-art models, as reported by [2, 3, 4]. Well-established CNN architectures such as VGG16, Inception, Xception, and EfficientNet have demonstrated strong performance across a wide range of image classification tasks. VGG16, introduced by Simonyan et al. [9], employs a deep stack of small convolutional filters and remains a common baseline. The Inception family of architectures, proposed by Szegedy et al. [10, 11, 12], addresses computational efficiency through multi-scale feature extraction, while Xception, introduced by Chollet [13], further improves efficiency through depthwise separable convolutions. EfficientNet [14] and EfficientNetV2 [15], respectively, focus on compound scaling to balance network depth, width, and resolution for improved performance and efficiency.

In the domain of insect pest classification, Wu et al. [8] introduced the IP102 dataset, which contains over 75,000 images across 102 insect pest categories. The dataset was constructed with expert annotation and reflects real-world challenges, including class imbalance. Several studies have utilised this dataset to evaluate deep learning models for pest identification. Agarwal et al. [16] conducted a comparative study using multiple pretrained CNN models on IP102 and reported that VGG16 achieved the highest classification accuracy among the evaluated architectures.

Recent studies have also explored the application of vision transformers to fine-grained image classification tasks outside agriculture. Pan [17] demonstrated the effectiveness of vision transformers for penguin species classification, achieving high accuracy on a balanced dataset. Similarly, Li and Tanone [18] applied vision transformers to vegetable classification and reported performance comparable to, or slightly better than, that of CNNs when sufficient training data was available. These studies highlight the potential of vision transformers for visual classification tasks, while also noting their dependence on dataset size and computational resources.

Berroukham et al. [19] provided a comprehensive review of vision transformers, emphasising their ability to capture long-range dependencies through self-attention mechanisms. While vision transformers offer advantages in scalability and interpretability, their reliance on large, labelled datasets remains key limitation compared to CNNs with stronger inductive biases.

Overall, existing literature demonstrates the effectiveness of CNNs for insect pest classification and the growing potential of vision transformers for fine-grained visual recognition. Although transformer-based approaches have previously been applied to pest and agricultural image classification, limited work has systematically investigated the effect of ViT patch size on insect pest classification and compared different ViT configurations with established CNN architectures on the IP102 benchmark under a consistent training-from-scratch setting. This study addresses this gap by evaluating four ViT configurations with different patch sizes and comparing the best-performing configuration against VGG16, Xception, InceptionV3, and EfficientNetV2-B3 using consistent experimental conditions.

3. Materials and Methods

This study evaluates transformer-based and CNN architectures for insect pest classification using the IP102 dataset. A set of ViT models with varying patch sizes is compared against widely used CNNs under consistent training and evaluation settings.

3.1. Dataset and Data Partitioning

Experiments were conducted using the IP102 dataset introduced by Wu et al [8], which contains 75,222 images spanning 102 insect pest categories. A sample preview of the dataset is shown in Fig. 1. The dataset exhibits significant class imbalance, as illustrated in Figure 2, with class sizes ranging from 71 to 5,740 images.

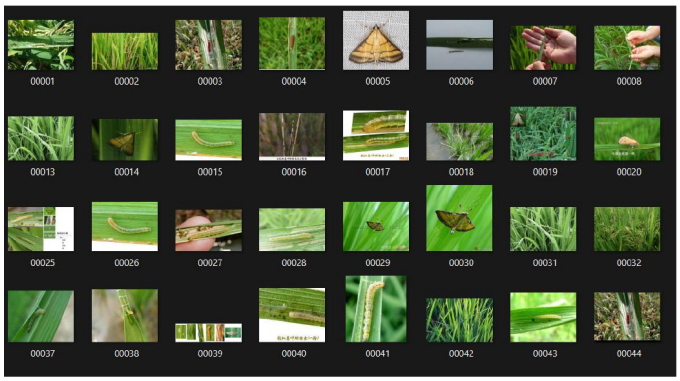

Fig. 1. Preview of the IP102 Dataset [8] showcasing the variety of classes.

