Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Highlights

PD-TR: End-to-end plant diseases detection using a transformer

Computers and Electronics in Agriculture xxx (xxxx) xxx

Hanxiang Wang, Tri-Hai Nguyen, Tan N. Nguyen, Minh Dang*

- A huge plant disease dataset for 6 species that contains over 123,000 images.
- An efficient transformer-based plant disease detection framework.
- Analysis of the localized disease region using the transformer's self-attention weights.
- The proposed model outperformed previous state-of-the-art object detection models.

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

PD-TR: End-to-end plant diseases detection using a transformer

Hanxiang Wang [a,b,c], Tri-Hai Nguyen [d], Tan N. Nguyen [e], Minh Dang [f,g,*]

[a] School of Computer Science, Qufu Normal University, Rizhao 276826, China
[b] Shandong Provincial Key Laboratory of Data Security and Intelligent Computing, China
[c] Rizhao-Qufu Normal University Joint Technology Transfer Center, China
[d] Faculty of Information Technology, School of Technology, Van Lang University, Ho Chi Minh City 700000, Viet Nam
[e] Department of Architectural Engineering, Sejong University, 209 Neungdong-ro, Gwangjin-gu, Seoul 05006, Republic of Korea
[f] Institute of Research and Development, Duy Tan University, Da Nang 550000, Viet Nam
[g] Faculty of Information Technology, Duy Tan University, Da Nang 550000, Viet Nam

ARTICLE INFO

Keywords:
Plant disease
Precision agriculture
Image processing
CNN
Deep learning
Machine learning

ABSTRACT

Plant diseases have a significant impact on both the quantity and quality of crop yields. Therefore, the timely detection of these diseases is crucial for facilitating early treatment. Given that most plant diseases affect leaves and fruits, visually inspecting plants for disease symptoms has become increasingly vital to minimize potential damage. In the field of precision agriculture, achieving accurate and rapid automated identification of plant diseases has become essential. To address this challenge, various computer vision and deep learning-based models have been employed. Thanks to their impressive performance, deep learning has emerged as the preferred method for plant disease detection. In this context, we present an effective plant disease identification framework based on a transformer structure, designed to capture long-range features. Additionally, we suggest enhancements such as BatchFormerV2, the layer-wise adaptive moments optimizer for batch training (LAMB), and complete intersection over union (CIoU) loss to the original model. These additions enrich the model's ability to learn fine-grained features and stabilize the training process. Importantly, the proposed model is interpretable, as demonstrated by the analysis of attention weights to make the prediction process transparent and comprehensible. The proposed model outperforms previous convolutional and vision transformer-based models. The model achieves the highest mAP value of 56.3 on the large-scale plant disease dataset used in this study.

1. Introduction

The Food and Agriculture Organization (FAO) of the United Nations predicts that the global population will exceed 9.1 billion by 2050, which requires at least a 70% increase in food production to tackle the pressing challenges of food insecurity and malnutrition (FAO, 2009). However, increasing food productivity faces numerous challenges, including the effects of climate change, limited agricultural land, and the availability of clean water sources. Additionally, plant diseases significantly impact both the quantity and quality of crops, resulting in severe economic consequences such as escalating food prices for consumers and falling farmer income. These damages can aggravate food shortages, hunger, and even starvation, particularly in underdeveloped regions with limited access to preventive measures.

Traditionally, the identification and diagnosis of plant diseases heavily relied on experienced professionals who conducted manual inspections. However, this approach has been proven to be both time-consuming and labor-intensive (Jogekar and Tiwari, 2021). Moreover, considering the vastness of crop fields, it is impractical for experts to comprehensively examine each individual plant for disease symptoms (Dang et al., 2024). Recognizing the significance of early-stage disease detection, there is a pressing need to develop automated recognition systems capable of accurately and promptly monitoring crop health. Such systems provide crucial information for farmers' decision-making. Faced with these challenges, the research community has gradually shifted its focus towards computer-aided methods in order to simplify disease detection tasks and establish practical and efficient plant disease detection systems.

In recent decades, the use of image processing and machine learning (ML) algorithms has witnessed a huge surge across various domains, including agriculture. For example, in a research conducted by Padol and Yadav (2016), leaf diseases in grape plants were successfully identified and classified using the support vector machine (SVM) algorithm. The authors performed disease segmentation using K-means clustering to

* Corresponding author at: Institute of Research and Development, Duy Tan University, Da Nang, 550000, Viet Nam
E-mail address: danglienminh@duytan.edu.vn (M. Dang).

https://doi.org/10.1016/j.compag.2024.109123
Received 11 September 2023; Received in revised form 29 May 2024; Accepted 1 June 2024

locate the affected areas, and then extracted color and texture features to train the SVM model. The experimental results demonstrated an accuracy rate of 88.89% in detecting and classifying the diseases. Das et al. focused on identifying various types of leaf diseases by extracting and combining various texture features using the Haralick algorithm to improve the disease detection performance (Das et al., 2020). Different ML models, including SVM, logistic regression, and random forest, were trained using the optimized texture features. The experimental results revealed that SVM outperformed other algorithms with an accuracy of 87.6%. However, the extraction of handcrafted features requires prior knowledge of the data, introducing bias into the process. While traditional ML approaches have shown promising plant disease detection performance on small-scale datasets, they often struggle with noise and complex backgrounds, resulting in relatively lower accuracy. Additionally, distinguishing between robust and discriminative information for effective recognition tasks becomes problematic due to the abundance of extracted features.

