

Comparative Evaluation of Convolutional and Transformer-Based Models for Cucumber Disease Classification

Maftun Aliyev^{1*} , Javanshir Zeynalov¹ , Yigitcan Cakmak² ,
Ishak Pacal² , and Sabina Mahmudova¹ 

Abstract. *Cucumber (*Cucumis sativus*) is an important horticultural crop, and disease-related losses can reduce both yield and market quality. Image-based deep learning methods have been increasingly used for plant disease recognition, yet practical model selection requires attention to both predictive performance and computational cost. In this study, four pretrained deep learning architectures were evaluated for eight-class cucumber disease classification: DenseNet201, Inception v4, ConvNeXt v2 Base, and DeiT Base. The experiments were conducted on a public dataset containing 7,689 images, including two healthy and six diseased categories, using a 70% training, 15% validation, and 15% test split. DeiT Base achieved the highest test accuracy among the evaluated models, with an accuracy of 0.9957. Inception v4 followed closely with an accuracy of 0.9948, whereas DenseNet201 provided a lower but still competitive accuracy of 0.9846 with the smallest computational burden, requiring 18.11M parameters and 8.58 GFLOPs. The results indicate that highly accurate cucumber disease classification can be achieved with both convolutional and Transformer-based models, but the preferred architecture depends on whether the priority is maximum accuracy or lower computational cost. Since the experiments were limited to a single public dataset, further validation on field-acquired images is needed before drawing conclusions about real-world deployment.*

Keywords: *cucumber disease classification, plant disease recognition, deep learning, transfer learning*

¹Nakhchivan State University, Nakhchivan, Azerbaijan

²Iğdir University, Iğdir, Türkiye

^{2,3}Iğdir University, Iğdir, Türkiye

Fenerbahçe University, Istanbul, Türkiye

*Corresponding author. E-mail: meftunaliyev@ndu.edu.az

Received: 4 May 2026; Accepted: 3 September 2026; Published online: 22 October 2026

© The Author(s) 2026. This is an open access article distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Xiyar xəstəliklərinin təsnifatı üçün konvolyusiya və transformator əsaslı modellərin müqayisəli qiymətləndirilməsi

Məftun Əliyev^{1*} , Cavanşir Zeynalov¹ , Yığitcan Çakmak² ,
İshak Paçal² , Səbinə Mahmudova¹ 

Xülasə. *Xiyar (*Cucumis sativus*) mühüm bağçılıq bitkisidir və xəstəliklərlə bağlı itkilər həm məhsuldarlığı, həm də bazar keyfiyyətini azalda bilər. Təsvirlərə əsaslanan dərin öyrənmə üsulları bitki xəstəliklərinin tanınmasında getdikcə daha geniş istifadə olunur. Bununla belə, praktik model seçimi yalnız proqnozlaşdırma dəqiqliyini deyil, həm də hesablama xərclərini nəzərə almağı tələb edir.*

Bu tədqiqatda səkkiz sinifli xiyar xəstəliyi təsnifatı üçün dörd əvvəlcədən öyrədilmiş dərin öyrənmə arxitekturası qiymətləndirilmişdir: DenseNet201, Inception v4, ConvNeXt v2 Base və DeiT Base. Eksperimentlər iki sağlam və altı xəstə kateqoriyanı əhatə edən 7,689 təsvirlik açıq məlumat dəsti üzərində aparılmış və məlumatlar 70% təlim, 15% validasiya və 15% test hissələrinə bölünmüşdür. Qiymətləndirilən modellər arasında DeiT Base ən yüksək test dəqiqliyinə nail olmuşdur və onun dəqiqlik göstəricisi 0.9957 təşkil etmişdir. Inception v4 0.9948 dəqiqliklə ona çox yaxın nəticə göstərmişdir. DenseNet201 isə nisbətən aşağı, lakin yenə də rəqabətə davamlı 0.9846 dəqiqlik əldə etmiş və ən aşağı hesablama yükünə malik olmuşdur; bu model 18.11 milyon parametr və 8.58 GFLOPs tələb etmişdir. Nəticələr göstərir ki, xiyar xəstəliklərinin yüksək dəqiqlikli təsnifatı həm konvolyusional, həm də Transformer əsaslı modellərlə mümkündür. Bununla belə, üstünlük veriləcək arxitektura maksimum dəqiqliyin, yoxsa daha aşağı hesablama xərclərinin prioritet olmasından asılıdır. Eksperimentlər yalnız bir açıq məlumat dəsti ilə məhdudlaşdığı üçün real sahə şəraitində tətbiq haqqında daha güclü nəticələr çıxarmaqdan əvvəl sahədən toplanmış təsvirlər üzərində əlavə validasiya aparılmalıdır.

Açar sözlər: *xiyar xəstəliklərinin təsnifatı, bitki xəstəliklərinin tanınması, dərin öyrənmə, transfer öyrənməsi*

¹Naxçıvan Dövlət Universiteti, Naxçıvan, Azərbaycan

²İğdır Universiteti, İğdır, Türkiyə

^{2,3}İğdır Universiteti, İğdır, Türkiyə

Fəxrəbaxça Universiteti, İstanbul, Türkiyə

*Məsul müəllif. E-poçt: meftunaliyev@ndu.edu.az

Daxil oldu: 4 May 2026; Qəbul edildi: 3 Sentyabr 2026; Onlayn dərc edildi: 22 Oktyabr 2026

© Müəllif(lər) 2026. Bu açıq girişli məqalə Creative Commons Attribution–NonCommercial 4.0 Beynəlxalq Lisenziyasının (CC BY-NC 4.0) şərtlərinə uyğun olaraq yayımlanmışdır.

