International Journal of Innovative Research in Computer Science and Technology (IJIRCST)
ISSN (Online): 2347-5552, Volume-14, Issue-5, Sep-Oct 2026
DOI: https://doi.org/10.55524/ijircst.2026.14.5.5
Article ID IRP-1789, Pages 40-47
www.ijircst.org

Genetic Algorithm-Optimized Vision Transformer for Automated Tomato Leaf Disease Classification

Gunjan Mishra[1], Harsh Dev[2], and Sudheer Kumar Singh[3]

[1]Research Scholar, Department of Computer Science & Engineering, Babu Banarasi Das University, Lucknow, Uttar Pradesh, India
[2]Professor, Department of Computer Science & Engineering, Babu Banarasi Das University, Lucknow, Uttar Pradesh, India
[3] Department of Computer Science & Engineering, Galgotias University, Greater Noida, Uttar Pradesh, India

Correspondence should be addressed to Gunjan Mishra; gunjan.mishracs@gmail.com

Received: 2 Sep 2026; **1st Revision:** 16 Sep 2026; **2nd Revision:** 26 Sep 2026; **Accepted:** 2 Oct 2026;
Published Online: 9 Oct 2026

Copyright © 2026 Gunjan Mishra et al. This is an open-access article distributed under the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ABSTRACT- Automated detection of diseases in tomato leaves will assist in the timely evaluation of the crop health. However, the performance of Vision Transformer (ViT) is highly dependent on the hyperparameters. The presented research concerns a Vision Transformer optimized by a Genetic Algorithm (GA-ViT) for the binary classification of Early Blight and healthy tomato foliage using 2,591 images divided according to a stratified 70:15:15 training-validation-testing framework. The images were resized to 128 x 128 pixels and divided into 256 individual non-overlapping 8 x 8 pieces. A Genetic Algorithm was used to tune the learning rate, dropout rate, feed-forward dimension, number of attention heads and number of transformer blocks with validation accuracy used as the fitness function. For each seed, GA optimization and final model training were performed independently. The test accuracies were 97.94%, 95.63%, and 94.86% for seeds 42, 52, and 62, respectively. The mean test accuracy was 96.14%, with a standard deviation of 1.61 percentage points. The mean weighted precision was 96.18%, weighted recall was 96.14%, and weighted F1-score was 96.13%. The mean test loss was 0.1276. The results indicate the potential of evolutionary hyperparameter optimization to improve ViT-based plant disease classification.

KEYWORDS- Deep Learning, Genetic Algorithm, Image Classification, Vision Transformer, Tomato Leaf Disease Classification

I. INTRODUCTION

Crop quality and yield can be reduced by plant diseases. Early and dependable diagnosis is an essential part of crop-health management [1]. The reason why image-based deep learning has been extensively studied is that a trained model can directly learn discriminative visual representations from leaf images. Previous work on large plant-leaf datasets has shown that deep neural networks [2] are viable for automated disease identification, but performance may drop when models are tested on images that are collected under different conditions than the training images.

Convolutional neural networks [3] have been the dominant approach for plant image classification, but Vision Transformers (ViTs) [4] use a different representation by leveraging image patches and self-attention. The original ViT showed that one can encode images as a sequence of patches and process them with transformer encoders [5]. Recently, tomato-disease researches have studied ViT variants and transformer-based systems, demonstrating the suitability of such approaches for plant disease recognition [6].

ViT performance is influenced by hyperparameters including learning rate, dropout, feed-forward dimension, attention-head configuration, and network depth. Researchers may need to choose these parameters by hand, usually through trial and error. Genetic Algorithms (GAs) are a kind of population-based search methods where candidate solutions are evaluated by a fitness function and evolved by means of selection, crossover and mutation. The application of GAs for neural-network design and documented in reviews of evolutionary neural-network optimization [7].

Therefore, the aim of this work is to develop and experimentally evaluate a true GA-ViT pipeline for the classification of tomato leaf diseases. The proposed implementation is not a GA that runs independently of model training. Each chromosome is decoded into a ViT configuration, the corresponding candidate model is trained on the training set, validation accuracy is measured, and the validation accuracy is returned as the GA fitness. The final chosen configuration is then trained and evaluated once on an held-out test set set.