In recent years, deep learning (DL) architectures have been proven to be highly effective substitutes for traditional ML algorithms. Their popularity has soared due to their remarkable capabilities across various domains. Convolutional Neural Networks (CNNs) have demonstrated outstanding performance in various computer vision (CV) domains, including image classification (Li et al., 2020), segmentation (Douarre et al., 2019), and object detection (Dang et al., 2020). Motivated by state-of-the-art performances achieved by both one-stage models like Single Shot MultiBox Detector (SSD) and You Only Look Once (YOLO), as well as two-stage models like Region-based Convolutional Neural Network (R-CNN), researchers have increasingly adopted CNN-based models for the accurate identification and localization of plant diseases. For instance, Xie et al. addressed the lack of a real-time detection system for detecting grape leaf diseases, a factor detrimentally affecting grape yield (Xie et al., 2020). The authors introduced a real-time detection system based on improved deep CNN, utilizing a manually collected and processed grape leaf disease dataset. The proposed Faster DR-IACNN model, which uses Faster R-CNN and incorporates Inception-v1, Inception-ResNet-v2 modules, and SE-blocks, achieved an impressive performance of 81 mAP on the grape dataset and an inference speed of 15 frames per second (FPS). The authors suggest that the proposed framework offers a viable solution for diagnosing grape leaf diseases and has implications for detecting other plant diseases. In another work, Albattah et al. proposed a robust plant disease classification framework using a custom CenterNet model with DenseNet-77 for deep keypoints extraction (Albattah et al., 2022). Performance assessment using the PlantVillage Kaggle database confirmed the method's superiority in detecting and classifying plant diseases compared to other approaches with the highest mAP of 99 and fastest testing time of 0.21 s per image. Current DL-based one-stage and two-stage detection structures heavily depend on manually designed parameters like proposals and anchors during the training process. Furthermore, extra post-processing steps are necessary to reduce duplicate predictions (Khan et al., 2022).

Taking inspiration from the influence of Transformer in natural language processing (NLP), researchers have made several adaptations to the Transformer architecture to extend its application to computer vision (CV) tasks. Notably, the end-to-end DEtection TRansformer (DETR) has been developed for object detection (Carion et al., 2020), which eliminates the requirement for laborious hand-crafted components found in conventional one-stage and two-stage detection models. Following the introduction of DETR, various extensions such as Deformable DETR (Zhu et al., 2020), DAB-DETR (Liu et al., 2022), and DN-DETR (Li et al., 2022) have been proposed to enhance the original model's performance. Although these extensions have demonstrated exceptional capabilities in CV tasks, they still lag behind equivalently sized CNN counterparts (Khan et al., 2022). For instance, the most successful DETR-based models currently achieve less than 50 AP on COCO. Additionally, the scalability of such DETR-based models remains an area that requires further investigation. Even in the domain of plant disease identification, transformer-based architectures have displayed inferior performance when compared to CNN-based models (Thakur et al., 2021). Consequently, the recent introduction of DETR with Improved deNoising anchOr box (DINO) has addressed lingering concerns associated with DETR-related architectures (Zhang et al., 2022). DINO has demonstrated outstanding scalability, setting a new benchmark with a remarkable 63.3 AP on the COCO dataset.

Motivated by the achievements of the DINO model, this research suggests a robust transformer-based system for the identification of 12 different plant diseases. The approach involves improving the original DINO model through fine-tuning and training on a large-scale dataset consisting of over 121,466 images captured by smartphones. Furthermore, the study extracts and visualizes mean feature maps using the DINO's attention weights from the decoder's final layers, effectively facilitating the interpretation of the model's predictions.

The rest of this paper is structured as follows. Section 2 describes the plant disease dataset used in this research. Section 3 outlines the proposed automated plant disease detection framework. A detailed description of each component of the framework is discussed in Section 4. The experimental results of the introduced framework are provided in Section 5. In Section 6, we delve into the main contributions and implications arising from this study. Lastly, Section 7 concludes the study and offers potential directions for future research.

2. Plant diseases detection data set

Earlier studies on plant disease identification have been limited in their scope, often relying on small-scale datasets. As indicated in Table 1, the majority of these datasets were primarily designed for plant disease classification tasks. With the exception of the PDD271 dataset introduced by Liu et al. (2021), the remaining datasets contain fewer than 100,000 images. Despite being the most comprehensive in terms of disease classes, the current benchmark PlantVillage dataset comprises merely 54,305 images (Hughes et al., 2015).