Introduction

Cucumber (*Cucumis sativus*) is widely cultivated for fresh consumption and food processing, and it represents an economically important horticultural crop in many agricultural regions (Y. Wang et al., 2024; Zeng et al., 2025). However, cucumber production is affected by several fungal, bacterial, and viral diseases that can reduce yield, fruit quality, and market value (Asha Rani & Gowrishankar, 2024; Cao et al., 2025). Diseases such as Downy Mildew, Anthracnose, Bacterial Wilt, Gummy Stem Blight, and Cucumber Mosaic Virus often produce visible symptoms on leaves, stems, or fruits, including discoloration, necrotic lesions, wilting, and tissue deformation. If these symptoms are not detected at an early stage, disease spread may increase production losses and lead to unnecessary or poorly timed pesticide applications (Shrestha et al., 2024).

In practical agricultural settings, cucumber disease diagnosis is still commonly based on visual inspection by farmers, agronomists, or plant protection specialists. Although expert evaluation is valuable, visual assessment is affected by symptom stage, observer experience, illumination, background conditions, and the visual similarity between disease classes (Emin Sahin et al., 2025; Li et al., 2024). Early symptoms may be subtle, and different diseases may share similar color changes, lesion boundaries, or tissue damage patterns. These factors can delay diagnosis or result in incorrect treatment decisions. Image-based automated screening systems can therefore be useful as complementary decision-support tools, particularly in situations where expert assessment is not immediately available.

Deep learning has become a widely used approach for plant disease recognition because it can learn discriminative visual features directly from image data. CNN-based architectures have been particularly effective in this field due to their ability to capture local texture, color, and lesion patterns

through hierarchical feature extraction (Kunduracioglu & Pacal, 2024; Pacal & Işık, 2024; Rahman et al., 2024). More recently, modern ConvNet designs and Transformer-based models have also been investigated for agricultural image classification. Modern ConvNets such as ConvNeXt incorporate several design principles inspired by recent vision architectures while retaining convolutional operations, whereas Vision Transformers use self-attention to model relationships between distant image regions (Singh et al., 2024; Woo et al., 2023). These architectural differences may influence both classification performance and computational requirements.

Several recent studies have reported strong results for cucumber and broader plant disease recognition using CNNs, Transformer-based networks, and hybrid approaches (Asha Rani & Gowrishankar, 2024; Cao et al., 2025; Emin Sahin et al., 2025; Li et al., 2024; Zeng et al., 2025). However, many studies primarily emphasize classification accuracy, while computational cost is often reported only partially or not discussed in relation to deployment constraints. This is a relevant limitation because agricultural disease recognition systems may need to operate on mobile devices, embedded platforms, or low-power hardware in field environments. In such cases, a model with marginally lower accuracy but substantially lower computational demand may be more practical than a larger model with slightly higher performance.

Another limitation in the existing literature is that model comparisons are often difficult to interpret because different studies use different datasets, class definitions, preprocessing pipelines, augmentation strategies, and validation protocols. As a result, it is not always clear whether reported performance differences are caused by the model architecture itself or by differences in experimental design. A controlled comparison under the same dataset split, input resolution, training strategy, and evaluation metrics is therefore needed to better understand the relationship between predictive performance and computational complexity in cucumber disease classification.

To address this issue, the present study evaluates four pretrained deep learning architectures under a unified experimental protocol: DenseNet201, Inception v4, ConvNeXt v2 Base, and DeiT Base. DenseNet201 and Inception v4 were selected as representative CNN architectures with different levels of complexity. ConvNeXt v2 Base was included as a modern convolutional architecture, while DeiT Base was selected as a Transformer-based model. All models were evaluated on an eight-class public cucumber disease dataset using the same preprocessing, training, validation, and test configuration. The aim is not to introduce a new architecture, but to provide a controlled comparison of accuracy, precision, recall, F1-score, parameter count, and GFLOPs. The main contributions of this study are as follows:

- A unified comparison of CNN, modern ConvNet, and Transformer-based architectures is provided for eight-class cucumber disease classification.
- Predictive performance is reported together with parameter count and GFLOPs to evaluate the accuracy–complexity trade-off.
- The results compare high-accuracy models with more compact alternatives, supporting model selection under different computational constraints.
- The study provides a controlled baseline for future cucumber disease classification studies using the same public dataset.

1. Related Work

Deep learning has been widely adopted for plant disease recognition, particularly because disease symptoms are often expressed through visual changes in color, texture, lesion shape, and tissue structure. In cucumber disease classification, early studies mainly relied on CNN-based architectures due to their strong capacity for local feature extraction. Jain et al. (2024) evaluated several standard CNN architectures, including AlexNet, GoogLeNet, and ResNet50, for cucumber disease

identification and reported the best performance with ResNet50. Pradhan et al. (2024) combined CNN-based feature extraction with a Support Vector Machine classifier and achieved competitive classification performance. Bilal et al. (2025) proposed a fuzzy deep CNN framework for cucumber plant disease detection and classification, with a focus on improving robustness under noisy image conditions.

CNN-based models have also been used in broader plant disease studies because convolutional filters are effective in capturing local disease patterns such as spots, necrotic regions, discoloration, and mildew-like textures. Kunduracioglu and Pacal (2024) investigated deep learning models for grape leaf disease classification, while Pacal and Işık (2024) compared CNN and Vision Transformer models for corn leaf disease identification. These studies show that CNNs remain strong baselines for agricultural image classification tasks, especially when the disease classes are visually separable through local texture and color cues. However, CNN-based models may vary considerably in computational cost depending on architectural depth, feature reuse strategy, and multi-scale design.

