

Article

Visual Intelligence in Precision Agriculture: Exploring Plant Disease Detection via Efficient Vision Transformers

Sana Parez ¹, Naqqash Dilshad ², Norah Saleh Alghamdi ³, Turki M. Alanazi ⁴ and Jong Weon Lee ^{1,*}¹ Department of Software, Sejong University, Seoul 05006, Republic of Korea; parez.sana@gmail.com² Department of Convergence Engineering for Intelligent Drone, Sejong University, Seoul 05006, Republic of Korea; dilshad.naqqash@gmail.com³ Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia; nosalghamdi@pnu.edu.sa⁴ Department of Electrical Engineering, College of Engineering, Jouf University, Sakaka 72388, Saudi Arabia; tmanazi@ju.edu.sa

* Correspondence: jwlee@sejong.ac.kr

Abstract: In order for a country's economy to grow, agricultural development is essential. Plant diseases, however, severely hamper crop growth rate and quality. In the absence of domain experts and with low contrast information, accurate identification of these diseases is very challenging and time-consuming. This leads to an agricultural management system in need of a method for automatically detecting disease at an early stage. As a consequence of dimensionality reduction, CNN-based models use pooling layers, which results in the loss of vital information, including the precise location of the most prominent features. In response to these challenges, we propose a fine-tuned technique, *GreenViT*, for detecting plant infections and diseases based on Vision Transformers (ViTs). Similar to word embedding, we divide the input image into smaller blocks or patches and feed these to the ViT sequentially. Our approach leverages the strengths of ViTs in order to overcome the problems associated with CNN-based models. Experiments on widely used benchmark datasets were conducted to evaluate the proposed *GreenViT* performance. Based on the obtained experimental outcomes, the proposed technique outperforms state-of-the-art (SOTA) CNN models for detecting plant diseases.

Keywords: agriculture monitoring; deep learning; embedded vision; Internet of Things (IoT); image classification; plant disease detection; vision transformers; precision agriculture



Citation: Parez, S.; Dilshad, N.; Alghamdi, N.S.; Alanazi, T.M.; Lee, J.W. Visual Intelligence in Precision Agriculture: Exploring Plant Disease Detection via Efficient Vision Transformers. *Sensors* **2023**, *23*, 6949. <https://doi.org/10.3390/s23156949>

Academic Editors: Laura Belli, Luca Davoli, Tommaso Ganino, Ilaria Marchioni and Giuseppe Vignali

Received: 6 July 2023

Revised: 31 July 2023

Accepted: 1 August 2023

Published: 4 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Over the past few decades, agriculture has emerged as the primary source of income for several countries, significantly contributing to the global economy. As per the World Bank report of 2018, agriculture engaged over a billion population, representing 28.5% of the total labor force, and amounted to about 10 million tons of food a day [1]. Although, the total potential of agriculture is prone to plant infections and diseases, food security is compromised by such infections. Major food crops, such as rice, wheat, potatoes, soybeans and maize can suffer losses of 10% to 40% due to plant viruses [2]. Addressing these challenges necessitates frequent inspection of disease symptoms, often inefficient and time consuming, particularly for huge crop fields. In order to achieve precision agriculture, plant infections must be detected effectively. Research groups have been motivated to work on Machine Learning (ML) as a result of the proliferation of this field to explore its potential in automating the detection of plant diseases by analyzing images obtained from fields. To identify diseases, these groups analyze the images and extract significant features. For example, in Ref. [3], Support Vector Machine (SVM) was applied after extracting image features using Scale Invariant Feature Transform (SIFT) to classify guava leaf diseases.

Before applying SVM, Ref. [4] used a Statistical Gray Level Co-occurrence Matrix instead of SIFT. A variety of feature extraction methods have been utilized by other studies to analyze plant diseases in order to achieve remarkable results, including [5–8].

To handle fiddly data designs and stupendous training data, some researchers used the k-Nearest Neighbors (k-NN) classifier instead of SVM. It was possible to classify Cotton Grey Mildew disease based on local statistical input features and k-NN in [9]. There has also been use of the k-NN algorithm in [10] for classifying paddy leaves and [11] for classifying groundnut leaf diseases. However, every approach requires multiple steps in order to prepare the data for preprocessing and feature extraction. There is also no evidence that they are effective at classifying more than one class of data and that they are sensitive to predefined parameters, such as the kernel parameter k in SVM and the kernel parameter f in K-NN [12]. To address these issues, some researchers have turned to Deep Learning (DL) methods for improved crop infection and disease detection. The authors in [13] created a DL-based system called PlantVillage (PV), which can accurately identify 26 different plant diseases. In contrast to explicit feature extraction techniques, DL techniques automatically learn and extract relevant features from the input images. However, traditional ANN classifiers lose spatial information when converting 2D images to 1D vectors for classification, leading to increased computational complexity and storage requirements. The article in [14] describes a method for defining diseases of plants based on a new dataset called DRLI.

Agricultural applications of neural networks, specifically convolutional neural networks (CNNs), have proven successful in overcoming previous limitations with Deep Neural Networks (DNNs) [15]. For example, Ref. [16] used the MaskRCNN model with transfer learning to detect fusarium head blight disease in wheat, achieving an average accuracy of 92.01% on the intended test data, which included around 450 images. In a similar manner, a process was applied to analyze [17] and identify apple leaf diseases with 77.65%, 75.59% and 73.50% recognition accuracy using ResNet152, Inception V3 and MobileNet models. According to [18], the authors proposed using the PV dataset to generate a custom DCNN to classify cucumber infections that performed up to 94% accurately, using a pretrained AlexNet model. In [19], the authors developed a Custom-Net model for classifying pearl millet diseases using Raspberry Pi (RPi), achieving an accuracy of 98.78%. Additionally, DL models have been applied to detect further leaf infections, mostly established on the PV dataset. According to the mentioned methods, the accuracy of classification has been high, especially in the case of tomato leaf diseases, which obtained 97.49% results in [20], and with banana leaf diseases, which obtained 99.72% results in [21].

Moreover, AI models and ML techniques have been deployed on drones, which are unmanned aerial vehicles used to mitigate various malicious factors in agriculture, including lack of rain, nutrition abnormalities, infections, weeds and pests. Precision agriculture uses drones widely because they are affordable, have an extensive range of operation and are AI-compatible, according to [22]. Based on the IoT architecture used by [23], the researchers equipped a drone with an RPi 4 to classify plant infections early in crop production by using the IoT. As reported in [24], a drone-based fidelity farming apparatus is being used to detect affected areas in open-field tomato crops using algorithmic neural networks. Depending on the level of infection, precise pesticides could be sprayed on afflicted areas. In a similar way, the [25] project constructed an automated system to detect and spray chemicals on infected plants by combining high-quality cameras with disease detection models based on ResNet architecture.