In contrast, this study distinguishes itself by utilizing a large-scale plant disease detection dataset comprising approximately 121,466 images across 12 distinct classes of plant diseases, surpassing the diversity and quantity of most previous datasets (NIA, 2023). The dataset spans six different plant species, namely cucumber, strawberry, grape, tomato, chili pepper, and paprika. Provided for research purposes by the National Information Society Agency of Korea (NIA),[1] this comprehensive dataset significantly enhances the study's robustness and practical applicability. The data collection was a collaborative effort, primarily coordinated by the Farm Hannong Company Limited[2] in partnership with various organizations. Notable contributors included Nonghyup University[3] for tomato data collection, Yonam College[4] for paprika data collection, Gyeongsangbuk-do Agricultural Research & Extension Services[5] for grape and chili pepper data collection, and Jeonbuktechnopark[6] for data refinement and processing tasks.

2.1. Plant disease dataset collection

The dataset for plant disease detection was captured using the Canon EOS 60D, an advanced digital single-lens reflex (DSLR) camera renowned for its exceptional features. Equipped with an APS-C-sized 18-megapixel CMOS sensor, this camera strikes a perfect balance between high resolution and impressive low-light performance. Its 9-point

[1] https://www.nia.or.kr/site/nia_kor/main.do.
[2] https://www.farmhannong.com/eng/main/index.do.
[3] http://nonghyup.ac.kr/e_main.asp.
[4] http://eng.yonam.ac.kr/mbshome/mbs/eng/index.do.
[5] https://www.gba.go.kr/english/index.do.
[6] https://www.jbtp.or.kr/eng/index.jbtp.

Table 1
Detailed descriptions of some well-known plant disease datasets.

Research	Year	Task	# species	# classes	# images
Plant Pathology 2021 (FGVC8) (Thapa et al., 2020)	2021	Classification	Apple	12	23,000
PDD271 (Liu et al., 2021)	2021	Classification	42	271	220,592
PlantDoc (Singh et al., 2020)	2020	Classification	13	27	2598
SAMIR (2018)	2018	Classification	14	38	87,000
PlantVillage (Hughes et al., 2015)	2015	Classification	14	38	54,305
CropDisease dataset (This study)	2023	Detection	6	12	121,466

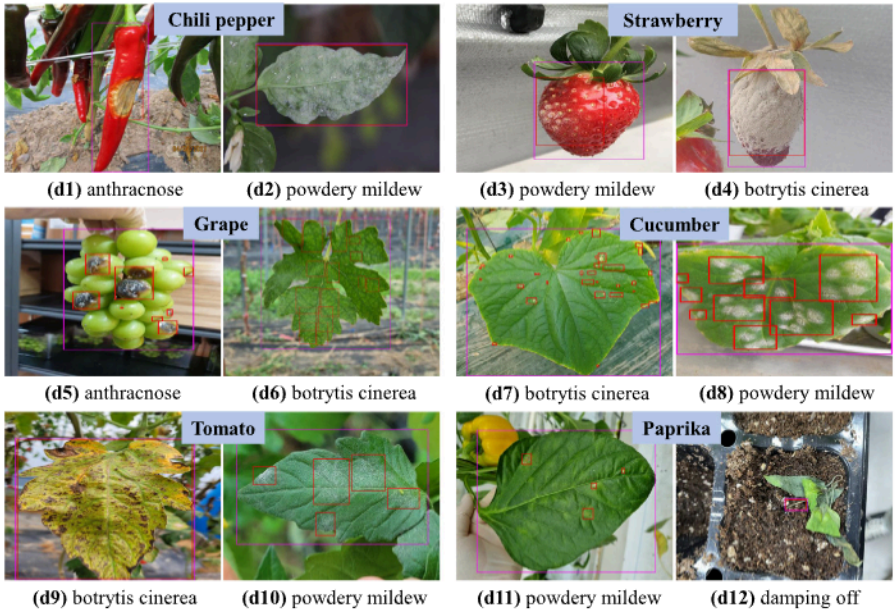

Fig. 1. Depiction of the 12 main classes of plant disease from the plant disease detection dataset. **Note:** The plant disease detection dataset illustrates the 12 primary classes of plant diseases, with the defect regions highlighted by red bounding boxes.

cross-type autofocus (AF) system ensures precise and swift focusing, particularly when utilizing the viewfinder, and the inclusion of AI Servo AF enables continuous tracking of moving subjects. Each image within the dataset boasts dimensions of 5184 by 3456 pixels, showcasing the camera's remarkable image quality. For visual reference, please refer to Fig. 1, which presents illustrative sample images for each class of the plant disease identification dataset.

Plants exhibiting symptoms of disease, such as discoloration, lesions, or other abnormalities were visually inspected in controlled environments like outdoor fields and greenhouses. Annotations were made at the level of individual lesions or affected areas. Each lesion was carefully examined, and its characteristics, such as size, shape, and color, were considered for disease identification. The annotation criteria were established based on well-defined characteristics associated with each disease is provided as follows.