Recent work has increasingly explored Transformer-based and hybrid architectures for plant disease recognition. The motivation behind these models is that disease symptoms may not always be localized in a single region; instead, diagnostic cues can appear across multiple areas of a leaf, stem, or fruit. F. Wang et al. (2024) improved a Swin Transformer model for vegetable foliar disease identification and reported strong classification performance. Similarly, Wang Xuewei and Liu Jun (2025) developed CucumberDet by incorporating a Swin Transformer backbone to improve feature representation for small disease targets. These studies suggest that attention-based models can be useful when global context and multi-region symptom patterns are important.

Hybrid CNN-Transformer models have also been proposed to combine local and global feature learning. Yang et al. (2024) introduced LDI-NET, a framework that uses CNN and Transformer components for plant type, disease, and severity identification. Iqbal (2025) proposed PlantHealthNet, a multi-stage framework involving Swin Transformer, Detection Transformer, and SAM2-UNet components for disease diagnosis, localization, and severity estimation. Such models may improve representation capacity, but their increased architectural complexity can make them harder to deploy in practical agricultural settings unless computational cost, inference time, and memory demand are carefully considered.

Several cucumber-specific studies have reported high classification accuracy, but their experimental settings are often difficult to compare directly. Differences in dataset source, class definition, image acquisition conditions, split strategy, preprocessing, augmentation, and evaluation metrics can strongly affect the reported results. For example, a model tested on a controlled public dataset may not perform similarly on field-acquired images with complex backgrounds, variable illumination, occlusion, or overlapping symptoms. Therefore, accuracy alone is not sufficient for assessing the practical value of a disease recognition model.

A further limitation in the current literature is that computational efficiency is often treated as a secondary issue. Many studies prioritize the highest possible classification accuracy, whereas fewer studies examine how accuracy changes with parameter count, GFLOPs, and model size. This is important for cucumber disease recognition because practical systems may need to operate on mobile phones, embedded boards, or low-power agricultural monitoring devices. In such scenarios, a smaller model with slightly lower accuracy may be more useful than a larger model that requires substantially higher computational resources.

The present study addresses this gap by comparing four pretrained architectures under the same experimental protocol: DenseNet201, Inception v4, ConvNeXt v2 Base, and DeiT Base. DenseNet201 and Inception v4 are used as conventional CNN-based models with different complexity levels. ConvNeXt v2 Base is evaluated as a modern convolutional architecture, while

DeiT Base is included as a Transformer-based model. Instead of proposing a new architecture, this study focuses on a controlled comparison of predictive performance and computational complexity. This design allows the models to be assessed not only by accuracy, precision, recall, and F1-score, but also by parameter count and GFLOPs, which are important for model selection in resource-constrained agricultural applications.

2. Materials and Methods

Dataset Description and Data Partitioning

This study used the publicly available Cucumber Disease Recognition Dataset obtained from the Kaggle repository. The dataset contains 7,689 images from eight cucumber health categories: Anthracnose, Bacterial Wilt, Belly Rot, Downy Mildew, Fresh Cucumber, Fresh Leaf, Gummy Stem Blight, and Pythium Fruit Rot. Six categories represent disease conditions, while two categories correspond to healthy cucumber samples. The class-wise distribution and the train-validation-test split are given in Table 1. Representative examples from each class are shown in Figure 1.



Figure 1
Representative images from the cucumber disease dataset

Figure 1 shows sample images from the eight categories used in this study, including six diseased classes and two healthy classes. Visual differences among classes include leaf discoloration, necrotic regions, fruit surface damage, mildew-like patterns, and healthy leaf or fruit appearance.

The dataset was divided into training, validation, and test subsets using a 70%, 15%, and 15% split, respectively. This partition resulted in 5,382 training images, 1,153 validation images, and 1,154 test images. The same split was used for all evaluated architectures to ensure that model comparisons were made under identical data conditions. The validation set was used to monitor training behaviour, while the test set was used only for final performance evaluation.

Although the class counts are nearly balanced, this does not remove all possible sources of bias. Public plant disease datasets may contain visually similar samples, repeated acquisition conditions, or near-duplicate images.

Table 1

Class-wise distribution of the cucumber disease dataset

Class	Total	Train (70%)	Validation (15%)	Test (15%)
Anthracnose	960	672	144	144
Bacterial Wilt	960	672	144	144
Belly Rot	960	672	144	144
Downy Mildew	960	672	144	144
Fresh Cucumber	960	672	144	144
Fresh Leaf	960	672	144	144
Gummy Stem Blight	960	672	144	144
Pythium Fruit Rot	969	678	145	146
Total	7689	5382	1153	1154

Image Preprocessing and Data Augmentation

All images were resized to 224×224 pixels before being used as model input. This size was selected to provide a common input resolution for all evaluated architectures. Pixel values were scaled to the $[0, 1]$ range before training. The same preprocessing pipeline was applied to DenseNet201, Inception v4, ConvNeXt v2 Base, and DeiT Base.

Data augmentation was applied only to the training subset. The validation and test subsets were not augmented. The augmentation operations included random horizontal flipping, random vertical flipping, random rotation up to 20 degrees, and random zooming up to 15%. These transformations were selected because they introduce moderate spatial variation while preserving the disease-related visual characteristics of cucumber leaves and fruits. Similar augmentation practices are commonly used in image-based plant disease studies to reduce sensitivity to acquisition and orientation differences (Mumuni et al., 2024; Z. Wang et al., 2024).