The dataset was provided with predefined training, validation, and test splits to maintain consistent class distributions. The partitions comprise 45,095 training images, 7,508 validation images, and 22,619 test images. Images were organised into directory structures compatible with TensorFlow's image data loading utilities. Labels were encoded using one-hot representations to avoid ordinal bias in classification.

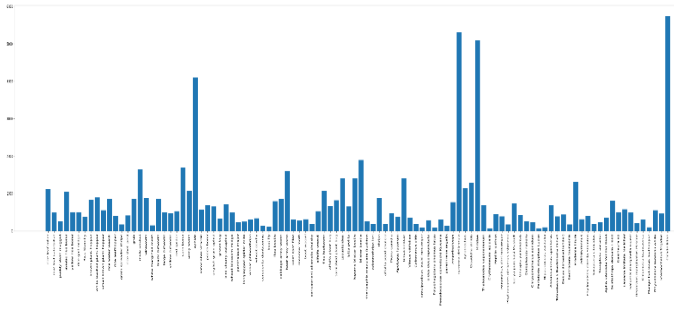

Figure 2. Bar Plot Demonstrating the Imbalance of the Classes [8].

The present study addresses image-level classification, where each input image is assigned to one of the 102 pest categories. It does not address object localisation, instance detection, or estimating the number of individual pests within an image. Consequently, the experiments do not assess how the classifier would identify and differentiate multiple pest instances in an arbitrary field image outside the IP102 test distribution

3.2. Data Preprocessing and Augmentation

All images were processed in RGB format. To ensure fair comparison across architectures, identical augmentation strategies were applied to all models, except for input resizing and normalisation, which were model-specific. Common augmentation techniques included random horizontal flipping, random rotation, and random zooming, implemented as a data augmentation block within the model pipeline. Model-specific preprocessing steps were as follows:

- ViT input images were resized to 72×72 pixels and normalised to the range [0, 1].
- VGG16 input images were resized to 224×224 pixels and normalised to [0, 1].
- Inception V3 and Xception input images were resized to 299×299 pixels and normalised to the range [−1, 1].
- EfficientNetV2-B3 input images were resized to 300×300 pixels, with normalisation handled internally by the model.

3.3. Model Architectures

ViT models were implemented from scratch, following the architecture proposed [5], using TensorFlow and Keras. Each model consisted of an image patch extraction layer, a patch embedding and positional encoding layer, eight transformer encoder blocks with multi-head self-attention and residual connections, and fully connected layers for classification.

Image-to-patch conversion is illustrated in Fig. 3. Four ViT variants were evaluated, differing only in patch size while maintaining identical depth:

- 4×4 patches, resulting in 324 patches per image
- 6×6 patches, resulting in 144 patches per image
- 8×8 patches, resulting in 81 patches per image
- 12×12 patches, resulting in 36 patches per image

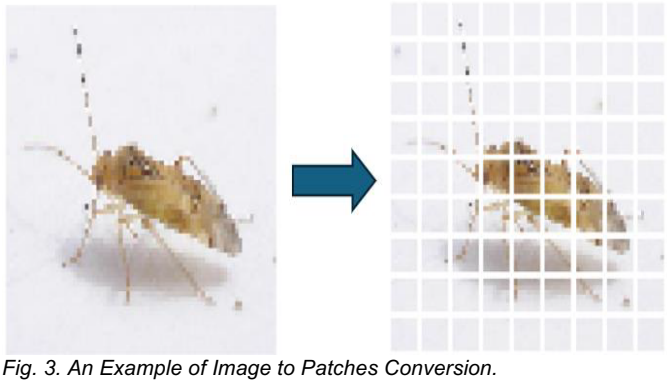

Fig. 3. An Example of Image to Patches Conversion.

The total number of parameters varied substantially across these models due to changes in patch count, as summarised in **Table *1***.