While SOTA DL models can achieve high-performance results and are suitable for use with drones, they require significant computational resources for training. In contrast, ViT which avoids CNN and has similar performance to SOTA CNN models, is a promising alternative [26]. As a derivative of Transformer, ViT employs a self-attention mechanism that determines a global reference pixel-by-pixel during training. Each given image is split into equal patches and each patch is embedded with its position. After learning from patches, the self-attention block can accurately represent them for vision tasks. In the

last layers of the ViT model, the cosine similarity between patch representations increases significantly, suggesting that increasing the layers does not enhance model performance [27]. The memory requirements of ViT make handling high-resolution images difficult due to their four times length requirement.

Several studies have aimed to overcome the limitations of Transformer-based models, and in general, it falls into two categories: hybrid models and pure-Transformer enhancement. Hybrid models combine the strengths of CNN and Transformer to improve performance. A model based on CNN called Ghost-Enlightened Transformer, for example, was proposed by [28] to construct intermediate feature maps. In the next step, the self-attention mechanism is used to convert those maps into deep semantic features. Based on 12,615 images collected by the author, this model achieved 98.14% accuracy. A similar system is outlined in [29]. As is PlantXViT, it incorporates a VGG16 network, a transmission block and an encoder layer called Transformer. The VGG16 and inception block provide better capture of local image features than SOTA CNN models currently available. Furthermore, multiple studies have incorporated CNN layers into Transformer architectures to amplify the capability of extraction of most prominent features [30–32]. As a result of this approach, the model becomes more accurate because it is able to learn local features through the CNN architecture, but the training and inference times are significantly extended and the memory used is huge.

In contrast, pure-Transformer enhancement variants operate primarily based on optimizing the self-attention mechanism to improve performance. Based on shifted windows, the Swin Transformer, for example, calculates local attention efficiently while maintaining connections across windows [33]. Additionally, Ref. [34] developed k-NN attention, which determines the attention matrix based on the top-k related tokens found in the keys, thereby reducing training time. In RegionViT, local self-attention is employed to retain global information through a regional-to-local concept [35]. Several studies have also proposed modifying the self-attention mechanism by using feature channels instead of tokens in the calculation of the self-attention matrix [36] and revamping the spatial attention mechanism to include small-distance, large-distance and all-inclusive information [37]. It involves optimizing the attention matrix calculation process in order to decrease the model's complexity while maintaining the global connection. There are, however, some studies that maintain the original architecture of the self-attention mechanism, leading to a huge number of trainable parameters in each self-attention head in comparison with previous studies. Thus, existing Transformer-based models retain their complexity while being larger. Transformer-based models have these limitations, which hinder their application to intelligent edge applications, such as drones and single-board computers, where resources are limited. We designed the models so that they could be deployed and operated on products that have limited resources, with the aim of minimizing transmission latency and network bandwidth consumption [38]. In summary, this study made the following contributions:

1. Plant disease detection is now significantly improved using CNN-based models, based on the latest research findings. However, the particular models exhibit limitations such as translation invariance, locality sensitivity and a lack of global image comprehension. To address these shortcomings inherent in CNN-based approaches, this study introduces a new approach utilizing a Vision Transformer-based model for improved and effective plant disease classification.
2. Drawing inspiration from the Vision Transformer (ViT) proposed by Alexey Dosovitskiy et al. [26], we conducted training and fine-tuning of the ViT model, specifically for fire detection, resulting in notable advancements surpassing the SOTA CNN models. By improving the architecture of the ViT model, it has been possible to reduce the number of learning parameters from 85 million to 21.65 million as a result of the fine-tuning process, which has resulted in an increase in the accuracy of the model at the same time.

3. The proposed *GreenViT* model exhibits exceptional accuracy and effectively reduced the occurrence of false alarms. Consequently, the developed system proves to be highly suitable for accurate plant disease detection, ultimately mitigating the risks associated with food scarcity and security.

This paper is further divided into the following sections: In Section 2, the proposed methodology is presented, outlining the key steps and techniques employed in the study. Section 3 provides a brief description of the experimental results obtained from the conducted experiments. Finally, in Section 4, the paper is concluded, summarizing the main findings, contributions and prospective approach for future work.

2. Material and Methods

This section begins by introducing the experimental dataset used in the study. Subsequently, the plant disease detection model, named *GreenViT*, is presented. It is necessary to review the experimental environment, as well as evaluation metrics in order to evaluate the performance of the model.

2.1. Datasets

To gauge the effectiveness of the proposed model, the study utilized two popular standard datasets, namely PV and DRLI. Furthermore, to test the model's resilience, a new dataset called PC dataset was utilized, which was created by integrating both datasets. The combined datasets' statistics are listed in Table 1 while the comprehensive details are provided below.

Table 1. The statistics of all three included datasets.

S. No.	Dataset	Training	Testing	Validation	Total Images
1	PlantVillage [13]	39,100	10,861	4344	54,305
2	Data Repository of Leaf Images [14]	3241	901	360	4502
3	Plant Composite [39]	42,341	11,762	4704	58,807

2.1.1. Plant Village

A feasibility study was conducted to assess the effectiveness of the newly presented *GreenViT* method; the authors conducted experiments on two well-known benchmark datasets PV and DRLI. The PV dataset has been widely utilized in previous studies due to its large size, public availability and free access to data on crop leaf disease classification. To validate the classification accuracy of the employed approach, the authors carried out several experiments on this dataset, which comprises images of plants with various types of diseases. The dataset contains a total of 54,303 images from 14 plant species, which are categorized into 38 classes. Of these, 26 classes correspond to infected plants, while 12 belong to healthy plants. The dataset includes images of plants such as tomatoes, strawberries, grapes and oranges. In addition to variations in color, size and lighting, the dataset features image distortions such as noise, blurring and color variations, making it a challenging dataset for detecting and categorizing affected plant leaf regions.