No augmentation was applied before data partitioning. Applying augmentation before data partitioning may introduce visually related samples into both training and evaluation subsets; therefore, augmentation was restricted to the training set after splitting.

Evaluated Architectures

The evaluated architectures are summarized in Table 2. The study included two conventional CNN models, one modern convolutional model, and one Transformer-based model. DenseNet201 and Inception v4 were selected as CNN architectures with different complexity levels. ConvNeXt v2 Base was evaluated as a modern ConvNet. Although it adopts several design choices associated with recent vision architectures, it remains convolution-based and was not treated as a Vision Transformer. DeiT Base was included as the Transformer-based comparator.

Table 2

Evaluated architectures and their role in the comparative analysis

Model	Architectural family	Main characteristic	Role in this study	Reference
DenseNet201	CNN	Dense feature reuse	Compact CNN comparator	Huang et al. (2016)
Inception v4	CNN	Multi-scale convolutional modules	Larger CNN comparator	Szegedy et al. (2016)

ConvNeXt v2	Modern	Modernized	Modern ConvNet	Woo et al.
Base	ConvNet	convolutional design	comparator	(2023)
DeiT Base	Vision	Patch-based self-	Transformer-based	Touvron et al.
	Transformer	attention	comparator	(2020)

For CNN-based architectures, local visual patterns are extracted through convolutional kernels. The convolution operation can be written as Eq. (1):

$$X_k^{(l+1)} = \sigma \left(\sum_{c=1}^C W_{k,c}^{(l)} * X_c^{(l)} + b_k^{(l)} \right), \quad (1)$$

where $X_c^{(l)}$ denotes the c -th input feature map at layer l , $W_{k,c}^{(l)}$ is the convolution kernel, $b_k^{(l)}$ is the bias term, $*$ denotes convolution, and $\sigma(\cdot)$ is the activation function. This operation represents the local feature extraction principle used by CNN-based models such as DenseNet201 and Inception v4 (Huang et al., n.d.; Szegedy et al., 2016).

For Transformer-based image classification, the image is represented as a sequence of patches, and patch interactions are modeled using self-attention. The scaled dot-product attention operation is given in Eq. (2):

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V, \quad (2)$$

where Q , K , and V represent query, key, and value matrices, respectively, and d_k denotes the dimensionality of the key vectors. DeiT Base uses this attention mechanism to model relationships between spatially separated image patches (Touvron et al., 2020).

Hybrid CNN–Transformer models are discussed in the literature because they combine local convolutional features with attention-based global representations. A general feature-fusion formulation is shown in Eq. (3):

$$z = \phi \left([f_{\text{CNN}}(\mathbf{x}); f_{\text{ViT}}(\mathbf{x})] \right), \hat{y} = \text{softmax}(W_c z + b_c), \quad (3)$$

where $f_{\text{CNN}}(\mathbf{x})$ denotes convolutional feature extraction, $f_{\text{ViT}}(\mathbf{x})$ denotes Transformer-based feature extraction, $[\cdot; \cdot]$ indicates feature concatenation, $\phi(\cdot)$ is a fusion function, and $\hat{y}$ is the predicted class-probability vector. This formulation is included only to clarify the hybrid design family discussed in the related work. The present study does not propose or train a new hybrid architecture.

Transfer Learning Strategy

Transfer learning was used for all four models. Each architecture was initialized with ImageNet-pretrained weights. The original classification layer was replaced with a new task-specific output layer containing eight units, corresponding to the eight-cucumber disease and healthy classes.

Training was carried out in two stages. In the first stage, the pretrained feature extractor was frozen, and only the newly added classification layer was trained. In the second stage, the full model was

fine-tuned using a lower learning rate. The same transfer learning strategy was used for all architectures to reduce differences caused by the training procedure rather than the model design.

Training Configuration

The training configuration used for all evaluated models is summarized in Table 3. All experiments were implemented in Python using PyTorch. Training was performed using an NVIDIA GeForce RTX 5090 GPU. Adam was used as the optimizer, and the initial learning rate during fine-tuning was set to 1×10^{-4} . The batch size was fixed at 16, and each model was trained for a maximum of 300 epochs.

A learning-rate scheduler was used during training. When the validation loss stopped improving, the learning rate was reduced by a factor of 0.1. The same input size, optimizer, batch size, split ratio, and evaluation procedure were used for all models.

Table 3

Training configuration used for all evaluated models

Configuration item	Setting
Framework	PyTorch
Programming language	Python
Hardware accelerator	NVIDIA GeForce RTX 5090 GPU
Input size	224×224 pixels
Pretraining	ImageNet-pretrained weights
Output classes	8
Train/validation/test split	70% / 15% / 15%
Optimizer	Adam
Initial fine-tuning learning rate	1×10^{-4}
Batch size	16
Maximum epochs	300
Learning-rate schedule	Reduced by 0.1 after validation-loss plateau
Data augmentation	Applied only to training images
Final evaluation	Held-out test set

Performance Evaluation Metrics

The evaluation metrics and computational indicators used in this study are listed in Table 4. Classification performance was assessed using accuracy, precision, recall, and F1-score. Parameter count and GFLOPs were used to compare computational complexity.