- Anthracnose (d1 and d5): a fungal disease that affects a wide range of plants. It is caused by various species of fungi in the genus Colletotrichum (Soytong et al., 2005). Anthracnose commonly manifests as dark, sunken lesions on leaves, stems, fruits, or flowers. These lesions may expand and develop distinctive spore-producing structures. The disease can cause defoliation, premature fruit drop, and overall plant decline.
- Powdery mildew (d2, d3, d8, d10, and d11): a fungal disease caused by different species of the order Erysiphales (Panstruga and Schulze-Lefert, 2002). It affects a wide range of plants, including ornamentals, fruits, and vegetables. The disease appears as a white or grayish powdery coating on the surfaces of leaves, stems, flowers, and fruits. It can cause stunted growth, leaf curling, and reduced photosynthesis. Powdery mildew thrives in warm and humid conditions, spreading through airborne spores.
- Botrytis cinerea (d4, d6, d7, and d9): Botrytis cinerea, also known as gray mold, is caused by the fungus Botrytis cinerea (Williamson et al., 2007). The disease causes a grayish-brown fuzzy mold to develop on infected plant parts, including flowers, leaves, and fruits. It thrives in cool, humid conditions and can spread rapidly, leading to severe decay and loss of the affected plant tissues.
- Damping off (d12): Damping off is a common fungal disease that primarily affects young seedlings and is caused by various fungi, including species of Pythium, Rhizoctonia, and Fusarium (Lamichhane et al., 2017). It typically occurs in excessively moist or poorly drained soil. Damping off causes the seedlings to become weak, wilt, and eventually collapse at the soil level. It can lead to significant losses in nurseries and seedbeds.

The annotation criteria went beyond just the visual aspects of lesions, incorporating any surrounding symptoms or patterns that could aid in accurate disease identification. To achieve this comprehensive annotation, a team of 15 experts from Jeonbuktechnopark participated in a nine-month image labeling task. Each expert annotated around 50 images daily, ensuring thorough examination of diverse disease manifestations. This process was facilitated by a custom annotation tool developed in Python, streamlining the workflow and ensuring consistency.

Fig. 2 provides a comprehensive view of the dataset, presenting the total number of images for each plant disease type. The dataset contains a total of 121,446 annotated images. To facilitate the training, validation, and testing processes, a data split ratio of 8:1:1 was adopted. Accordingly, 123,237 images, equivalent to 80% of the original data, were randomly chosen as the training dataset. The remaining 15,405 images were allocated for validation, while another 15,405 images were set aside for testing purposes.

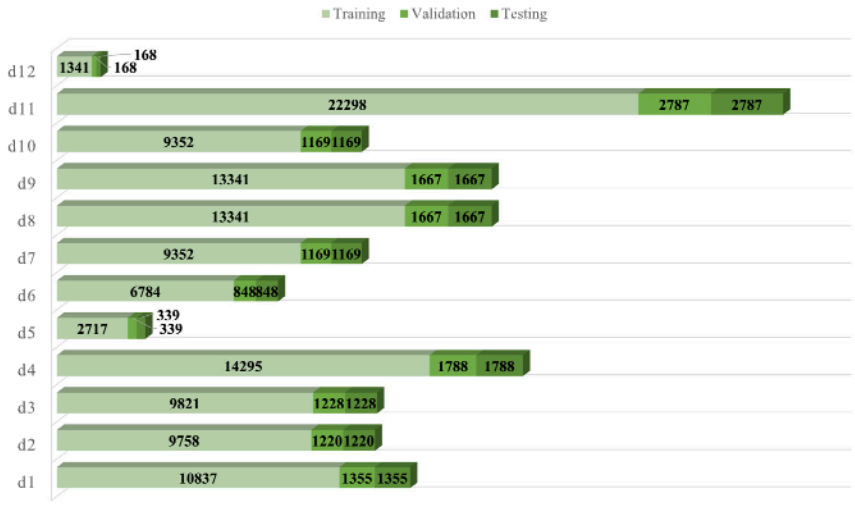

Fig. 2. A bar chart depicting the number of images for each disease class from d1 to d12.

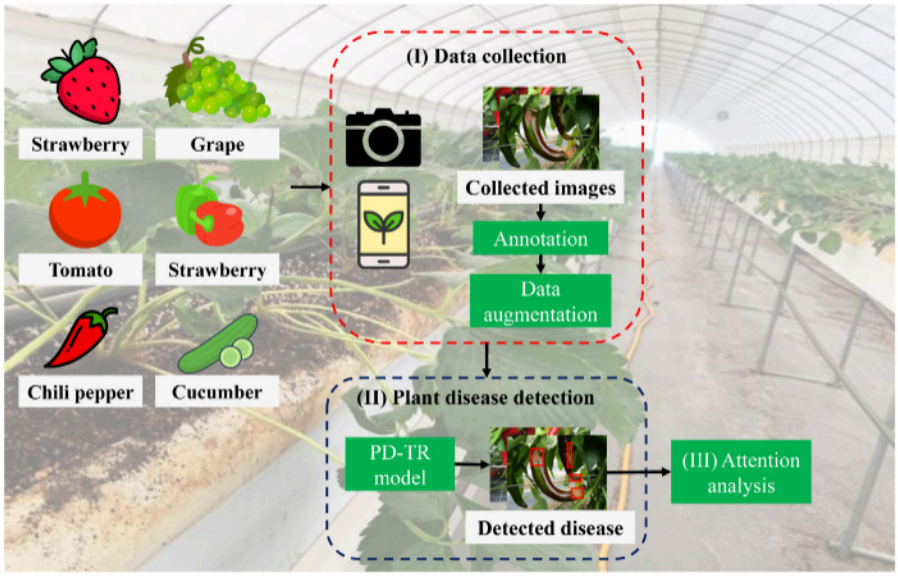

Fig. 3. Depiction of the main components of the proposed transformer-based plant disease detection framework (PD-TR).