2.1.2. Data Repository of Leaf Images

The intricate interaction between plants and their surroundings leads to the production of various substances that enhance the environment and help in controlling greenhouse gases and climate change. However, in the past, humans have ruthlessly exterminated many plant species, resulting in the loss of biodiversity and further exacerbating climate change. To address this, the identification, detection and diagnosis of plant diseases have become crucial. In this dataset, the authors have chosen twelve plant species, including guava, arjun, mango, alstonia, bael, scholaris, jatropha, jamun, pomegranate, basil and lemon. The leaves of these plants were photographed in both healthy and infected states and were divided into two categories: healthy and infected. The entire dataset contains

approximately 4503 photos, with 2278 healthy leaves and 2225 diseased leaves, taken from March to May 2019 at the University of Shri Mata Vaishno Devi in Katra. The dataset was divided into 22 subject groups based on plant species, and the photographs were captured in an enclosed space using a Nikon D5300 camera (Nikon, Tokyo, Japan) with an 18–55 mm lens and sRGB color representation. The photos were taken with 1000 ISO and without flash, resulting in a single JPEG photo in 0.58 s per frame and a RAW + JPEG photo in 0.63 s per frame.

2.1.3. Plant Composite

In order to evaluate the robustness of the proposed *GreenViT* model, the authors conducted an experiment using a combination of publicly available datasets: PV and the DRLI. By merging these datasets, a new and more diverse dataset was created, which posed greater challenges for the model. The composite dataset consists of a total of 58,807 images, making it 7.6% larger than PV and 92.3% larger than the data repository of leaf images. This increased size and diversity of plant species within the dataset necessitated a meticulous training process for the model. As a result, the model demonstrated improved the generalization ability and enhanced the reliability for real-time plant disease detection scenarios by providing a visual representation.

2.2. The Proposed *GreenViT* Plant Disease Detection Method

The proposed framework has been thoroughly outlined in this section. A Transformer model forms the foundation for our framework. Currently, the Transformer model is widely regarded as the SOTA in handling sequential data processing, particularly in Natural Language Processing (NLP) tasks for instance speech recognition, language modeling and machine translation. The Transformer architecture, introduced by [40], revolves around an encoder–decoder module that facilitates the rearranging and incorporating of a given sequence of elements into a new sequence. The primary objective behind the development of Transformers was to enable parallel processing of data. The purpose of this study is to evaluate the performance of the ViT model in predicting plant diseases. As depicted in Figure 1, the ViT architecture is employed, which takes an input image with dimensions of 72×72 pixels. Initially, the input image is divided into patches, and the number of patches utilized depends on the specific scenario being addressed. In this study, the input image is converted into six image patches. To accommodate 2D images with height (H), width (W) and (C) channels, the image, denoted as $X \in \mathbb{R}^{(H \times W \times C)}$ is reshaped into a sequence structure resembling word embedding. This transformed representation is then used as input to the transformer network, which processes the 2D patches (P) $X_p \in \mathbb{R}^{N(P^2, C)}$. This is a representation of the actual image X_p , and the resolution of the patches is characterized by (P, P) . The most functional length of the sequence for the transformer is determined via $N = HW/P^2$. In the transformer network, these patches are treated in a similar manner as tokens in NLP. In each layer of the transformer, a fixed width is maintained, and a trainable linear projection is applied to map each vectorized patch to the model dimension D . The resulting outputs are referred to as patch embedding. The ViT model incorporates three main components: the embedding layer, the encoder layer and the classifier layer. These components will be discussed in detail as follows:

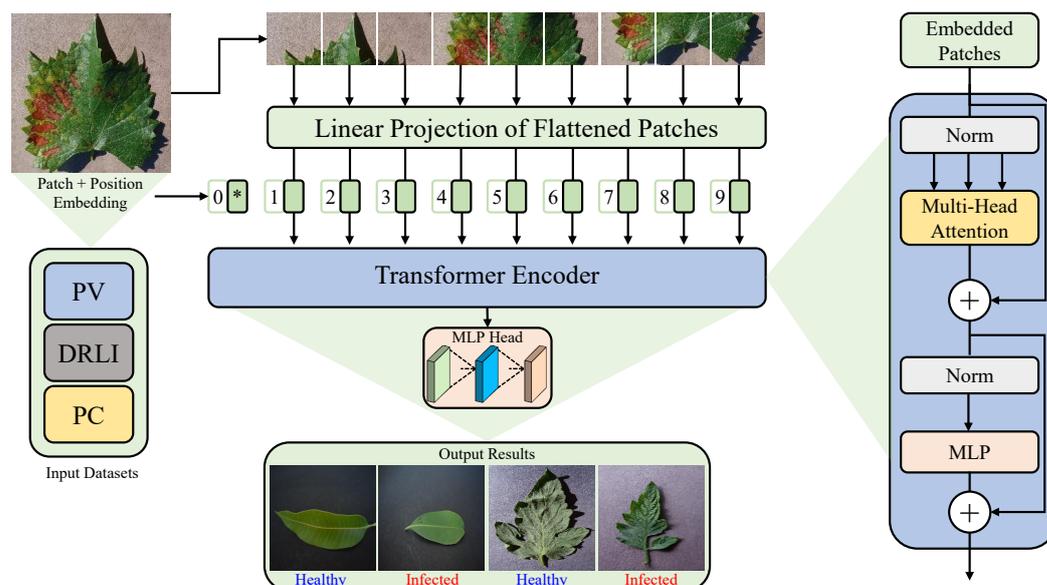


Figure 1. An overview of the proposed *GreenViT* framework for diagnosing plant diseases. The (*) notation in patch + position embedding represents class token, indicating that it depicts the information about the entire image in the sequence of patch embeddings.

2.2.1. Embedding Layer

Transform models treat patches individually as tokens and map them to higher dimensions through learnable linear projections. These embedded projections are then combined with a learnable class token U_{Class} that plays a crucial role in the classification process. To preserve the positional information and to retain the spatial positioning of the patches, positional embedding $E_{Position}$ is employed. Each patch in the image can be located precisely based on these positional embedding. The patch concatenated with the token γ_0 is represented by the following Equation (1):

$$\gamma_0 = (U_{Class}; X_p^1 E; X_p^2 E; \dots; X_p^n E) \quad (1)$$

This equation captures the fusion of the class token U_{Class} with the encoded patches to form the final input representation for further processing in the model.

2.2.2. Encoding Layer

In this particular step, the transformer encoder plays a crucial role in processing a sequence of embedded patches, denoted as γ_0 . The ViT utilizes a set of L encoder blocks, which are further subdivided into two distinct sub-components: Multi-Head Self-Attention (MHSA) and the Multi-Layer Perceptron (MLP). The MHSA block serves as a pivotal component within the encoder block, incorporating self-attention and concatenation layers. Specifically, given an input $x = x_1, x_2, \dots, x_n$, an attention operation is performed with the transformer on a set of queries Q using all available keys K and values V . This process is represented in Equation (2).