Table 4

Evaluation metrics and computational indicators used in this study

Metric / indicator	Purpose	Interpretation
Accuracy	Measures the overall correct classification rate	Higher values indicate better overall classification performance
Precision	Measures the correctness of positive predictions	Useful for assessing false-positive behaviour

Recall	Measures the ability to identify class samples	Useful for assessing missed samples
F1-score	Combines precision and recall	Useful when precision and recall are considered together
Parameters	Estimates model size	Lower values indicate a more compact model
GFLOPs	Estimates computational workload	Lower values indicate lower computational demand

The classification metrics were computed using Eqs. (4)–(7). In these equations, TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (4)$$

$$Precision = \frac{TP}{TP + FP}, \quad (5)$$

$$Recall = \frac{TP}{TP + FN}, \quad (6)$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall}. \quad (7)$$

3. Results

Overall Classification Performance

The test-set results of the evaluated models are presented in Table 5. The comparison includes accuracy, precision, recall, F1-score, parameter count, and GFLOPs. These measures provide a joint view of predictive performance and computational demand. Since all models were trained and evaluated using the same dataset split, input size, preprocessing strategy, and evaluation protocol, the differences in Table 5 mainly reflect model-level behaviour under the adopted experimental setting.

Table 5

Test-set classification performance and computational complexity of the evaluated models

Models	Accuracy	Precision	Recall	F1-score	Params	GFLOPs
ConvNeXt v2 (Woo et al., 2023)	0.9913	0.9914	0.9913	0.9913	87.70M	30.7075
DeiT Base (Touvron et al., 2020)	0.9957	0.9957	0.9957	0.9957	85.80M	33.6955
DenseNet 201 (Huang et al., 2016)	0.9846	0.9772	0.9772	0.9772	18.11M	8.5795
Inception v4 (Szegedy et al., 2016)	0.9948	0.9949	0.9948	0.9948	41.16M	12.2450

DeiT Base achieved the highest performance among the evaluated architectures, with an accuracy, precision, recall, and F1-score of 0.9957. Inception v4 followed closely, reaching an accuracy of 0.9948 and an F1-score of 0.9948. ConvNeXt v2 Base also produced a high classification score, with an accuracy of 0.9913. DenseNet201 obtained the lowest accuracy among the four models, but its accuracy remained high at 0.9846.

The small difference between DeiT Base and Inception v4 indicates that the dataset can be classified accurately by both Transformer-based and convolutional architectures. However, this result should not be interpreted as evidence that one architectural family is universally superior. The difference between 0.9957 and 0.9948 is small, and additional repeated runs or statistical testing would be required to determine whether this difference is practically meaningful.

Accuracy–Complexity Trade-off

The computational cost values in Table 5 show that high classification performance was achieved with different levels of model complexity. DeiT Base produced the highest accuracy but required 85.80M parameters and 33.6955 GFLOPs. ConvNeXt v2 Base had a similar parameter count, with 87.70M parameters, and required 30.7075 GFLOPs. Inception v4 reached a test accuracy close to DeiT Base while using 41.16M parameters and 12.2450 GFLOPs. DenseNet201 was the most compact model, requiring only 18.11M parameters and 8.5795 GFLOPs.

These results show that the most accurate model is not necessarily the most efficient model. DeiT Base may be preferable when the main objective is to maximize classification accuracy under the current experimental setting. In contrast, DenseNet201 may be more suitable when computational cost is a major constraint. Inception v4 appears to offer a useful balance between performance and complexity, since its accuracy was close to DeiT Base while requiring substantially fewer GFLOPs.

However, parameter count and GFLOPs should not be treated as complete deployment indicators. They provide useful architecture-level information, but they do not directly measure inference latency, memory consumption, throughput, or energy use on target devices.

Confusion Matrix Analysis

To examine class-level prediction behaviour, the confusion matrix of the best-performing model, DeiT Base, is shown in Figure 2. The diagonal dominance of the matrix indicates that most test samples were correctly assigned to their true classes. Only a small number of errors appeared outside the diagonal, suggesting that the model made limited confusion among the eight cucumber categories.