A. Contributions

- A reproducible binary classification structure for analyzing tomato leaves with a 70:15:15 stratified split for training, validation, and testing.
- A Vision Transformer working on 128x128-pixel RGB images categorized into 256 non-overlapping (8x8) segments.
- An optimization loop from GA to ViT is established,

where each candidate chromosome represents the actual hyperparameters for ViT and is evaluated based on the validation accuracy, which serves as its fitness.

- A comparative analysis of a manually configured baseline ViT tested independently and a GA-optimized ViT.
- Extensive documentation of classification metrics, confusion matrices, training trajectories, and genetic algorithm search patterns.

II. RELATED WORK

A. Deep Learning for Plant Disease Recognition

Deep learning [8] has become an important approach for automated plant disease recognition. It can learn discriminative features directly from leaf images. Various deep learning architectures have been investigated for plant disease identification and classification. These models often achieve strong performance on controlled and benchmark datasets. They also reduce the need for manually designed features and support end-to-end classification. However, their performance can be affected by image quality, illumination, background conditions, disease symptoms, and dataset characteristics. Therefore, robust evaluation using diverse datasets is essential. Validation under real-world field conditions is also important for developing reliable plant disease recognition systems.

B. Vision Transformers in Agricultural Imaging

ViT uses a patch-based representation and self-attention mechanism instead of a traditional convolutional image-based framework. In agricultural imaging, transformer models [9] have been explored for tomato disease recognition and related plant health activities. For instance, ViT-SmartAgri evaluated ViT-based tomato disease detection in a mobile-focused setting [10]. Other recent studies have investigated transformer-based transfer learning and improved transformer architectures for tomato leaf disease classification[11]. These studies suggest that attention-centric models are pertinent to the assessment of agricultural images, but the composition of datasets, evaluation methodologies and model configurations have a substantial impact on the reported outcomes.

C. Evolutionary Optimization of Neural Networks

Genetic Algorithms can investigate design options that are either discrete or continuous, by means of evolutionary mechanisms. The current study uses the Distributed Evolutionary Algorithms in Python (DEAP) framework providing clear data structures and evolutionary operators for fast prototyping. Hyperparameter optimization [12] is a well-known challenge in Deep Learning [13], and systematic reviews describe metaheuristic and other approaches to reduce reliance on manual tuning.

D. Research Gap

The specific experimental void addressed here is not simply to use a Transformer or a genetic algorithm in isolation. It represents the seamless combination of GA candidate encoding, ViT architecture generation, candidate training, fitness evaluation with validation, evolution, and independent final evaluation in a single reproducible setup. This research investigates whether this fusion may improve the efficiency of a static baseline ViT on a binary tomato leaf dataset.

III. MATERIALS AND METHODS

A. Dataset

For this research work, the tomato leaf images is selected from the PlantVillage dataset Mohanty et al. [14] to validate the GA-optimized ViT hybrid model. The implementation given here is fetching the tomato leaf images from two local folders, Tomato_Early_blight and Tomato_healthy. The dataset consists of 2,591 tomato leaf images of which 1,000 are early blight images and 1,591 are healthy leaves. Therefore, the classes are imbalanced, where healthy leaves make up about 61.4% of the dataset. The code provided points to a local directory rather than a dataset citation, so the document does not assume a public dataset identity apart from the class labels. Table 1 shows the number and approximate proportion of images belonging to the Tomato Early Blight and Tomato Healthy classes in the complete dataset of 2,591 images.

Table 1: Class Distribution of the Tomato Leaf Image Dataset

Class	Images	Approx. proportion
Tomato_Early_blight	1,000	38.6%
Tomato_healthy	1,591	61.4%
Total	2,591	100%

B. Preprocessing and Data Partitioning

The images are processed using OpenCV [15], converting them from BGR to RGB format and resizing them to 128×128 pixels. The stratification division is performed in two stages, namely training for 70% of the total dataset, validation and held-out test seting for the remaining 30% of the dataset, which is equally divided. The resulting split consists of 1,813 training images, 389 validation images and 389 held-out test set images. The stratified split guarantees that the ratios of classes are preserved in all three splits. Table 2 summarizes the stratified 70:15:15 dataset partition, consisting of 1,813 training, 389 validation, and 389 held-out test set images.