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

In Equation (2), the weight matrices W^Q , W^K and W^V are trainable parameters that determine the importance or weight assigned to the value, query and key, respectively. The process involves calculating the dot product of the queries Q across all keys K , scaling it by the square root of D and applying a SoftMax classifier for classification. The transformer executes multiple parallel iterations of scaled dot product attention using different weights,

known as attention heads. The outputs of these attention heads are then merged together to calculate the end result, as listed in the Equation (3).

$$\text{MHSA}(Q, K, V) = \text{Concatenate}\left(\text{Attention}^1, \text{Attention}^2, \dots, \text{Attention}^n\right)W^0 \quad (3)$$

In Equation (3), W_i^Q , W_i^K , W_i^V and W^0 refer to the trainable parameter matrices. The final output at the l th layer of the MHSA block is formulated in Equation (4).

$$Z_l' = \text{MHSA}(\text{LN}(Z_{l-1})) + Z_{l-1}, \text{ where } l = 1, 2, 3, \dots, L \quad (4)$$

The MLP comprises two fully connected layers, which are connected sequentially. Following the fully connected layers, the ReLU activation function is applied. The output of the MLP is provided in Equation (5).

$$Z_l = \text{MLP}(\text{LN}(Z_l')) + Z_l', \text{ where } l = 1, 2, 3, \dots, L \quad (5)$$

2.2.3. Classification Layer

In the given sequence, the very initial entity Z_1^0 is extracted and passed on to an external head classifier responsible for predicting the last layer of the encoder. The head classifier performs classification by assigning the input to one of two corresponding class labels: "Healthy" or "Infected". The formulation for this classification process is provided below in Equation (6).

$$y = \text{LN}(Z_1^0) \quad (6)$$

Alexey Dosovitskiy et al. [26] proposed three fundamental versions of the ViT, namely ViT-Base, ViT-Large and ViT-Huge. In each version, the number of encoders, hidden dimensions, attention heads and classifiers differ. The ViT-Base variant is trained using a patch size of 16×16 , employing 12 layers in the encoder, a hidden size of 768 and 12 attention heads. On the other hand, the ViT-Large and ViT-Huge versions are computationally more demanding. For a detailed overview of the specifications for each version, please refer to Table 2.

Table 2. The three basic ViT architectures, namely ViT-Base, ViT-Large and ViT-Huge, can be compared based on their layer count, hidden size (D), number of attention heads and parameters. The proposed *GreenViT* model is highlighted in blue color.

Model	No. of Layers	Hidden Size (D)	Heads	Parameters (M)
ViT Base	12	768	12	86
ViT Large	24	1024	16	307
ViT Huge	32	1280	16	632
<i>GreenViT</i>	8	768	4	21.65

During the experiments, the ViT-Base model is fine-tuned with specific configurations. The projection dimensions, number of heads, transformer layers and MLP head units are set to 64, 4, 8 and 1024, respectively. Following the MLP heads, a SoftMax classifier is employed for classification, distinguishing between two classes: "Healthy" and "Infected". The tweaking process of the ViT-Base method successfully reduces the total number of learning parameters without compromising the overall performance.

3. Experimental Results

This section delves into the evaluation and assessment metrics, and graphical outcomes. We begin by describing the experimental setup and performance metrics. Then, we discuss the evaluated results. All models, including our proposed *GreenViT*, underwent training for a total of 10 epochs, employing a low learning rate to ensure the retention of previously acquired knowledge. The pre-trained model continually updated its learning parameters to optimize performance on the designated dataset. After obtaining the results,

each model underwent retraining using its default input size of 224×224 while the proposed *GreenViT* utilized 72×72 , also employing a batch size of 32. The Adam optimizer was utilized with a learning rate of 1×10^{-4} and momentum of 0.9. The experiments were conducted on an NVIDIA GeForce RTX 3090 Graphical Processing Unit (GPU) that has 24 GB on-chip memory, equipped with 64 GB of onboard memory (Nvidia Corporation, Santa Clara, CA, USA). The single-precision floating-point computing capability of the GPU can achieve a peak performance of 36 TFLOPS. For implementation, we utilized the Keras DL framework with TensorFlow 2.9.1 serving as the backend.

3.1. Evaluation Metrics

The proposed *GreenViT* model was assessed based on various evaluation metrics, such as precision, recall, F1-score and accuracy, where TP represents True Positive, TN represents True Negative, FP represents False Positive and FN depicts False Negative.

$$\text{Accuracy} = \left(\frac{\text{TP} + \text{FN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \right), \quad (7)$$

$$\text{Precision} = \left(\frac{\text{TP}}{\text{TP} + \text{FP}} \right), \quad (8)$$

$$\text{Recall} = \left(\frac{\text{TP}}{\text{TP} + \text{FN}} \right), \quad (9)$$

$$\text{F1-score} = 2 \times \left(\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \right). \quad (10)$$

3.2. Quantitative Results

This study conducted a comparison between the proposed *GreenViT* and various pre-trained CNN-based architectures for plant disease detection. The evaluation focused on the parameters, precision, recall, F1-score and accuracy. Among the models examined, such as VGG19, VGG16, EfficientNetB0, MobileNetV1 and MobileNetV3Small, most of them demonstrated similar performance. However, the base ViT performed the worst compared with the other models; on the other hand, the proposed *GreenViT* model achieved superior accuracies of 100%, 98% and 99% on all three datasets, while also exhibiting the lowest False Alarm Rate (FAR) compared with the other SOTA models. Notably, when comparing the proposed *GreenViT* with MobileNetV1, both models demonstrated computational efficiency, but the proposed *GreenViT* showcased low FAR and still outperformed all the included datasets. A detailed performance comparison of the employed models is listed in Table 3. It is evident that the pre-trained models achieve high performance with a low FAR. Nevertheless, the FAR remains elevated and necessitates improvement. Consequently, this research explores the refinement and pre-training of a CNN architecture, specifically *GreenViT*, with a focus on accuracy and reducing incorrect predictions. Following fine-tuning, *GreenViT* demonstrates the best performance among the other models, exhibiting fewer false predictions. Furthermore, the proposed *GreenViT* performance was evaluated employing 5-fold and 10-fold cross-validation on all the included datasets. The cross-validation accuracies show that our *GreenViT* maintains a competitive performance across all folds, even though there is a slight decrease in average test accuracy when the training samples in each fold are smaller compared with the whole dataset. This consistent performance demonstrates the robustness and reliability of *GreenViT*. Tables 4 and 5 list a comprehensive overview of the 5-fold and 10-fold cross-validation accuracies for each dataset, including the average test accuracy across the 5 and 10 folds. These results reaffirm the effectiveness of our *GreenViT* in handling diverse datasets and its ability to yield consistent and promising results in real-world applications.