Table 1. Summary of the Number of Parameters for each ViT Model.

Patch Size	Total Parameters	Trainable Parameters	Non-Trainable Parameters
4x4	45,361,709	45,361,702	7
6x6	21,761,069	21,761,062	7
8x8	13,504,877	13,504,870	7
12x12	7,619,117	7,619,110	7

All ViT models were trained without pretrained weights to ensure a fair comparison with convolutional models. Training employed the AdamW optimiser with a learning rate of 0.001 and a weight decay of 0.0001, using categorical cross-entropy as the loss function. Batch sizes were set to 64 for the 12×12 and 8×8 models, and to 32 for the 4×4 and 6×6 models, due to memory constraints. Training was conducted for up to 100 epochs with early stopping based on validation loss.

The training histories in Figures 4 to 7 illustrate the convergence behaviour of the four ViT configurations, with noticeable differences in loss and classification performance across patch sizes. The training histories show clear differences in convergence behaviour across the ViT configurations. The 4×4 patch model shows relatively high training loss and limited improvement in classification accuracy, suggesting difficulty learning an effective representation despite its substantially larger parameter count. In contrast, the 6×6 and 8×8 configurations converge more consistently, with the 8×8 model achieving the strongest training accuracy and top-5 accuracy. The 12×12 model also converges effectively but reaches lower performance than the 8×8 configuration. These trends are consistent with the final evaluation results and suggest that an intermediate patch size provides a more effective balance between spatial detail and sequence length for this dataset.

All CNN models were trained using the Adamax optimiser with a learning rate of 0.001 and categorical cross-entropy loss. Batch sizes were set to 32 for VGG16 and 16 for the remaining models due to memory limitations. Each model was trained for up to 100 epochs with early stopping applied. Training histories for these models are shown in Fig. 4-7.

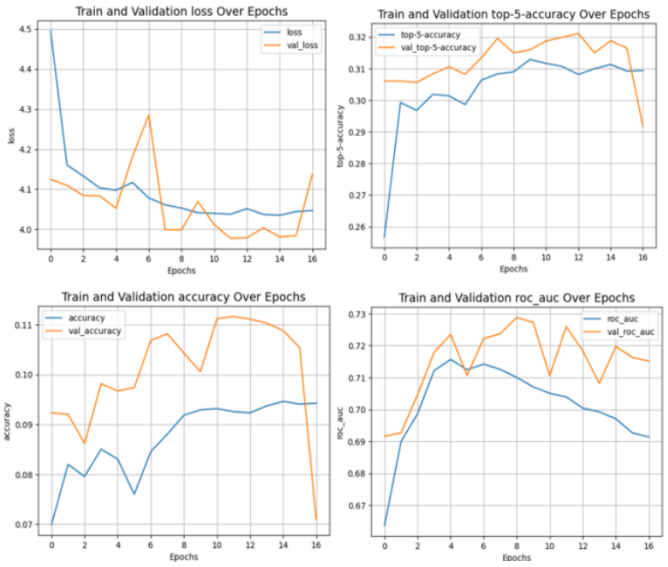

Fig. 4. Training History for 4x4 Showing Loss, Top 5 Accuracy, Accuracy, and ROC AUC.