Table 2: Dataset Distribution Across Training, Validation, and Test Sets

Subset	Images	Percentage	Role
Training	1,813	69.97%	Parameter learning
Validation	389	15.01%	GA fitness, early stopping, LR scheduling
Held-out test set	389	15.01%	Final performance evaluation

C. Data Augmentation

The training-time augmentation is implemented as a Keras Sequential layer that applies random horizontal flipping, random rotation with angle 0.15, random zoom with amount 0.15, and random translation with height and width factor 0.10. This augmentation layer is part of the model graph, so it is active during training but inactive during inference/evaluation.

D. Vision Transformer Architecture

The input tensor is of size 128 × 128 × 3. The patch size of 8×8 yields 16×16=256 patches. Each raw patch is composed of 8×8×3=192 values. The patch representation is obtained by a convolution operation with 192 filters, 8×8 kernel and stride of 8, which is then reshaped into a sequence of 256 patch tokens. A compact layer maps tokens into a 192-dim embedding space. This research work uses learnable positional embeddings to keep the order of the patches.

Each Transformer block consists of layer normalization, multi-head self-attention, residual connection, another layer normalization, feed-forward network with Gaussian Error Linear Unit (GELU) activation [16], dropout, a projection back to 192 dimensions, and a residual connection. After the last block, global average pooling over the token dimension produces a 192-dimensional representation, followed by dropout and a softmax classification layer with two units. Fig. 1 illustrates the overall GA-ViT workflow, including image preprocessing, stratified data partitioning, ViT patch generation, Transformer encoding, GA-based hyperparameter optimization, and held-out test seting.

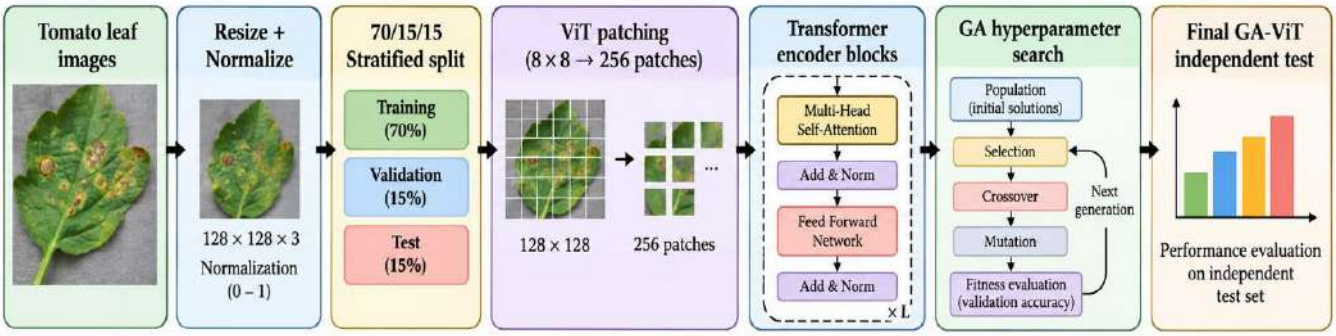

Figure 1: Experimental GA-ViT processing and optimization pipeline

E. Baseline ViT Configuration

Table 3 summarizes the architectural and training parameters used to configure and train the baseline Vision Transformer model [17].

Table 3: Baseline Vision Transformer Configuration and Training Parameters

Parameter	Baseline value
Learning rate	0.001
Dropout	0.10
Feed-forward dimension	128
Attention heads	4
Transformer blocks	8
Batch size	32
Maximum epochs	15
Optimizer	Adam
Loss	Sparse categorical cross-entropy
Early stopping	Patience 4; restore best weights
LR reduction	Factor 0.5; patience 2; min LR 1e-6

F. Genetic Algorithm Design

The chromosome is composed of five integer indices that correspond to the five hyperparameters of the ViT: learning rate, dropout rate, feed-forward dimension, number of attention heads and number of transformer blocks.

Table 4 presents the search space explored by the Genetic Algorithm [18] for optimizing the learning rate, dropout, FFN dimension, attention heads, and Transformer blocks.

The GA search space contains 900 possible configurations (5 × 5 × 4 × 3 × 3). The GA uses a population of four individuals and runs for three generations. A crossover probability of 0.7 and a mutation probability of 0.3 are applied. Gene-wise crossover and tournament selection are used during optimization. The complete GA procedure is executed independently for random seeds 42, 52, and 62.