Table 3. Quantitative evaluation of *GreenViT* in contrast to SOTA models using the included datasets. The proposed *GreenViT* model is highlighted in blue. The upward arrow (\uparrow) depicts higher value is better.

Model	Class	PV				DRLI				PC			
		P	R	F1	ACC \uparrow	P	R	F1	ACC \uparrow	P	R	F1	ACC \uparrow
VGG19 [41]	Healthy	1.00	0.95	0.98	0.99	0.96	0.97	0.97	0.97	0.99	0.96	0.97	0.98
	Infected	0.98	1.00	0.99		0.97	0.96	0.96	0.97	0.98	1.00	0.99	
VGG16 [41]	Healthy	0.99	0.99	0.99	1.00	0.98	0.94	0.96	0.96	0.99	0.99	0.99	0.99
	Infected	1.00	1.00	1.00		0.93	0.98	0.96	0.96	0.98	1.00	1.00	0.99
EfficientNetB0 [42]	Healthy	1.00	1.00	1.00	1.00	0.99	0.83	0.90	0.89	1.00	0.97	0.98	0.99
	Infected	1.00	1.00	1.00		0.78	0.98	0.87	0.89	0.99	1.00	0.99	0.99
MobileNetV1 [43]	Healthy	1.00	0.99	0.99	1.00	0.97	0.96	0.97	0.97	0.99	0.99	0.99	0.99
	Infected	1.00	1.00	1.00		0.96	0.97	0.96	0.97	1.00	1.00	1.00	0.99
MobileNetV3Small [44]	Healthy	1.00	1.00	1.00	1.00	0.93	0.99	0.96	0.96	1.00	0.99	0.99	0.99
	Infected	1.00	1.00	1.00		0.99	0.93	0.95	0.96	0.99	1.00	1.00	0.99
ViT Base [26]	Healthy	0.92	0.98	0.95	0.95	0.81	0.62	0.70	0.75	0.87	0.95	0.91	0.94
	Infected	0.98	0.92	0.95		0.71	0.86	0.78	0.75	0.98	0.94	0.96	
<i>GreenViT</i>	Healthy	1.00	1.00	0.99	1.00	0.97	0.95	0.96	0.98	0.98	1.00	0.99	0.99
	Infected	0.99	1.00	1.00		0.98	0.99	0.98	0.98	0.99	0.98	0.99	0.99

Table 4. Five-fold cross validation accuracies of *GreenViT* for the included datasets.

Fold	Dataset		
	PV	DRLI	PC
1	0.9836	0.9314	0.9540
2	0.9749	0.9425	0.9377
3	0.9723	0.9425	0.9471
4	0.9611	0.9623	0.9632
5	0.9839	0.9447	0.9680
Average Test Accuracy	0.9752	0.9446	0.9540

Table 5. Ten-fold cross validation accuracies of *GreenViT* for the included datasets.

Fold	Dataset		
	PV	DRLI	PC
1	0.9837	0.9647	0.9665
2	0.9543	0.9736	0.9603
3	0.9795	0.9713	0.9401
4	0.9681	0.9802	0.9540
5	0.9696	0.8450	0.9552
6	0.9812	0.9669	0.9620
7	0.9791	0.9425	0.9590
8	0.9828	0.9669	0.9580
9	0.9716	0.9337	0.9666
10	0.9698	0.9004	0.9574
Average Test Accuracy	0.9740	0.9445	0.9579

Figure 2 illustrates the confusion matrix of the *GreenViT* method trained on different benchmark datasets. The dark green diagonal corresponds to TP, while the saturation indicates accurate classifications. The proposed *GreenViT* demonstrates superior overall classification accuracy compared with the SOTA models, although there are some misclassifications within both categories. The training accuracy and loss graphs are visualized in

Figure 3. The vertical axis represents accuracy and loss, while the horizontal axis represents the total number of epochs. It is evident from Figure 3 that *GreenViT* effectively detects plant diseases. As the number of training and validation iterations increases, the line graphs of training and validation accuracy change, as depicted in Figure 3a. The proposed *GreenViT* converges at seven epochs, achieving training and validation accuracies of 100%, 98% and 99% on the PV, DRLI and PC datasets, respectively. Similarly, the training and validation loss values change and decrease to 0.0 and 0.09, respectively, as depicted in Figure 3b. In addition, the suggested *GreenViT* is compared with the other pre-trained models in Table 3. The results indicate that the proposed *GreenViT* outperforms the other pre-trained models listed in Table 3.

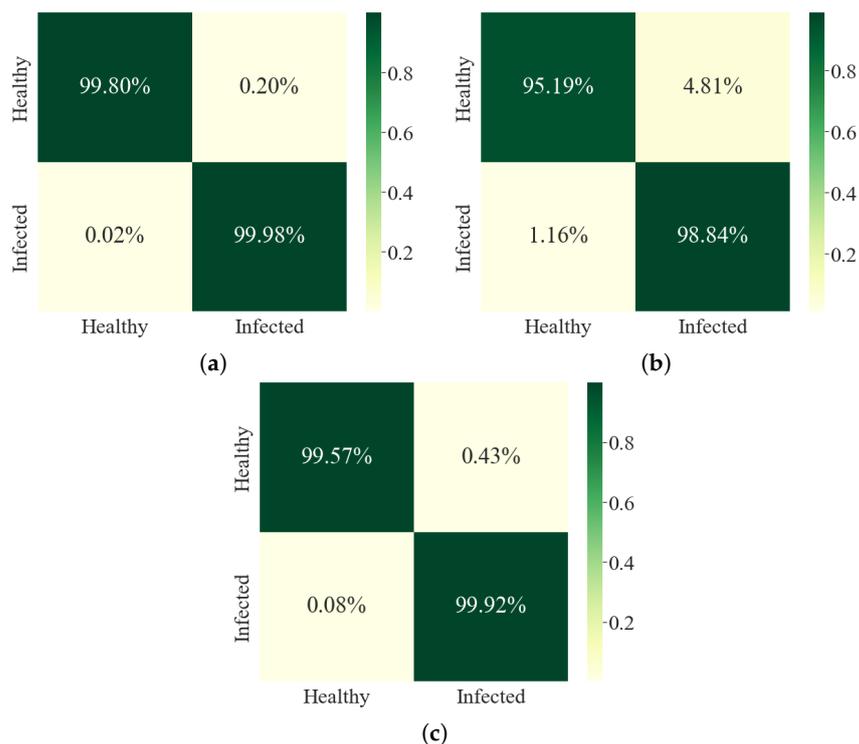


Figure 2. Confusion matrices of the proposed *GreenViT* for all the included datasets. (a) PlantVillage. (b) Data Repository of Leaf Images. (c) Plant Composite.