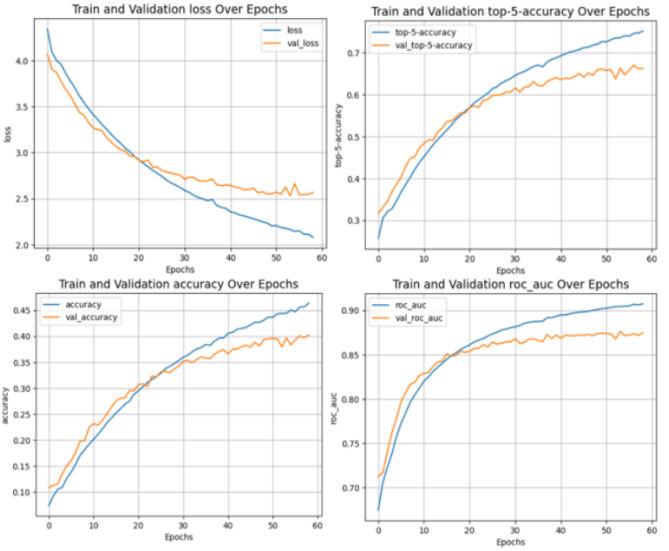

Fig. 5. Training History for 6x6 Showing Loss, Top 5 Accuracy, Accuracy, and ROC AUC.

In addition to the ViR models, four CNN architectures were evaluated for comparison: VGG16, Inception V3, Xception, and EfficientNetV2-B3. These models were instantiated using the Keras Applications module without pretrained ImageNet weights. Model parameter statistics are reported in **Table *2***.

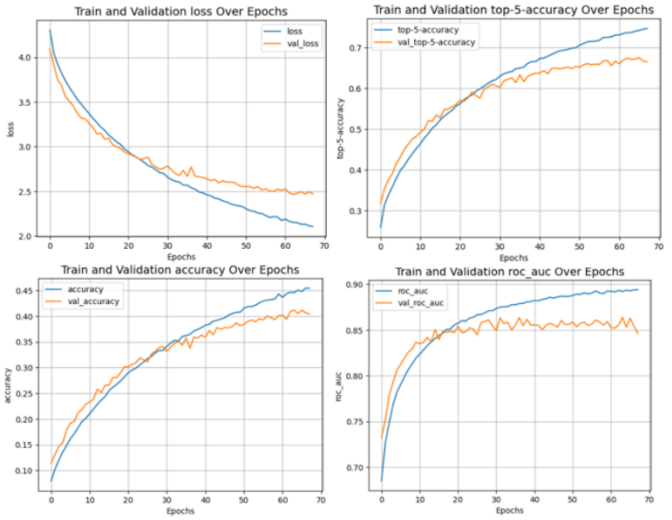

Fig. 6. Training History for 8x8 Showing Loss, Top 5 Accuracy, Accuracy and ROC AUC.

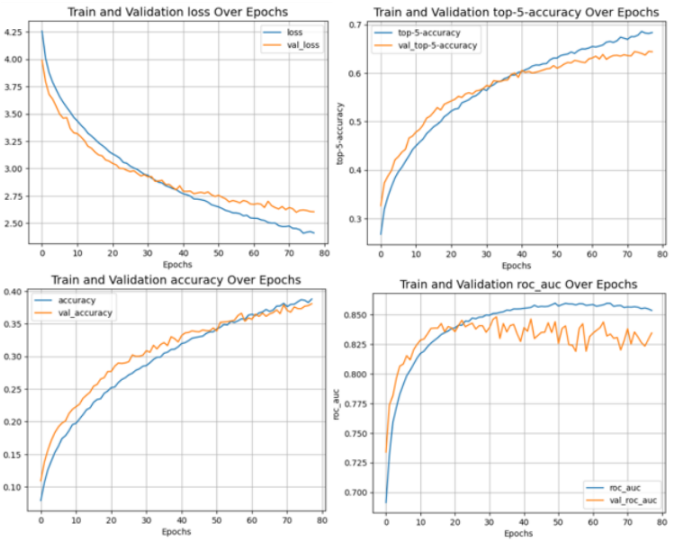

Fig. 7. Training History for 12x12 Showing Loss, Top 5 Accuracy, Accuracy, and ROC AUC.

Table 2. Summary of Number of Parameters for Different CNN Models.