Table 4: Hyperparameter Search Space for Genetic Algorithm-Based ViT Optimization

Hyperparameter	Search values
Learning rate	0.00001, 0.00005, 0.0001, 0.0005, 0.001
Dropout	0.1, 0.2, 0.3, 0.4, 0.5
FFN dimension	64, 128, 256, 512
Attention heads	2, 4, 8
Transformer blocks	4, 6, 8

For each chromosome, a new ViT model is created using the decoded hyperparameters. Each model is trained according to the defined candidate-evaluation budget. Validation accuracy is used as the GA fitness value. Previously evaluated duplicate chromosomes can reuse their computed fitness values. Therefore, the number of newly evaluated candidates may vary across generations. The held-out test set set is not used during GA optimization or hyperparameter selection.

G. Final GA-ViT Training and Held-out test seting

After three GA generations within each seed-specific run, the best chromosome for that run is decoded and used to create a new final ViT. The final model is trained for 10 epochs with batch size 32 using the training subset and validation monitoring. The corresponding held-out test set subset is evaluated only after model training. This complete GA-search–training–test sequence is repeated separately for seeds 42, 52, and 62.

IV. EXPERIMENTAL RESULTS

A. Multi-Seed Experimental Protocol

Three independent GA-ViT experiments were conducted using random seeds 42, 52, and 62. Each experiment used a stratified 70:15:15 train-validation-test split. The split contained 1,813 training images, 389 validation images, and 389 held-out test set images. GA optimization and final model training were performed separately for each seed. The held-out test set set was used only for final evaluation.

The test accuracies were 97.94%, 95.63%, and 94.86% for seeds 42, 52, and 62, respectively. Across the three runs, the mean test accuracy was 96.14%, with a standard deviation (SD) of 1.61 percentage points. The mean weighted precision was 96.18%, weighted recall was 96.14%, and weighted F1-score was 96.13%.

The mean test loss was 0.1276. The complete seed-wise metrics and their summary statistics are reported in Table 5.

Table 5: Held-out test set Performance Across Three GA-ViT Runs

Seed	Accuracy (%)	Weighted Precision (%)	Weighted Recall (%)	Weighted F1 (%)	Test Loss
42	97.94	97.95	97.94	97.94	0.0778
52	95.63	95.73	95.63	95.60	0.1664
62	94.86	94.85	94.86	94.85	0.1387
Mean	96.14	96.18	96.14	96.13	0.1276
SD	1.61	1.60	1.61	1.61	0.0453

Fig. 2 shows the accuracy, weighted precision, weighted recall, and weighted F1-score obtained for seeds 42, 52, and 62. The four metrics follow a similar pattern across the three independent runs.