3. System overview

Fig. 3 depicts the main processes of a comprehensive plant disease identification and analysis framework, abbreviated as PD-TR. In this context, PD signifies plant disease detection and analysis, while TR refers to the transformer-based DINO model. The three main processes are described as follows.

- Data collection: Given the inherent variability in real-life data, including factors like uneven brightness, darkness, haziness, angles, and noise, the implementation of data augmentation becomes imperative. This process involves replicating these conditions within the dataset, with the aim of increasing the number of images and ultimately enhance the detection model's robustness.
- Plant disease detection: Although existing one-stage and two-stage detection models such as RCNN (Bharati and Pramanik, 2020), YOLO (Mathew and Mahesh, 2022), and SSD (Liu et al., 2016) have demonstrated remarkable performance on well-known object detection benchmarks like COCO and Pascal VOC, they face complexity due to the requirement of learnable hyper-parameters that need to be manually initialized and optimized. Furthermore, their training process is comparatively complex when compared to the end-to-end training approach (Carion et al., 2020). In this research, we implement DINO (Zhang et al., 2022), a DL-based detection model motivated by the transformer structure, to efficiently detect various plant diseases. Notably, this model can be trained in an end-to-end manner.
- Attention weight analysis: Exploring the attention weights of the trained DINO-based plant disease identification system provides a more interpretable approach to comprehend the model's effectiveness and robustness in identifying disease regions. This information is particularly important as it enhances confidence in the model's predictions (Vaswani et al., 2017). To interpret the model's performance and gain a deeper understanding of its behavior, this study introduces the visualization and analysis of the attention weights learned by the model.

4. Methodology

4.1. Image augmentation

Within this section, diverse data augmentation techniques were employed to augment the number of images for each disease class, thereby

Fig. 4. Sample output images after implementing various data augmentation techniques.

enhancing both the size and diversity of the dataset. Initially, the Colorjitter augmentation method was implemented to address the potential impact of weather conditions on image intensity. This technique involves introducing random adjustments to the brightness, contrast, saturation, and hue of the raw images, leading to a wide array of output images. The ranges for adjusting brightness, contrast, and saturation are confined to positive values, while the hue adjustment range is restricted to values less than 0.5. In this study, the specific ranges chosen for contrast, brightness, saturation, and hue are [0, 2], [0.5, 1.5], [0.9, 1.1], and [−0.5, 0.5], respectively. Moreover, to simulate variations in camera angles and leaf orientations, rotations of 90°, 180°, and 270°, along with both vertical and horizontal symmetries, were applied. Gaussian noise was also introduced to emulate equipment-related influences. As a result, the original dataset underwent a 14-fold expansion, substantially augmenting its overall quality. Fig. 4 visually demonstrates the outcomes of the diverse image augmentation techniques applied to a sample image.

4.2. DINO model

DINO enhances the original DETR architecture by integrating various novel components (Zhang et al., 2022). It is composed of a backbone, a transformer encoder, and a decoder. The complete workflow is illustrated in Fig. 5.

Building upon the concepts introduced by DAB-DETR (Liu et al., 2022), DINO redefines each positional query in the DETR model as a 4D anchor box that gets updated automatically across decoder layers. Importantly, DINO incorporates both multi-scale features and deformable attention (Zhu et al., 2020) to further improve performance. As a result, these updated anchor boxes play a crucial role in shaping deformable attention in a sparse yet flexible manner. Similar to the principles outlined by DN-DETR (Li et al., 2022), DINO implements denoising training and advances it with contrastive denoising techniques to accelerate the convergence of training.

Moreover, DINO introduces novel training schemes including a mixed query selection strategy for initializing positional queries in the decoder and a look-forward-twice technique to enhance the optimization of box gradient back-propagation. DINO stands out for its rapid training convergence and streamlined predictions, all accomplished with a modest parameter count compared to other state-of-the-art DETR-based models.

4.2.1. Attention mechanism

The attention mechanism is a fundamental component of transformer-based structures. It grants DINO the capability to selectively concentrate on relevant areas of the input data, providing the ability to learn long-range dependencies and important contextual information. In its standard form, attention calculates a weighted summation of the input elements guided by their importance scores, which are determined by a compatibility function between a query q and a set of key–value pairs (k_i, v_i) for $i = 1, 2, \ldots, N$. This process can be mathematically expressed as follows.

$$e_i = Compatibility(q, k_i) \tag{1}$$

$$attention(Q, K, V) = \sum_{i=1}^{N} Softmax(e_i) v_{k_i} \tag{2}$$

The compatibility function calculates the importance score e_i for each key–value pair, indicating how relevant the key–value pair is to a specific query. Subsequently, the softmax function is applied to normalize the importance scores into attention weights, guaranteeing their cumulative value adds up to 1.