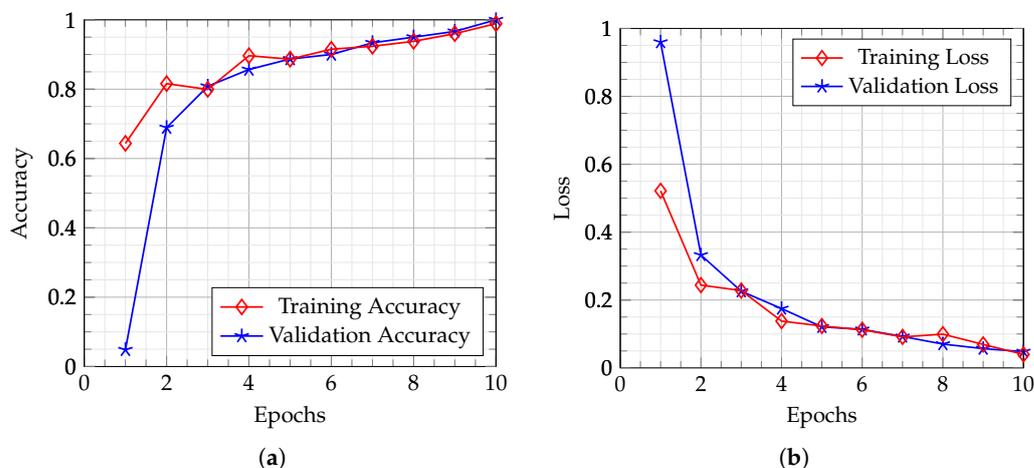


Figure 3. Training and validation accuracy and loss of the proposed *GreenViT* method on PC dataset. (a) Accuracy. (b) Loss.

3.3. Qualitative Results

We performed a visual analysis to determine the qualitative results of the proposed *GreenViT* model in distinguishing images with infection from those that are healthy plants based on class activation. The results, as shown in Figure 4, demonstrate the robustness of *GreenViT* in detecting diseased regions within a given input image. Figure 4 showcases the visual outcomes of the proposed *GreenViT* model for the samples obtained from the all three included datasets. The first row represents the input images from the PV dataset. The second row depicts the images from DRLI dataset. The third row contains images from the newly created PC dataset. All the samples are quite different from each other in type, size, geometry and color schema. The fourth row represents the ground truth (GT) labels which are the actual labels for each input image, while the last row shows the predicted label by the proposed *GreenViT* model. The infected images are highlighted in red, while the healthy samples are denoted by blue. The analysis depicted in Figure 4 offers compelling evidence of the remarkable capabilities of the proposed *GreenViT* model in accurately detecting infected and healthy regions of plant leaves. The visual representation vividly showcases the model's proficiency in this aspect.

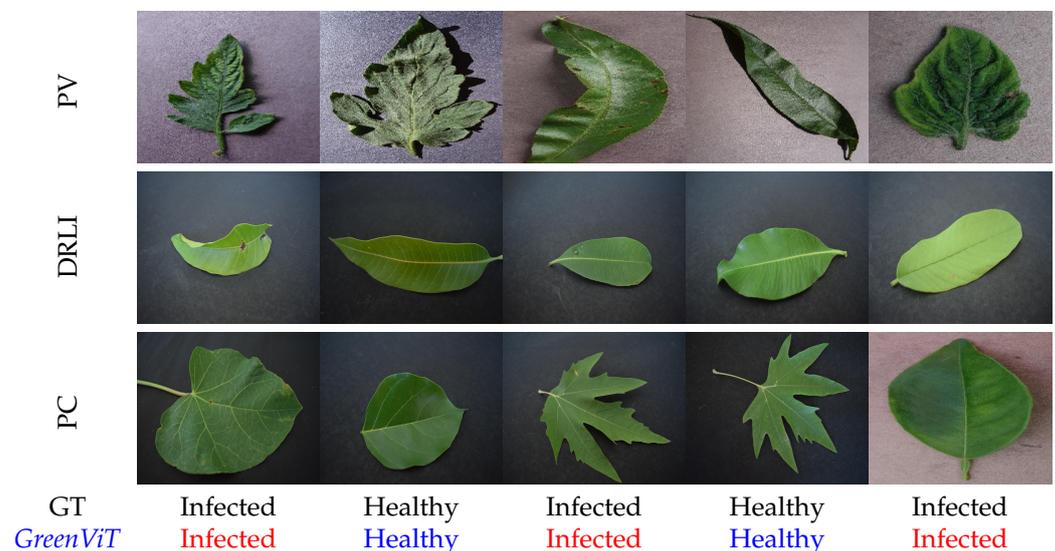


Figure 4. The objective of plant disease detection is to identify the presence of infections, but certain unique leaf images present challenges to the human eye and are not easily distinguishable without assistance. In our study, we addressed this issue by utilizing *GreenViT* to visually compare its performance on various datasets. The included figure displays a series of sample leaf images from different datasets. The first row consists of images from the PV, the second row showcases images from the DRLI and the third row represents images from the PC dataset. The second last row contains ground truth (GT) labels, where healthy samples are highlighted in **blue** text, while infected samples are indicated in **red** text. Through this visual comparison, we evaluated the effectiveness of *GreenViT* in detecting plant diseases across different datasets.

3.4. Time Complexity

In order to evaluate the effectiveness, performance and suitability for deployment of a DL model, it is crucial to conduct real-time assessments on various devices, including small edge device like the RPi 4 (Model B+), which incorporates a Central Processing Unit (CPU). The RPi 4B+ features a quad-core Cortex-A72 64-bit processor with 1.5 GHz, also comes with four GB of main memory. The specifications of the CPU analyzing the Frames Per Second (FPSs) of the proposed *GreenViT* model can be found in Section 3 of this paper. The established criterion for evaluating the model's performance in optimal applications achieves an FPS of 30 or higher, which is considered optimal for real-world scenarios according to References [38,45]. To assess the model's performance, the authors recorded a brief video of plants using a mobile phone. The FPSs obtained for the proposed

GreenViT model when utilizing the RPi 4B+ and CPU are 0.34 and 22.19, respectively. Table 6 presents a comparison of the proposed *GreenViT* model's FPS with that of several baseline models.