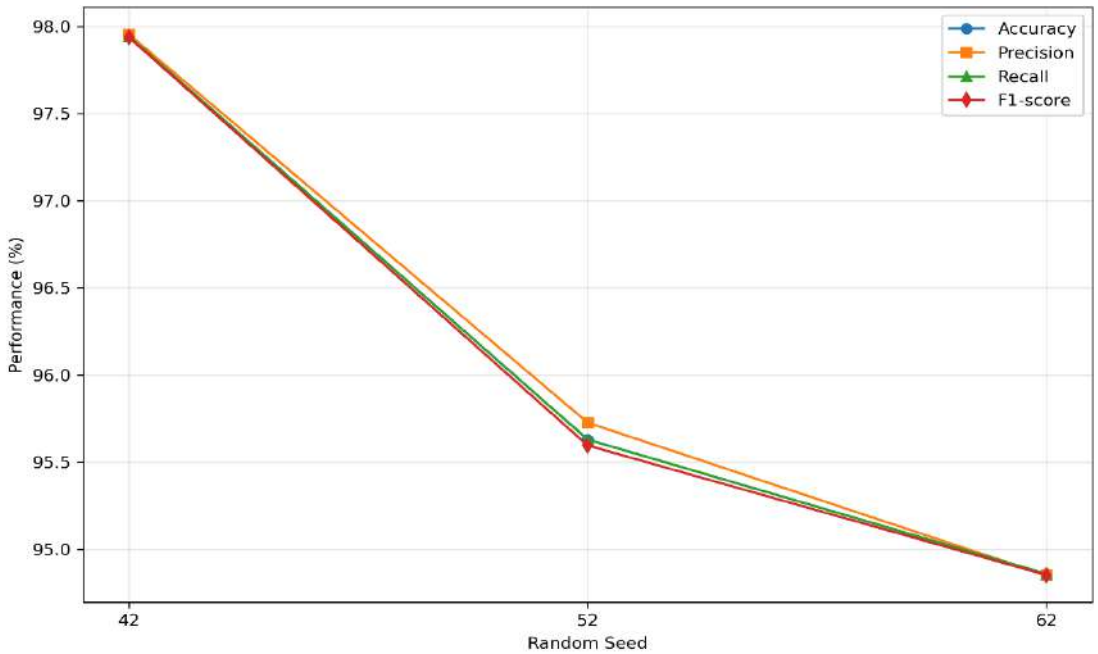

Figure 2: GA-ViT Performance Across Three Independent Random Seeds

B. GA Search Results

The GA search space contains 900 possible configurations (5 × 5 × 4 × 3 × 3). The GA uses a population of four individuals and runs for three generations. The crossover probability is 0.7 and the mutation probability is 0.3. Gene-wise crossover and tournament selection are used during optimization. The complete GA procedure is executed independently for seeds 42, 52, and 62.

For each chromosome, a new ViT model is created using the decoded hyperparameters. Each candidate is trained under the defined evaluation budget. Validation accuracy is used as the fitness value. Duplicate chromosomes may reuse previously computed fitness values. Thus, the number of newly evaluated candidates can vary between generations. Held-out test set images are not used for GA fitness or hyperparameter selection. The best hyperparameter configuration identified for each seed is reported in Table 6.

Table 6: Best GA Hyperparameters for Each Random Seed

Seed	Learning Rate	Dropout	FFN Dimension	Heads	Blocks	Best Validation Accuracy
42	0.00005	0.1	64	8	4	93.57%
52	0.00005	0.1	512	4	8	91.52%
62	0.00005	0.1	256	8	4	88.69%

C. Final GA-ViT Results

Held-out test set accuracy was 97.94%, 95.63%, and 94.86% for seeds 42, 52, and 62, respectively. The corresponding weighted precision values were 97.95%, 95.73%, and 94.85%, while weighted recall values were 97.94%, 95.63%, and 94.86%. The weighted F1-scores were 97.94%, 95.60%, and 94.85%, respectively. Across the three independent runs, the GA-ViT model achieved a mean accuracy of 96.14%, weighted precision of 96.18%, weighted recall of 96.14%, and weighted F1-score of 96.13%. The mean test loss was 0.1276.

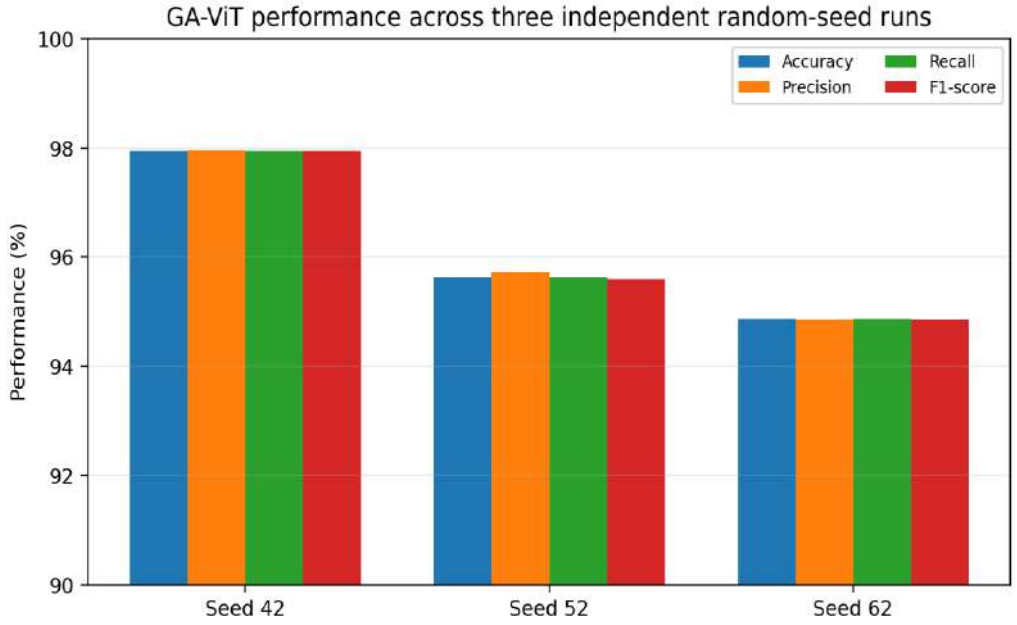

Figure 3. Independent GA-ViT performance across different random seeds.