In CV tasks, standard attention mechanisms might encounter difficulties with complex spatial relationships and uneven data distributions. To tackle these challenges, deformable attention, also referred to as deformable self-attention, has been introduced as an extension. Originally proposed by Zhu et al. (2020), deformable attention introduces learnable offsets and deformable kernels, which allow the attention mechanism to dynamically adjust attention positions and weights based on the input context (Dang et al., 2022). This enhanced flexibility enables the model to handle spatial transformations and effectively capture fine-grained relationships between different elements in the input data. As a result, the model becomes more resilient and adaptable to varying spatial structures.

Consider an input feature map denoted as $x \in \mathbb{R}^{C \times H \times W}$, where C represents the number of channels, and H and W indicate the height and width of the map, respectively. Within this context, a query element q is defined, comprising a content feature $z_q \in \mathbb{R}^D$ and a reference point index $p_q \in \mathbb{R}^2$. The concept of deformable attention, as outlined by Zhu et al. in Zhu et al. (2020), can be expressed using the following equation:

$$\mathrm{DeformAttn}\left(z_q, p_q, x\right) = \sum_{m=1}^{M} W_m \left[\sum_{k=1}^{K} A_{mqk} \cdot W'_m x\left(p_q + \Delta p_{mqk}\right)\right] \tag{3}$$

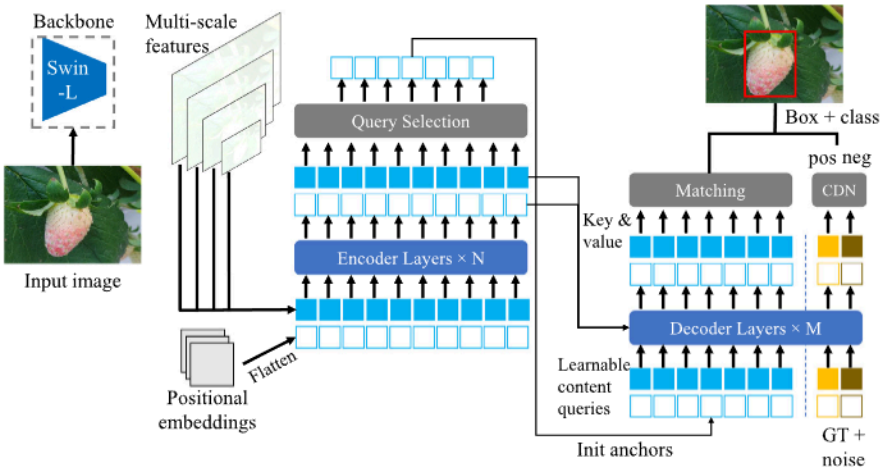

Fig. 5. Illustration of the primary parts of the PD-TR plant disease identification system.

where M denotes the total attention heads, k represents the sampled keys, and K denotes the total count of sampled keys. The parameter Δp_{mqk} signifies the sampling offset, which measures the deviation of the sampling point from the reference point p_q in the mth attention head. Furthermore, A_{mqk} indicates the attention weight of the kth sampling point in the mth attention head. It represents the importance of the sampled key in relation to the query element q. The matrices W_m and W'_m are linear transformation matrices used to project the input feature map x in order to capture different aspects of the deformable attention mechanism.

4.2.2. Contrastive deNoising training (CDT)

DETR is particularly sensitive to classifying as negative in scenarios where anchor regions contain no objects, and it often encounters difficulties in accurately identifying objects when they overlap (Khan et al., 2022). These challenges can profoundly influence the model's overall performance, and therefore become a prominent concern. DINO takes a significant step in mitigating this issue through the implementation of the Contrastive DeNoising (CDN) technique (Zhang et al., 2022).

The CDN technique involves generating both Positive and Negative Query sets, which are subsequently fed into the decoder. The Positive Query set indicates image regions containing the true ground truth bounding boxes (BB) of objects, whereas the Negative Query set characterizes background areas outside of these objects (Banerjee et al., 2023). When N objects were involved, the generation of $2 \times n$ noise queries (samples) enables efficient handling of instances where objects overlap. The positive query undergoes Generalized Intersection over Union (GIOU) loss, whereas the negative query is subjected to focal loss. Therefore, the CDN-equipped decoder assesses the presence of objects for noise sample queries while simultaneously denoising during the process of BB center coordinates prediction. The key steps involved in CDT are as follows:

- Generation of negative BB: To introduce noise, the ground truth BB is intentionally corrupted, resulting in the generation of noisy BB predictions. This process involves applying random transformations or perturbations to the anchor box parameters. These negative samples lack a meaningful connection to the actual objects and are used as dissimilar examples for contrastive learning.
- Pair creation: For each ground truth object, a pair of BB is created, containing one with positive predictions and another with negative predictions.
- Contrastive loss: The primary objective of CDT is to increase the similarity between positive BB predictions while reducing the similarity between positive predictions and their corresponding negative ones. This contrastive loss function encourages the model to effectively differentiate between positive and negative BB predictions. This CDN training process is repeated over several epochs, allowing the model to gradually become more robust to noise and enhance its accuracy in predicting BB.