The experimental findings demonstrate the FPS achieved with different models, namely VGG19, VGG16, EfficientNetB0 and MobileNetV1, when employing the RPi 4B+ and CPU. For the VGG19 model, the obtained FPS values are 0.47 and 9.49. Similarly, the FPS for the VGG16 model is 0.62 and 11.09, while the EfficientNetB0 model achieves FPS values of 2.69 and 19.74. As for the MobileNetV1 and MobileNetV3Small model, the respective FPS values are 8.23, 22.96 and 7.43, 27.94. Comparing the inference speed of the ViT base variant and the proposed *GreenViT*, it becomes evident that the proposed model performs more favorably than the ViT Base and the modified *GreenViT* is a more suitable option for edge devices. This whole comparison supports the notion that the execution of the newly proposed *GreenViT* method is satisfactory. Therefore, the model exhibits a capability for real-time processing and operation.

Table 6. An assessment of the proposed *GreenViT* FPS against several other DL models. This analysis provides the relative performance of each model in terms of inference speed. The proposed *GreenViT* model is highlighted in blue. The downward arrow (↓) illustrates that smaller value is better while upward arrow (↑) depicts higher value is better.

Model	Parameters (M) ↓	Size (MB) ↓	FPS ↑	
			RPi 4B+	CPU
VGG19	200.25	229.0	0.47	9.49
VGG16	147.15	168.0	0.62	11.09
EfficientNetB0	4.05	46.9	2.69	19.74
MobileNetV1	3.23	37.1	8.23	22.96
MobileNetV3Small	1.53	18.0	7.43	27.94
Vit Base	86.00	345.0	0.21	19.83
<i>GreenViT</i>	21.65	247.0	0.34	22.19

4. Conclusions

According to the proposed study, it introduces a plant disease and infection detection method based on transformers that outperform existing SOTA studies. Additionally, to enhance the performance of the method, the proposed *GreenViT* was fine-tuned to bring down the number of parameters from 86 M to around 21.65 M. A total of three datasets, namely, the PV, DRLI and PC datasets, were employed to evaluate the proposed *GreenViT*. The study also showcases a comprehensive quantitative and qualitative analysis to prove the model generalization ability in real-world scenarios. In order to validate the efficacy and efficiency of the proposed approach, future experiments will utilize edge devices or drones that utilize a variety of leaf diseases. In the context of intelligent edge devices, the application of an attention-based model shows promise as a viable avenue of exploration that could be explored effectively.

Author Contributions: Methodology, S.P.; Writing—original draft, S.P.; Writing—review & editing, N.D.; Visualization, N.D.; Formal analysis, T.M.A.; Supervision, J.W.L.; Project administration, J.W.L.; Funding acquisition, N.S.A. and J.W.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financially supported by the Ministry of Trade, Industry and Energy (MOTIE) and Korea Institute for Advancement of Technology (KIAT) through the International Cooperative R&D program (Project No. P0016038) and was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2022-RS-2022-00156354) supervised by the IITP (Institute for Information and Communications Technology Planning and Evaluation) and by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R40), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. Link to the PV: <https://github.com/spMohanty/PlantVillage-Dataset> and DRLI: <https://data.mendeley.com/datasets/hb74ynkcn/1> (accessed on 5 July 2023).

Acknowledgments: The authors acknowledge the financial support by the Ministry of Trade, Industry and Energy (MOTIE) and Korea Institute for Advancement of Technology (KIAT) through the International Cooperative R&D program (Project No. P0016038) and the support by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2022-RS-2022-00156354) supervised by the IITP (Institute for Information and Communications Technology Planning and Evaluation) and by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R40), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. World Bank. World Bank Survey. 2021. Available online: <https://data.worldbank.org/indicator/SL.AGR.EMPL.ZS> (accessed on 5 June 2023).
2. World Food Clock. 2014. Available online: <http://worldfoodclock.com/> (accessed on 5 June 2023).
3. Thilagavathi, M.; Abirami, S. Application of image processing in diagnosing guava leaf diseases. *Int. J. Sci. Res. Manag.* **2017**, *5*, 5927–5933.
4. Gavhale, K.R.; Gawande, U.; Hajari, K.O. Unhealthy region of citrus leaf detection using image processing techniques. In Proceedings of the International Conference for Convergence for Technology-2014, Pune, India, 6–8 April 2014; pp. 1–6.
5. Padol, P.B.; Yadav, A.A. SVM classifier based grape leaf disease detection. In Proceedings of the 2016 Conference on Advances in Signal Processing (CASP), Pune, India, 9–11 June 2016; pp. 175–179.
6. Masazhar, A.N.I.; Kamal, M.M. Digital image processing technique for palm oil leaf disease detection using multiclass SVM classifier. In Proceedings of the 2017 IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA), Putrajaya, Malaysia, 28–30 November 2017; pp. 1–6.
7. Islam, M.; Dinh, A.; Wahid, K.; Bhowmik, P. Detection of potato diseases using image segmentation and multiclass support vector machine. In Proceedings of the 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), Windsor, ON, Canada, 30 April–3 May 2017; pp. 1–4.
8. Agrawal, N.; Singhai, J.; Agarwal, D.K. Grape leaf disease detection and classification using multi-class support vector machine. In Proceedings of the 2017 International Conference on Recent Innovations in Signal Processing and Embedded SYSTEMS (RISE), Bhopal, India, 27–29 October 2017; pp. 238–244.
9. Parikh, A.; Raval, M.S.; Parmar, C.; Chaudhary, S. Disease detection and severity estimation in cotton plant from unconstrained images. In Proceedings of the 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Montreal, QC, Canada, 17–19 October 2016; pp. 594–601.
10. Suresha, M.; Shreekanth, K.; Thirumalesh, B. Recognition of diseases in paddy leaves using knn classifier. In Proceedings of the 2017 2nd International Conference for Convergence in Technology (I2CT), Mumbai, India, 7–9 April 2017; pp. 663–666.
11. Vaishnnave, M.; Devi, K.S.; Srinivasan, P.; Jothi, G.A.P. Detection and classification of groundnut leaf diseases using KNN classifier. In Proceedings of the 2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN), Pondicherry, India, 29–30 March 2019; pp. 1–5.
12. Liu, H.; Lang, B. Machine learning and deep learning methods for intrusion detection systems: A survey. *Appl. Sci.* **2019**, *9*, 4396. [[CrossRef](#)]
13. Mohanty, S.P.; Hughes, D.P.; Salathé, M. Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **2016**, *7*, 1419. [[CrossRef](#)]
14. Chouhan, S.S.; Singh, U.P.; Kaul, A.; Jain, S. A data repository of leaf images: Practice towards plant conservation with plant pathology. In Proceedings of the 2019 4th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 21–22 November 2019; pp. 700–707.
15. Dhaka, V.S.; Meena, S.V.; Rani, G.; Sinwar, D.; Ijaz, M.F.; Woźniak, M. A survey of deep convolutional neural networks applied for prediction of plant leaf diseases. *Sensors* **2021**, *21*, 4749. [[CrossRef](#)]
16. Qiu, R.; Yang, C.; Moghimi, A.; Zhang, M.; Steffenson, B.J.; Hirsch, C.D. Detection of fusarium head blight in wheat using a deep neural network and color imaging. *Remote Sens.* **2019**, *11*, 2658. [[CrossRef](#)]
17. Bi, C.; Wang, J.; Duan, Y.; Fu, B.; Kang, J.R.; Shi, Y. MobileNet based apple leaf diseases identification. *Mob. Netw. Appl.* **2022**, *27*, 172–180. [[CrossRef](#)]
18. Lee, S.H.; Goëau, H.; Bonnet, P.; Joly, A. New perspectives on plant disease characterization based on deep learning. *Comput. Electron. Agric.* **2020**, *170*, 105220. [[CrossRef](#)]