The standard deviations were 1.61, 1.60, 1.61, and 1.61 percentage points for accuracy, weighted precision, weighted recall, and weighted F1-score, respectively, while the standard deviation of test loss was 0.0453. Fig. 3 presents the accuracy, weighted precision, weighted recall, and weighted F1-score of the GA-ViT model across three independent random-seed runs (42, 52, and 62).

D. Class-Wise Test Performance

Class-wise results were recorded separately for each seed. For seed 42, 144 of 150 Early Blight images and 237 of 239 healthy images were classified correctly. For seed 52, the corresponding values were 136 and 236. For seed 62, they were 139 and 230. These results show variation in class-level predictions across independent runs.

Table 7 presents the class-wise performance of the GA-ViT model across three independent random seeds. Both classes achieved high precision, recall, and F1-scores in all runs. The results show some performance variation across the three random seeds.

Table 7: Class-Wise Performance Across the Three Runs

Seed	Class	Precision	Recall	F1-score	Support
42	Tomato_Early_blight	0.9863	0.9600	0.9730	150
42	Tomato_healthy	0.9753	0.9916	0.9834	239
52	Tomato_Early_blight	0.9784	0.9067	0.9412	150
52	Tomato_healthy	0.9440	0.9874	0.9652	239
62	Tomato_Early_blight	0.9392	0.9267	0.9329	150
62	Tomato_healthy	0.9544	0.9623	0.9583	239

E. Training Behavior

The final GA-ViT model was trained for 10 epochs in each independent run. The final validation accuracies were 96.40%, 96.14%, and 92.80% for seeds 42, 52, and 62, respectively. The variation across runs supports reporting the mean and standard deviation rather than relying on one run.

F. Reproducibility and Experimental Protocol

The principal settings used in the three-seed experiments are summarized in Table 8.

Table 8: Reproducibility Configuration

Setting	Value
Random seeds	42, 52, 62
Input size	128 × 128 × 3
Classes	2
Split	70% train / 15% validation / 15% test
Train / validation / test	1,813 / 389 / 389
GA population	4
GA generations	3
Crossover probability	0.7
Mutation probability	0.3
GA fitness	Validation accuracy
Final training epochs	10
Held-out test set use	Final evaluation only

V. DISCUSSION

A. Effect of Evolutionary Hyperparameter Search

The GA selected different ViT configurations across the three seeds. All runs selected a learning rate of 0.00005 and a dropout rate of 0.1. The FFN dimension varied from 64 to 512. The number of attention heads and Transformer blocks also varied. This variation reflects the stochastic nature of the optimization and training process.

Validation accuracy was used as the GA fitness measure. The best validation fitness values were 93.57%, 91.52%, and 88.69% for seeds 42, 52, and 62. These values describe the configurations selected within the defined GA search budget. They do not establish global optimality.

B. Held-out test set Evaluation

Each held-out test set set contained 389 images. The test images were excluded from GA fitness evaluation and model selection. The three runs achieved test accuracies of 97.94%, 95.63%, and 94.86%. The mean test accuracy was 96.14%, with a standard deviation of 1.61 percentage points. The mean weighted precision, recall, and F1-score were 96.18%, 96.14%, and 96.13%, respectively.

C. Comparison with Existing Literature

Transformer-based methods have reported promising results for tomato leaf disease classification. Direct numerical comparison remains difficult when datasets, class definitions, data partitions, preprocessing, and evaluation protocols differ. For example, ViT-SmartAgri reported 90.99% test accuracy in a broader multiclass setting. A 2024 comparison of ViT variants reported 97% accuracy for CompactViT. Transformer transfer learning [19], evolutionary architecture optimization [20], and tomato disease classification [21] have also been investigated. The present results are limited to the two-class dataset and evaluation protocol used in this study.

D. Computational Considerations

The experiments were executed on CPU because no GPU was detected. The GA was run independently for all three seeds. The recorded runtimes were 44.36, 83.65, and 63.85 minutes for seeds 42, 52, and 62. The total experimental time was approximately 3.20 hours. The optimization therefore adds computational cost to model selection.