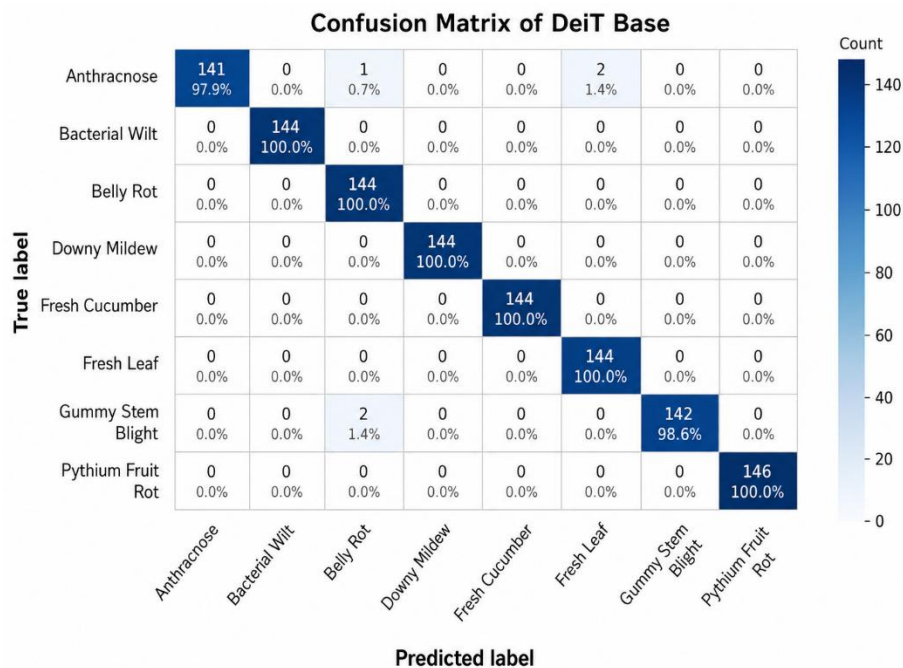


Figure 2
Confusion matrix of DeiT Base on the held-out test set

Figure 2 shows that the DeiT Base model correctly classified 1149 of 1154 test images, corresponding to an overall accuracy of 99.57%. At the class level, Bacterial Wilt, Belly Rot, Downy Mildew, Fresh Cucumber, Fresh Leaf, and Pythium Fruit Rot achieved perfect class-wise recall, with 144/144, 144/144, 144/144, 144/144, 144/144, and 146/146 true-positive predictions, respectively. However, perfect recall did not always coincide with perfect precision. For example, the Belly Rot class received three false-positive predictions, including one Anthracnose sample and two Gummy Stem Blight samples, while Fresh Leaf received two false-positive predictions from the Anthracnose class.

The main residual errors were concentrated in Anthracnose and Gummy Stem Blight. Anthracnose produced 141 true positives out of 144 samples (97.9%), with three false negatives, including one sample misclassified as Belly Rot and two samples misclassified as Fresh Leaf. Gummy Stem Blight yielded 142 true positives out of 144 samples (98.6%), with two false negatives, both assigned to Belly Rot. Notably, Anthracnose and Gummy Stem Blight showed high class-wise precision because no samples from other classes were incorrectly assigned to Anthracnose or Gummy Stem Blight. These findings indicate that the remaining errors were limited and were mainly associated with visually similar disease categories, particularly where lesion appearance, necrotic texture, or surface deterioration overlapped across classes.

4. Discussion

The results indicate that cucumber disease classification can be performed with high accuracy using different architectural families when the models are trained under the same experimental protocol. As shown in Table 5, DeiT Base achieved the highest test-set performance, while Inception v4 produced a very close result with a lower computational burden. This narrow difference suggests that the present dataset is not exclusively favourable to a single model family. Both convolutional and Transformer-based representations appear to capture the visual patterns needed for separating the eight cucumber classes. However, the small gap between DeiT Base and Inception v4 should be interpreted within the limits of the current split, since repeated runs and statistical testing were not included.

The computational comparison gives a more practical view of the results. DeiT Base and ConvNeXt v2 Base required substantially higher GFLOPs than DenseNet201 and Inception v4, whereas DenseNet201 was the most compact architecture among the evaluated models. This pattern shows that the highest classification score does not automatically correspond to the most suitable model for practical use. In server-side or laboratory settings, DeiT Base may be preferred when the main objective is maximum test-set performance. In more resource-limited settings, DenseNet201 or Inception v4 may be more appropriate depending on the acceptable balance between accuracy and computational cost. Since parameter count and GFLOPs are architecture-level indicators, they should be complemented with latency, memory, and energy measurements before making deployment-oriented decisions.

The class-level behaviour in Figure 2 is consistent with the high overall performance reported in Table 5. Most test images were correctly assigned to their true classes, and the remaining errors were concentrated in a small number of visually related categories. Anthracnose and Gummy Stem Blight accounted for the main false-negative cases, while Belly Rot and Fresh Leaf received a small number of false-positive predictions. This pattern is plausible because cucumber diseases can share similar surface discoloration, necrotic texture, and tissue deterioration patterns, especially when symptoms are mild or partially overlapping. Even so, the current interpretation is based on the confusion matrix rather than expert inspection of misclassified images. Therefore, the findings should be viewed as strong within-dataset evidence, while external validation with field-acquired images is still needed to assess robustness under natural agricultural conditions.

5. Limitations and Future Work

The main limitation of this study is its reliance on a single public dataset. Although the dataset is nearly balanced, it may not represent the full variability of field conditions. In practical agricultural settings, images may include strong illumination changes, shadows, occlusion, overlapping leaves, different cameras, background clutter, and disease symptoms at different stages. Therefore, the reported results should be interpreted as within-dataset performance rather than direct evidence of field-level generalization.

Second, the study does not report duplicate or near-duplicate image screening, repeated training with different random seeds, confidence intervals, or statistical significance testing. These analyses are important because the reported accuracies are very high and the performance gap between DeiT Base and Inception v4 is small. Future studies should include repeated runs, mean $\pm$ standard deviation reporting, and statistical testing to determine whether the observed differences between models are stable and meaningful.

Third, computational efficiency was evaluated using parameter count and GFLOPs, but these indicators do not fully describe deployment performance. Actual inference speed, memory use, throughput, and energy consumption can change depending on the target hardware and software backend. Future work should therefore evaluate the models on independent field-acquired images and report device-level latency, model size, memory consumption, and energy use. Model compression methods such as pruning, quantization, and knowledge distillation may also be investigated for real-time agricultural use.

6. Conclusion

This study compared DenseNet201, Inception v4, ConvNeXt v2 Base, and DeiT Base for eight-class cucumber disease classification using a public image dataset. Under the adopted experimental protocol, DeiT Base achieved the highest test performance with an accuracy, precision, recall, and F1-score of 0.9957. Inception v4 produced a very similar result with an accuracy of 0.9948, while ConvNeXt v2 Base reached 0.9913. DenseNet201 achieved 0.9846 accuracy and had the lowest computational cost among the evaluated models. The results show that high classification accuracy can be obtained with both convolutional and Transformer-based architectures on this dataset. However, accuracy alone is not sufficient for model selection. DeiT Base may be preferred when maximum predictive performance is required, whereas DenseNet201 is more suitable when computational cost is the main constraint. Inception v4 appears to offer a useful balance between accuracy and complexity. Since the study is limited to a single public dataset and does not include external validation or device-level testing, the findings should be viewed as a controlled benchmark rather than proof of deployment readiness.