19. Kundu, N.; Rani, G.; Dhaka, V.S.; Gupta, K.; Nayak, S.C.; Verma, S.; Ijaz, M.F.; Woźniak, M. IoT and interpretable machine learning based framework for disease prediction in pearl millet. *Sensors* **2021**, *21*, 5386. [[CrossRef](#)]
20. Rangarajan, A.K.; Purushothaman, R.; Ramesh, A. Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia Comput. Sci.* **2018**, *133*, 1040–1047. [[CrossRef](#)]
21. Amara, J.; Bouaziz, B.; Algergawy, A. A deep learning-based approach for banana leaf diseases classification. In *Datenbanksysteme für Business, Technologie und Web (BTW 2017)-Workshopband, Proceedings of the Workshop Big (and Small) Data in Science and Humanities (BigDS17), Stuttgart, Germany, 6–10 March 2017*; Gesellschaft für Informatik e.V.: Bonn, Germany, 2017.
22. Barbedo, J.G.A. A review on the use of unmanned aerial vehicles and imaging sensors for monitoring and assessing plant stresses. *Drones* **2019**, *3*, 40. [[CrossRef](#)]
23. Thai, H.T.; Tran-Van, N.Y.; Le, K.H. Artificial cognition for early leaf disease detection using vision transformers. In Proceedings of the 2021 International Conference on Advanced Technologies for Communications (ATC), Ho Chi Minh City, Vietnam, 14–16 October 2021; pp. 33–38.
24. Hasan, M.; Tanawala, B.; Patel, K.J. Deep learning precision farming: Tomato leaf disease detection by transfer learning. In Proceedings of the 2nd International Conference on Advanced Computing and Software Engineering (ICACSE), Sultanpur, India, 8–9 February 2019.
25. Latif, G.; Alghazo, J.; Maheswar, R.; Vijayakumar, V.; Butt, M. Deep learning based intelligence cognitive vision drone for automatic plant diseases identification and spraying. *J. Intell. Fuzzy Syst.* **2020**, *39*, 8103–8114. [[CrossRef](#)]
26. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
27. Gong, C.; Wang, D.; Li, M.; Chandra, V.; Liu, Q. Vision transformers with patch diversification. *arXiv* **2021**, arXiv:2104.12753.
28. Lu, X.; Yang, R.; Zhou, J.; Jiao, J.; Liu, F.; Liu, Y.; Su, B.; Gu, P. A hybrid model of ghost-convolution enlightened transformer for effective diagnosis of grape leaf disease and pest. *J. King Saud Univ.—Comput. Inf. Sci.* **2022**, *34*, 1755–1767. [[CrossRef](#)]
29. Thakur, P.S.; Khanna, P.; Sheorey, T.; Ojha, A. Explainable vision transformer enabled convolutional neural network for plant disease identification: PlantXViT. *arXiv* **2022**, arXiv:2207.07919.
30. Wu, H.; Xiao, B.; Codella, N.; Liu, M.; Dai, X.; Yuan, L.; Zhang, L. Cvt: Introducing convolutions to vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Online, 11–17 October 2021; pp. 22–31.
31. Yuan, K.; Guo, S.; Liu, Z.; Zhou, A.; Yu, F.; Wu, W. Incorporating convolution designs into visual transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Online, 11–17 October 2021; pp. 579–588.
32. Li, Y.; Zhang, K.; Cao, J.; Timofte, R.; Van Gool, L. Localvit: Bringing locality to vision transformers. *arXiv* **2021**, arXiv:2104.05707.
33. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Online, 11–17 October 2021; pp. 10012–10022.
34. Wang, P.; Wang, X.; Wang, F.; Lin, M.; Chang, S.; Li, H.; Jin, R. Kvt: K-rn attention for boosting vision transformers. In *Computer Vision—ECCV 2022, Proceedings of the 17th European Conference, Tel Aviv, Israel, 23–27 October 2022*; Part XXIV; Springer: Cham, Switzerland, 2022; pp. 285–302.
35. Child, R.; Gray, S.; Radford, A.; Sutskever, I. Generating long sequences with sparse transformers. *arXiv* **2019**, arXiv:1904.10509.
36. Ali, A.; Touvron, H.; Caron, M.; Bojanowski, P.; Douze, M.; Joulin, A.; Laptev, I.; Neverova, N.; Synnaeve, G.; Verbeek, J.; et al. Xcit: Cross-covariance image transformers. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 20014–20027.
37. Chu, X.; Tian, Z.; Wang, Y.; Zhang, B.; Ren, H.; Wei, X.; Xia, H.; Shen, C. Twins: Revisiting the design of spatial attention in vision transformers. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 9355–9366.
38. Dilshad, N.; Ullah, A.; Kim, J.; Seo, J. LocateUAV: Unmanned Aerial Vehicle Location Estimation via Contextual Analysis in an IoT Environment. *IEEE Internet Things J.* **2023**, *10*, 4021–4033. [[CrossRef](#)]
39. Parez, S.; Dilshad, N.; Alanazi, T.M.; Lee, J.-W. Towards Sustainable Agricultural Systems: A Lightweight Deep Learning Model for Plant Disease Detection. *Comput. Syst. Sci. Eng.* **2023**, *47*, 515–536. [[CrossRef](#)]
40. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
41. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
42. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 10–15 June 2019; pp. 6105–6114.
43. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
44. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.
45. Dilshad, N.; Khan, T.; Song, J. Efficient Deep Learning Framework for Fire Detection in Complex Surveillance Environment. *Comput. Syst. Sci. Eng.* **2023**, *46*, 749–764. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.