VI. CONCLUSION

This study presents a GA-ViT model to perform binary classification between Tomato Early Blight and healthy leaves. The study used 2,591 images from the PlantVillage dataset, which contains 1,000 images of Early Blight disease leaves and 1,591 images of healthy leaves. The model was tested in three independent runs with random seeds 42, 52, 62. The accuracy of the test was 97.94%, 95.63%, and 94.86% respectively. The proposed model

achieved a mean accuracy of 96.14%, weighted precision of 96.18%, weighted recall of 96.14%, weighted F1-score of 96.13%, and a mean test loss of 0.1276. The results show that ViT-based classification of tomato leaves can effectively be improved through hyperparameter optimization using Genetic Algorithms. The performance variations between the three trials emphasize the importance of evaluating the model with different random seeds. The present study is restricted to two selected classes from a single source, PlantVillage dataset, and three experiments with random seeds. Future work will be adding more random seeds, multiclass disease classification, field collected images, and independent external datasets.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

ACKNOWLEDGMENT

The authors thank their supervisor for invaluable guidance and support throughout this research. We also thank the reviewers for their constructive comments and suggestions, which helped improve the quality of this paper.

REFERENCES

[1] J. Dhanushya, M. Ruba, M. Ponkarthika, B. Soundarya, V. Sakthi Priya, S. Dhanushri, S. Gomathi, and S. Harikrishnan, "AI-powered multipurpose agrobot for precision farming: Crop health management, fertilizer recommendation and yield prediction," in *Proc. 2025 Int. Conf. Recent Adv. Electr., Electron., Ubiquitous Commun., Comput. Intell. (RAEEUCCI)*, 2025, pp. 1–6. Available from: https://doi.org/10.1109/RAEEUCCI63961.2025.11048251

[2] M. J. Rasch, F. Carta, O. Fagbohungbe, and T. Gokmen, "Fast and robust analog in-memory deep neural network training," *Nature Commun.*, vol. 15, Art. no. 7133, 2024. Available from: https://doi.org/10.1038/s41467-024-51221-z

[3] A. Bala Ayyappan, T. Gobinath, M. Kumar, and A. Sivaramakrishnan, "Rice plant disease detection using convolutional neural networks," *Discover Artif. Intell.*, vol. 5, Art. no. 50, 2025. Available from: https://doi.org/10.1007/s44163-025-00277-x

[4] Y. Wang *et al.*, "Vision transformers for image classification: A comparative survey," *Technologies*, vol. 13, no. 1, Art. no. 32, 2025. Available from: https://doi.org/10.3390/technologies13010032

[5] D. Álvarez-Fidalgo and F. Ortin, "CLAVE: A deep learning model for source code authorship verification with contrastive learning and transformer encoders," *Inf. Process. Manage.*, vol. 62, no. 3, Art. no. 104005, 2025. Available from: https://doi.org/10.1016/j.ipm.2024.104005

[6] X. Yue, K. Qi, X. Na, Y. Liu, F. Yang, and W. Wang, "Deep learning for recognition and detection of plant diseases and pests," *Neural Comput. Appl.*, vol. 37, no. 17, pp. 11265–11310, 2025. Available from: https://doi.org/10.1007/s00521-025-11125-5

[7] H. T. Ünal and F. Başçiftçi, "Evolutionary design of neural network architectures: A review of three decades of research," *Artif. Intell. Rev.*, vol. 55, no. 3, pp. 1723–1802, 2022. Available from: https://doi.org/10.1007/s10462-021-10049-5

[8] S. Suthaharan, "Deep learning models," in *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*. Boston, MA, USA: Springer, 2016, pp. 289–307. Available from: https://doi.org/10.1007/978-1-4899-7641-3_12

[9] P. S. Thakur, S. Chaturvedi, P. Khanna, T. Sheorey, and A. Ojha, "Vision transformer meets convolutional neural network for plant disease classification," *Ecol. Inform.*, vol. 77, Art. no. 102245, 2023. Available from: https://doi.org/10.1016/j.ecoinf.2023.102245

[10] D. Bondar *et al.*, "Mobile-focused spatial inspection of industrial parts using 2D image processing and LiDAR," *Measurement*, vol. 266, Art. no. 120522, 2026. Available from: https://doi.org/10.1016/j.measurement.2026.120522

[11] M. Moussafir, H. Chaibi, R. Saadane, A. Chehri, A. El Rharras, and G. Jeon, "Design of efficient techniques for tomato leaf disease detection using genetic algorithm-based and deep neural networks," *Plant Soil*, vol. 479, no. 1, pp. 251–266, 2022. Available from: https://doi.org/10.1007/s11104-022-05513-2