Declarations

Funding

The authors received no specific funding for this study.

Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Data Availability

The dataset used in this study is publicly available from the Kaggle repository under the title Cucumber Disease Recognition Dataset. The dataset contains cucumber disease and healthy-category images used for the eight-class classification experiments.

Declaration of Generative AI and AI-Assisted Technologies

During the preparation of this manuscript, AI-assisted tools were used only for language editing, grammar correction, and improving the clarity of academic expression. These tools were not used to generate the scientific methodology, experimental results, data analysis, model outputs, figures, or conclusions. All content was reviewed, verified, and approved by the authors, who take full responsibility for the accuracy and integrity of the manuscript.

References

- Asha Rani, K. P., & Gowrishankar, S. (2024). State-of-the-Art Deep Learning Architectures for Identifying Diseases in Cucumber and Black Gram Plants. *IEEE International Conference on Data Engineering and Communication Systems, ICDECS 2024*. <https://doi.org/10.1109/ICDECS59733.2023.10503021>
- Bilal, A., Khan, J. A., Alzahrani, A., Almohammadi, K., Alamri, M., & Liu, X. (2025). Fuzzy deep learning architecture for cucumber plant disease detection and classification. *Journal of Big Data*, 12(1). <https://doi.org/10.1186/s40537-025-01156-z>
- Cakmak, Y. (2025). Machine Learning Approaches for Enhanced Diagnosis of Hematological Disorders. *Computational Systems and Artificial Intelligence*, 1(1), 8–14. <https://doi.org/10.69882/ADBA.CSAI.2025072>
- Cakmak, Y., & Maman, A. (2025). Deep Learning for Early Diagnosis of Lung Cancer. *Computational Systems and Artificial Intelligence*, 1(1), 20–25. <https://doi.org/10.69882/ADBA.CSAI.2025074>
- Cakmak, Y., & Pacal, N. (2025). Deep Learning for Automated Breast Cancer Detection in Ultrasound: A Comparative Study of Four CNN Architectures. *Artificial Intelligence in Applied Sciences*, 1(1), 13–19. <https://doi.org/10.69882/ADBA.AI.2025073>
- Cakmak, Y., & Zeynalov, J. (2025). A Comparative Analysis of Convolutional Neural Network Architectures for Breast Cancer Classification from Mammograms. *Artificial Intelligence in Applied Sciences*, 1(1), 28–34. <https://doi.org/10.69882/ADBA.AI.2025075>
- Cao, Y., Sun, G., Yuan, Y., & Chen, L. (2025). Small-sample cucumber disease identification based on multimodal self-supervised learning. *Crop Protection*, 188. <https://doi.org/10.1016/j.cropro.2024.107006>
- Cucumber Disease Recognition Dataset*. Retrieved October 10, 2025, from <https://www.kaggle.com/datasets/sujaykapadnis/cucumber-disease-recognition-dataset>
- Emin Sahin, M., Özkaya, U., Arisoy, C., Coşar, H. İ., & Ulutaş, H. (2025). CucuNetCNNs: Application of novel ensemble deep neural networks for classification of cucumber leaf disease. *Ain Shams Engineering Journal*, 16(5). <https://doi.org/10.1016/j.asej.2025.103380>
- Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. 2016. *Densely Connected Convolutional Networks*. Retrieved July 23, 2025, from <https://github.com/liuzhuang13/DenseNet>.
- Iqbal, A. (2025). PlantHealthNet: Transformer-Enhanced Hybrid Models for Disease Diagnosis and Severity Estimation in Agriculture. *IEEE Access*, 13, 101160–101176. <https://doi.org/10.1109/ACCESS.2025.3576990>
- Jain, P., Gupta, V. K., Sharma, G., Dhiman, G., Meena, J., & Singh, M. K. (2024). Detection of Cucumber Diseases using Deep Learning models. *2024 5th IEEE Global Conference for*