4.2.3. Mixed query selection (MQS)

The Deformable DETR model implemented top-K features to enhance both positional and content queries. However, the chosen features were preliminary content features that lacked subsequent processing. This characteristic could potentially introduce ambiguity and misinterpretation into the decoder's operation (Zhang et al., 2022). In contrast, the MQS technique, employed in the DINO model, serves as a method for initializing anchor boxes as positional queries in the decoder. This approach involves handpicking initial anchor boxes from the encoder's output while maintaining the flexibility for content queries to be learned. This strategic initialization guides the initial decoder layer to prioritize the inherent spatial context in the data. As a result, the model's performance is significantly enhanced by this approach (Zhang et al., 2022).

4.2.4. Look forward twice box prediction

An iterative process for box refinement was implemented in the Deformable DETR model, but this approach posed challenges to gradient backpropagation, potentially affecting training stability (Zhu et al., 2020). This technique was termed "look forward once," as it involved updating the parameters of layer i solely through the auxiliary loss of boxes b_i.

The DINO model introduced a novel technique for box prediction called "look forward twice" (Zhang et al., 2022). This concept is based on the idea that improved box information from a subsequent layer can effectively refine box predictions from preceding layers. In this strategy, the parameters of $layer - i$ are affected by the losses of both $layer - i$ and $layer - (i + 1)$. The incorporation of the "look forward twice" technique accelerates training convergence speed and leads to substantial performance improvements. For a given input box b_{i-1} at the $(i-1)$th layer, the final predicted box $b_i^{(\text{pred})}$ is derived through the following process:

$$\Delta b_i = \text{Layer}_i\left(b_{i-1}\right), \qquad b'_i = \text{Update}\left(b_{i-1}, \Delta b_i\right),$$
$$b_i = \text{Detach}\left(b'_i\right), \qquad b_i^{(\text{pred})} = \text{Update}\left(b'_{i-1}, \Delta b_i\right),$$

where b'_i represents the undetached version of b_i. The term $\text{Update}(\cdot,\cdot)$ refers to a function responsible for refining the box b_{i-1} using the predicted box offset Δb_i. We adopt the same box updating approach as described in Deformable DETR (Zhu et al., 2020).

Table 2
Detailed description for model customization.

Model	Auxiliary	Activation	Loss	Optimization algorithm
DINO		ReLU	GIoU	Adam
PD-TR	BatchFormerV2 (Hou et al., 2022)	LeakyReLU	CIoU	LAMB

4.3. Model customization

While DINO can be trained to perform plant disease detection, its performance can be affected by certain components. To improve DINO's effectiveness in plant disease detection, various enhancements were made to the PD-TR model's architecture and optimization process, which are outlined in Table 2. These adjustments were carefully selected based on their demonstrated effectiveness in improving transformer-based models for CV tasks.

- Leaky Rectified Linear Unit (LeakyReLU): In the context of DINO, the default ReLU activation can sometimes lead to the "dying" ReLU problem, where certain neurons become inactive for specific inputs and disrupt the gradient flow during training (Lu et al., 2019). To tackle the problem, this study suggests replacing the ReLU activation with LeakyReLU activation, which introduces a small negative slope to negative inputs. This adjustment effectively mitigates the "dying" ReLU issue, resulting in a more stable training process. Incorporating LeakyReLU not only introduces additional non-linearity but also diversifies the model's outputs, thereby enhancing its overall expressive capabilities.
- BatchFormerV2: Hou et al. introduced a module known as BatchFormerV2 (BF), which, despite its simplicity, has proven highly effective in enabling transformer-based models to capture relationships among samples in each mini-batch. In the conventional approach, Transformer blocks are generally applied to pixel/patch-level feature maps (Hou et al., 2022). However, BF transformer blocks operate on feature maps with a length equivalent to the batch size. By incorporating the BF module, the PD-TR model adopts a two-stream pipeline with shared training weights. Both streams channel their outputs into the same transformer decoder. This two-stream approach ensures that all shared blocks are trained using shared weights during training. The original blocks can still function effectively without BF, avoiding any additional inference burden during testing. The integration of the BF module into various vision transformer models, including DETR (Carion et al., 2020) and Deformable-DETR (Zhu et al., 2020), consistently and significantly enhances performance, leading to improvements in the MSCOCO benchmark mAP results of over 1.3.
- Complete Intersection over Union (CIoU) loss: The Generalized IoU (GIoU) loss is an extension of the standard IoU implemented in DINO for box regression. It considers both the overlapping area between the predicted BB and the ground truth BB, as well as the non-shared areas (the union) (Rezatofighi et al., 2019). In this study, the CIoU was used in conjunction with L1 to compute box regression reconstruction loss. CIoU is an improvement over GIoU, which further incorporates additional terms to consider localization accuracy and aspect ratio consistency between the ground truth and predicted BB. This enhancement has been shown to result in faster convergence and higher detection rates compared to the GIoU loss.

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{d^2\left(p, p^{gt}\right)}{c^2} + \alpha V \tag{4}$$

where p and p^{gt} represent the central points of ground truth and predicted BB, respectively. c is the minimum enclosing box that contains both ground truth and predicted boxes. d is the Euclidean distance between the box centers, and V captures the aspect ratio consistency. The parameter α serves as a trade-off parameter in this context.