Model Type	Total Parameters	Trainable Parameters	Non-Trainable Parameters
VGG16	134,678,438	134,678,438	0
Inception V3	22,011,782	22,011,782	22,011,782
Xception	21,070,478	21,070,478	21,070,478
EfficientNet V2 B3	13,087,396	13,087,396	13,087,396

3.4. Evaluation Metrics

Model performance was evaluated using accuracy, top-5 accuracy, and receiver operating characteristic area under the curve (ROC-AUC). These metrics were computed on both the training and testing datasets during inference to assess classification performance and generalisation behaviour. Accuracy measures the proportion of correctly classified samples, while top-5 accuracy evaluates whether the true class appears among the five highest-confidence predictions, which is particularly relevant for large-scale multi-class problems. ROC-AUC quantifies a model's ability to distinguish between classes across varying decision thresholds and provides insight into its robustness to class imbalance.

In addition to these metrics, categorical cross-entropy loss was monitored during training and validation to evaluate convergence behaviour. Model performance is interpreted based on the joint objective of minimising loss while maximising evaluation metrics. Higher top-5 accuracy and ROC-AUC, and lower loss values, indicate superior model performance.

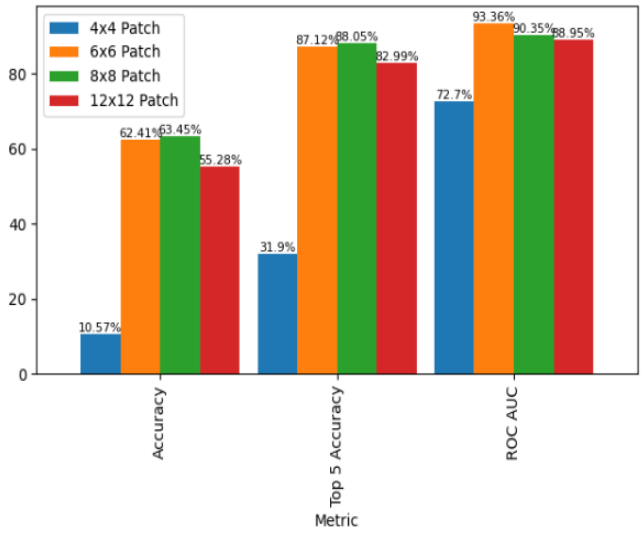

Fig. 8. Training Accuracy, Top 5 Accuracy, and ROC AUC for ViT models.

Summary evaluation results for the vision transformer models on the training dataset are reported in **Table *3*** and visualised in Fig. 8, with corresponding testing results discussed in the subsequent Results and Discussion section.

Table 3. Train Dataset Evaluation Summary of Vision Transformers across patch sizes 4x4, 6x6, 8x8, and 12x12.

Metric	4x4	6x6	8x8	12x12
Loss	3.97	1.41	1.41	1.71
Accuracy	10.57	62.41	63.45	55.28
Top 5 Accuracy	31.90	87.12	88.05	82.99
ROC AUC	72.70	93.36	90.35	88.95

3.5. Implementation Environment

All experiments were implemented in Python using TensorFlow and Keras. Data handling and preprocessing were performed using NumPy and Pandas, while Matplotlib and Seaborn were used for visualisation. Training was accelerated using NVIDIA CUDA and cuDNN libraries to leverage GPU computation.

4. Results and Discussion

The performance of the proposed ViT models and baseline CNN architectures was evaluated on the IP102 dataset using the metrics defined in Section III-D. All models were trained from scratch without pretrained weights to ensure a fair comparison between ViT and CNN architectures under identical resource constraints. Although the absence of pretraining may limit absolute performance, this setting enables meaningful relative performance analysis.

4.1. Vision Transformer Performance

Among the evaluated ViT configurations, the model with an 8×8 patch size consistently achieved the best performance across both training and testing datasets. It exhibited the lowest loss values and the highest top-5 accuracy, ROC-AUC, and overall accuracy among the ViT variants, as summarised in **Table *3*** and **Table *4*** and visualised in Fig. 8 and Fig. 9.

Table 4. Test Dataset Evaluation Summary of ViT models across patch sizes 4x4, 6x6, 8x8, and 12x12.