- Advancement in Technology, GCAT 2024.*
<https://doi.org/10.1109/GCAT62922.2024.10924118>
- Kumar, R. R., Jain, A. K., Sharma, V., Midha, M., & Das, P. (2024). Enhancing Cucumber Leaf Disease Severity Prediction through Convolutional Neural Networks and Random Forest Integration. *2024 5th International Conference for Emerging Technology, INCET 2024.* <https://doi.org/10.1109/INCET61516.2024.10593479>
- Kunduracioglu, I., & Pacal, I. (2024). Advancements in deep learning for accurate classification of grape leaves and diagnosis of grape diseases. *Journal of Plant Diseases and Protection*, 131(3), 1061–1080. <https://doi.org/10.1007/S41348-024-00896-Z/TABLES/7>
- Li, R., Su, X., Zhang, H., Zhang, X., Yao, Y., Zhou, S., Zhang, B., Ye, M., & Lv, C. (2024). Integration of Diffusion Transformer and Knowledge Graph for Efficient Cucumber Disease Detection in Agriculture. *Plants*, 13(17). <https://doi.org/10.3390/plants13172435>
- Liu, Y., Lan, Y., & Chen, X. (2025). Partial convolutional biformer: A transformer architecture for diagnosing crop diseases under complex backgrounds. *Crop Protection*, 193. <https://doi.org/10.1016/j.cropro.2024.107007>
- Mumuni, A., Mumuni, F., & Gerrar, N. K. (2024). A Survey of Synthetic Data Augmentation Methods in Machine Vision. *Machine Intelligence Research 2024 21:5*, 21(5), 831–869. <https://doi.org/10.1007/S11633-022-1411-7>
- Pacal, I. (2024). Enhancing crop productivity and sustainability through disease identification in maize leaves: Exploiting a large dataset with an advanced vision transformer model. *Expert Systems with Applications*, 238, 122099. <https://doi.org/10.1016/J.ESWA.2023.122099>
- Pacal, I., Kunduracioglu, I., Alma, M. H., Devenci, M., Kadry, S., Nedoma, J., Slany, V., & Martinek, R. (2024). A systematic review of deep learning techniques for plant diseases. *Artificial Intelligence Review 2024 57:11*, 57(11), 1–39. <https://doi.org/10.1007/S10462-024-10944-7>
- Pacal, I., & Işık, G. (2024). Utilizing convolutional neural networks and vision transformers for precise corn leaf disease identification. *Neural Computing and Applications*, 37(4), 2479–2496. <https://doi.org/10.1007/S00521-024-10769-Z/TABLES/5>
- Pradhan, G., Sharma, R. T., Shah, A. K., & Kukreja, V. (2024). Deep Learning for Cucumber Agriculture: A Hybrid CNN-SVM System for Disease Identification. *2024 International Conference on Automation and Computation, AUTOCOM 2024*, 57–61. <https://doi.org/10.1109/AUTOCOM60220.2024.10486173>
- Rahman, M. S., Ahamed, A., Pranto, M. N., Islam, M. A., Al Masum, A., Al-Sakib, A., Rahman, S. M. A., Haque, R., & Rahman, S. (2024). Effective Disease Recognition in Cucumbers: A Web-Based Application Using Transfer Learning Models. *2024 IEEE 3rd International Conference on Robotics, Automation, Artificial-Intelligence and Internet-of-Things, RAAICON 2024 - Proceedings*, 59–64. <https://doi.org/10.1109/RAAICON64172.2024.10928353>
- Shrestha, T. E., Aurnob, A. R., Tanim, S. A., Islam, M., & Nur, K. (2024). Revolutionizing Cucumber Agriculture: AI for Precision Classification of Leaf Diseases. *Proceedings - 6th International Conference on Electrical Engineering and Information and Communication Technology, ICEEICT 2024*, 776–781. <https://doi.org/10.1109/ICEEICT62016.2024.10534530>
- Singh, A. K., Rao, A., Chattopadhyay, P., Maurya, R., & Singh, L. (2024). Effective plant disease diagnosis using Vision Transformer trained with leafy-generative adversarial network-generated images. *Expert Systems with Applications*, 254. <https://doi.org/10.1016/j.eswa.2024.124387>
- Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. 2016. *Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning*. Retrieved July 23, 2025, from www.aaii.org
- Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H., & Ai, F. (2020). *Training data-efficient image transformers & distillation through attention*.
- Wang, F., Wang, Y., Liu, M., Chen, W., Wang, Y., & Xi, Y. (2024). Vegetable Disease Identification Algorithm Based on Improved Swin Transformer. *Proceedings of 2024 8th Asian Conference*

- on *Artificial Intelligence Technology, ACAIT* 2024, 1841–1845. <https://doi.org/10.1109/ACAIT63902.2024.11022041>
- Wang Xuwei, & Liu Jun. (2024). *Detection of small targets in cucumber disease images through global information perception and feature fusion _ Enhanced Reader*.
- Wang, Y., Li, T., Chen, T., Zhang, X., Taha, M. F., Yang, N., Mao, H., & Shi, Q. (2024). Cucumber Downy Mildew Disease Prediction Using a CNN-LSTM Approach. *Agriculture (Switzerland)*, 14(7). <https://doi.org/10.3390/agriculture14071155>
- Wang, Z., Wang, P., Liu, K., Wang, P., Fu, Y., Lu, C.-T., Aggarwal, C. C., Pei, J., & Zhou, Y. (2024). *A Comprehensive Survey on Data Augmentation*. <https://arxiv.org/abs/2405.09591v3>
- Woo, S., Debnath, S., Hu, R., Chen, X., Liu, Z., Kweon, I. S., & Xie, S. (2023). ConvNeXt V2: Co-designing and Scaling ConvNets with Masked Autoencoders. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2023-June*, 16133–16142. <https://doi.org/10.1109/CVPR52729.2023.01548>
- Yang, B., Li, M., Li, F., Wang, Y., Liang, Q., Zhao, R., Li, C., & Wang, J. (2024). A novel plant type, leaf disease and severity identification framework using CNN and transformer with multi-label method. *Scientific Reports*, 14(1). <https://doi.org/10.1038/s41598-024-62452-x>
- Zeng, Z., Mahmood, T., Wang, Y., Rehman, A., & Mujahid, M. A. (2025). AI-driven smart agriculture using hybrid transformer-CNN for real time disease detection in sustainable farming. *Scientific Reports*, 15(1). <https://doi.org/10.1038/s41598-025-10537-6>
- Zeynalov, J., Çakmak, Y., & Paçal, İ. (2025). Automated Apple Leaf Disease Classification Using Deep Convolutional Neural Networks: A Comparative Study on the Plant Village Dataset. *Journal of Computer Science and Digital Technologies*, 1(1), 5–17. <https://doi.org/10.61640/jcsdt.2025.0601>