[12] L. Franceschi, M. Donini, V. Perrone, A. Klein, C. Archambeau, M. Seeger, M. Pontil, and P. Frasconi, "Hyperparameter optimization in machine learning," *Found. Trends Mach. Learn.*, vol. 18, no. 6, pp. 975–1109, 2025. Available from: https://doi.org/10.1561/2200000088

[13] G. Mishra, H. Dev, and S. K. Singh, "Evolutionary algorithms and deep learning techniques for plant disease identification: A comprehensive review," in *Proc. Int. Conf. Semiconductor Technologies—Materials to Chips*, Singapore, 2024. Available from: https://doi.org/10.1007/978-981-95-3305-3_27

[14] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Front. Plant Sci.*, vol. 7, Art. no. 1419, 2016. Available from: https://doi.org/10.3389/fpls.2016.01419

[15] M. Rashid, B. Ram, R. S. Batth, N. Ahmad, H. M. E. I. Dafallaa, and M. B. Rehman, "Novel image processing technique for feature detection of wheat crops using Python OpenCV," in *Proc. 2019 Int. Conf. Comput. Intell. Knowl. Econ. (ICCIKE)*, 2019, pp. 559–563. Available from: https://doi.org/10.1109/ICCIKE47802.2019.9004432

[16] D. Feng, X. Zhuang, Z. Chen, S. Zhong, Y. Qi, H. Chen, and H. Ma, "Position information encoding FPN for small object detection in aerial images," *Neural Comput. Appl.*, vol. 36, no. 26, pp. 16023–16035, 2024. Available from: https://doi.org/10.1007/s00521-024-09917-2

[17] F. Gerz, M. G. Schneider, and M. Jelali, "Integration of vision transformer networks in YOLOv8 for object detection: Comparative study on plant disease detection," *IEEE Access*, 2026. Available from: https://doi.org/10.1109/ACCESS.2026.3665969

[18] D. Waysi, B. T. Ahmed, and I. M. Ibrahim, "Optimization by nature: A review of genetic algorithm techniques," *Indones. J. Comput. Sci.*, vol. 14, no. 1, 2025. Available from: https://doi.org/10.33022/ijcs.v14i1.4596

[19] P. Garg, K. Vora, D. Shah, M. V. Joshi, A. Kumar, and V. S. Palaparthy, "Transformer-based framework using in-house sensors and lightweight images for accurate plant disease classification," *IEEE Sensors J.*, 2026. Available from: https://doi.org/10.1109/JSEN.2026.3656207

[20] N. Gupta, M. Khosravy, N. Patel, and I. Sethi, "Evolutionary optimization based on biological evolution in plants," *Procedia Comput. Sci.*, vol. 126, pp. 146–155, 2018. Available from: https://doi.org/10.1016/j.procs.2018.07.218

[21] X. Wang and J. Liu, "TomatoGuard-YOLO: A novel efficient tomato disease detection method," *Front. Plant Sci.*, vol. 15, Art. no. 1499278, 2025. Available from: https://doi.org/10.3389/fpls.2024.1499278

ABOUT THE AUTHORS

Gunjan Mishra graduated from Dr. A.P.J. Abdul Kalam Technical University, Lucknow. He earned his M.Tech. in Computer Science and Engineering from Babu Banarasi Das University, Lucknow. He is currently pursuing Ph.D. in Computer Science and Engineering from Babu Banarasi Das University, Lucknow. His research interests include Data Structure, Artificial Intelligence and Machine Learning.

Harsh Dev earned his Ph.D. in Computer Science from Babasaheb Bhimrao Ambedkar University (Central University), Lucknow. He is currently serving as the Professor and Head of the Department of Computer Science and Engineering at Babu Banarasi Das University, Lucknow. His areas of specialization include Data Mining, Artificial Intelligence, Machine Learning, Deep Learning, and Data Analytics.

Sudheer Kumar Singh earned his Ph.D. in Computer Science and Engineering from Dr. A.P.J. Abdul Kalam Technical University, Lucknow. He is currently serving as an Associate Professor in the Department of Computer Science and Engineering at Galgotias University, Greater Noida. His areas of specialization include Software Engineering, Requirement Engineering, Data Structure and Machine Learning.