Metric	4x4	6x6	8x8	12x12
Loss	3.98	2.50	2.46	2.60
Accuracy	10.92	40.47	41.75	38.26
Top 5 Accuracy	31.72	66.78	67.28	64.59
ROC AUC	72.44	87.57	85.56	83.81

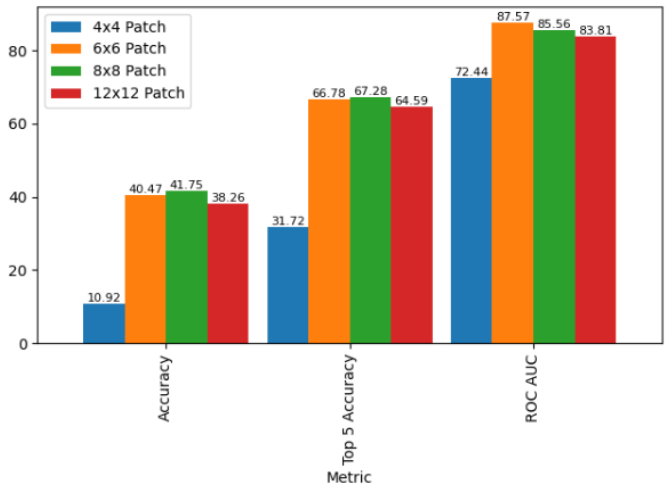

Fig. 9. Testing Accuracy, Top 5 Accuracy and ROC AUC for ViT models.

The 6×6 patch configuration produced competitive results but was consistently inferior to the 8×8 configuration, suggesting an imbalance between spatial resolution and sequence length. The 12×12 patch configuration yielded moderate performance, indicating that an insufficient number of patches limited the model's ability to capture fine-grained visual features. In contrast, the 4×4 patch configuration performed poorly, with high loss and low evaluation metrics, indicating that excessively small patches resulted in overly long token sequences that hindered effective learning. Based on these results, the 8×8 patch size was selected as the representative ViT model for comparison with CNN architectures.

4.2. Convolutional Neural Network Performance

The performance of the CNN models on the training and testing datasets is summarised **Table *5*** and **Table *6*** and illustrated in Fig. 10 and Fig. 11. Among the CNN architectures, Inception V3 achieved the strongest overall performance, attaining the lowest loss values and highest evaluation metrics on both datasets. EfficientNetV2-B3 followed closely, delivering comparable performance despite having nearly half the number of parameters.

Table 5. Train Dataset Evaluation Summary of CNN Models.

Metric	GG16	Xception	InceptionV3	EfficientNet V2B3
Loss	2.36	2.58	1.15	1.27
Accuracy	41.29	36.26	67.40	64.93
Top 5 Accuracy	69.30	64.84	90.72	88.33
ROC AUC	93.18	91.49	98.04	98.00

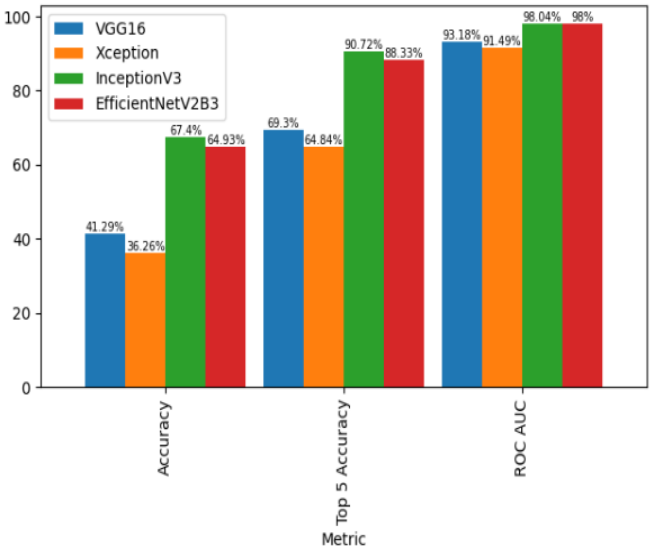

Fig. 10. Training Accuracy, Top 5 Accuracy and ROC AUC for CNN Models.

VGG16 exhibited the weakest performance among the CNN models, with higher loss values and lower accuracy on both training and testing sets, indicating limited representational capacity for this task. Xception achieved moderate results but did not outperform Inception V3 or EfficientNetV2-B3 under the same training conditions.

Table 6. Test Dataset Evaluation Summary of CNN Models.

Metric	GG16	Xception	InceptionV3	EfficientNet V2B3
Loss	3.17	2.88	2.31	2.33
Accuracy	29.48	31.69	47.61	44.61
Top 5 Accuracy	54.03	59.05	73.76	71.02
ROC AUC	86.35	88.94	91.21	91.76

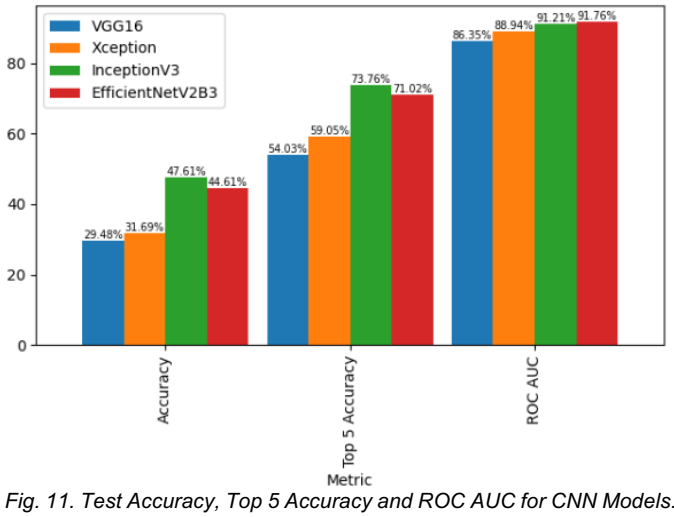

Fig. 11. Test Accuracy, Top 5 Accuracy and ROC AUC for CNN Models.

4.3. Comparative Analysis of ViT and CNN Models

The ViT model with an 8×8 patch configuration demonstrated performance comparable to that of Inception V3 and EfficientNetV2-B3 on the IP102 dataset. Notably, the ViT achieved this performance with a parameter count similar to that of EfficientNetV2-B3 while maintaining a significantly shallower architecture, comprising only 84 layers, compared to the much deeper CNN models.

This observation highlights the training efficiency of ViT architectures, as competitive classification performance was achieved without relying on deep hierarchical feature extraction. The results suggest that self-attention mechanisms enable ViTs to model global relationships effectively, even in fine-grained, multi-class agricultural image classification tasks. However, ViT's sensitivity to patch size underscores the importance of architectural configuration when applying transformers to vision tasks. Overall, these findings indicate that ViTs can serve as viable alternatives to CNNs for insect pest classification, particularly when model depth and parameter efficiency are key considerations. While CNNs such as Inception V3 remain strong performers, ViTs show promising potential for scalable, efficient computer vision applications in agriculture.

5. Conclusions

This paper investigated the effectiveness of ViT architectures for insect pest classification on the IP102 dataset and compared their performance against established CNN models. Experimental results demonstrate that ViT performance is highly sensitive to patch size, which directly affects model capacity and learning behaviour. Among the evaluated configurations, an 8×8 patch size provided the best balance between sequence length and feature representation, yielding superior performance compared to smaller or larger patch sizes.

The ViT model achieved classification performance comparable to high-performing CNN architectures such as Inception V3 and EfficientNetV2-B3, despite operating on significantly smaller input images and using a shallower architecture. This reduced input resolution enabled larger batch sizes and faster training under identical hardware constraints. Moreover, comparable performance was achieved with fewer layers and a parameter count similar to the smallest CNN evaluated, highlighting the training efficiency and representational capability of self-attention-based architectures.

Overall, the results indicate that ViTs can serve as competitive alternatives to CNNs for fine-grained agricultural image classification tasks. Overall, the results demonstrate the potential of ViT architectures for insect pest classification within the IP102 benchmark. However, these findings should be interpreted as a controlled benchmark evaluation rather than evidence of deployment readiness. Practical deployment would require further evaluation using field-collected images, including variations in illumination, background, image quality, camera devices, and previously unseen pest populations.

The primary limitation of this study is the absence of large-scale pretraining. Neither the ViT nor the CNN models were initialised with pretrained weights due to computational and time constraints. Because ViTs benefit substantially from pretraining on large datasets, this limitation likely constrained their peak performance on IP102. Additionally, the IP102 dataset, while large for an agricultural benchmark, may still be insufficient for fully exploiting the representational capacity of ViT architectures.

A further limitation is that IP102 is a curated benchmark dataset and therefore does not fully represent the variability encountered in real-world agricultural environments. Consequently, the reported results should not be interpreted as demonstrating immediate field deployment capability. Future work should evaluate the models on independently collected field data and investigate robustness to environmental and imaging variations before considering practical deployment.

Future work will focus on pretraining ViT models on large-scale image datasets before fine-tuning on IP102, as well as on exploring deeper transformer architectures and higher input resolutions. Further investigation into hybrid architectures and data-efficient training strategies may also improve performance and robustness in real-world agricultural applications.

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems,* 2017.

[2] M. Dhruv, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe and L. Van Der Maaten, "Exploring the limits of weakly supervised pretraining," in *Proceedings of the European conference on computer vision (ECCV)*, 2018.

[3] X. Qizhe, M.-T. Luong, E. Hovy and Q. V. Le, "Self-training with noisy student improves imagenet classification," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[4] A. Kolesnikov, L. Beyer, X. Zhai, J. Puigcerver, J. Yung, S. Gelly and N. Houlsby, "Big transfer (bit): General visual representation learning," in *European conference on computer vision*, 2020.

[5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations (ICLR)*, 2021.

[6] D. Ozdemir and M. S. Kunduraci, "Comparison of deep learning techniques for classification of the insects in order level with mobile software application," *IEEE Access,* vol. 10, pp. 35675-35684, 2022.

[7] F. Faithpraise, P. Birch, R. Young, J. Obu, B. Faithpraise and C. Chatwin, "Automatic plant pest detection and recognition using k-means clustering algorithm and correspondence filters," *International Journal of Advanced Biotechnology and Research,* vol. 4, no. 2, pp. 189-199, 2013.

[8] X. Wu, C. Zhan, Y.-K. Lai, M.-M. Cheng and J. Yang, "Ip102: A large-scale benchmark dataset for insect pest recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019.

[9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

[10] C. Szegedy, S. Ioffe, V. Vanhoucke and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the AAAI conference on artificial intelligence*, 2017.

[11] C. Szegedy, W. Liu, J. Yangqing, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015.

[12] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[13] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[14] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*, 2019.

[15] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *International conference on machine learning*, 2021.

[16] N. Agarwal, T. Kalita and A. K. Dubey, "Classification of insect pest species using cnn based models," in *International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, 2023.

[17] P. Luting, "Vision transformer and its application in penguin classification," in *International Conference on Image Processing, Computer Vision and Machine Learning (ICICML)*, 2022.

[18] L.-H. T. R. Li, "Vision transformer approach for vegetables recognition," in *International seminar on application for technology of information and communication (iSemantic)*, 2022.

[19] A. Berroukham, K. Housni and M. Lahraichi, "Vision transformers: a review of architecture, applications, and future directions," in *7th IEEE congress on information science and Technology (CiSt)*, 2023.