- LAMB optimizer: Although AdamW is generally regarded as the standard optimizer for various vision transformer-based models (Khan et al., 2022), Tessera et al. have demonstrated that instability in the training process can arise when the ratio of weights' L2-norm to gradients is high (Tessera et al., 2021). Therefore, this study opts for an alternative approach, utilizing the layer-wise adaptive large batch optimization (LAMB) optimizer. LAMB combines the advantages of both the Adam and Layer-wise Adaptive Rate Scaling (LARS) optimizers (You et al., 2019). Notably, LAMB employs layer-wise adaptive techniques by utilizing per-dimension normalization based on the square root of the second moment, in conjunction with layer-wise normalization. His approach is particularly well-suited for large-scale distributed training and has proven effective in training transformer models on extensive datasets.

$$\begin{aligned}
m_t &= \beta_1 m_t^{(\text{prev})} + \left(1-\beta_1\right) g_t \\
v_t &= \beta_2 v_t^{(\text{prev})} + \left(1-\beta_2\right) g_t^2 \\
m_t &= \frac{m_t}{1-\left(\beta_1\right)^t} \\
v_t &= \frac{v_t}{1-\left(\beta_2\right)^t} \\
r_t &= \frac{m_t}{\sqrt{v_t}+\epsilon} \\
x_{t+1}^{(i)} &= x_t^{(i)} - \eta_t \frac{\phi\left(\left\|x_t^{(i)}\right\|\right)}{\left\|r_t^{(i)}+\lambda x_t^{(i)}\right\|}\left(r_t^{(i)}+\lambda x_t^{(i)}\right)
\end{aligned} \tag{5}$$

where m_t is the first moment estimate at time t, while v_t corresponds to the second moment estimate at the same time step. The hyperparameters β_1 and β_2 regulate the momentum and weight decay effects, respectively. The hyperparameter λ controls the level of layer-wise adaptiveness. η_t is the learning rate vector at time t, and ϕ is the parameter vector at the same time instance. To prevent division by zero, a small constant ϵ is introduced. Additionally, r_t stands for the update ratio employed in the LAMB optimizer.

4.4. Attention weight analysis

After the completion of the training process, the averaged attention weights from the last layers of the decoder were obtained. These attention weights serve as indicators of the extent to which each attention head concentrates on different parts of the image features (Vaswani et al., 2017). Visualization of attention can be done by mapping the attention weights back to the input image. Regions with higher attention weights will be highlighted, signifying that the model is focusing on these areas during prediction. The computation of attention weights, denoted as W, follows the equation:

$$W = softmax(A/sqrt(d_k)) \tag{6}$$

where A represents the matrix of attention scores, computed through the dot product between the queries (Q) and keys (K), while d_k is the dimension of the key vectors. The computation of the attention scores matrix A can be expressed as:

$$A = Q * K^T \tag{7}$$

Each element w_i in the attention weights W represents the attention weight allocated to the ith region in the image. These weights values provide insights into how the model prioritizes different regions

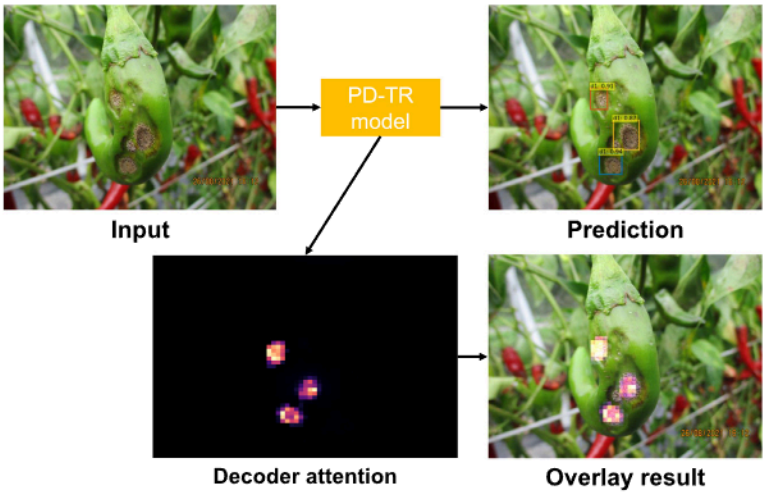

Fig. 6. Depiction of the attention analysis process. **Note:** The PD-TR's output for each prediction includes the disease type and the corresponding confidence score.

while generating predictions for a specific query region. The attention weights were extracted in the form of a square matrix with dimensions of $[H \times W, H \times W]$. To make it easier to visualize the attention weights, this matrix was reshaped into a $[H, W, H, W]$ format. This map serves as a valuable tool to assess the disease detection performance and robustness. In Fig. 6, the attention analysis for a sample input image is depicted, revealing how the PD-TR effectively focused on the disease areas through the extracted decoder attention.

4.5. Implementation description

The plant disease detection system was developed using PyTorch 1.6.0, a Python-based ML library. To guarantee consistent and fair experiments, a pre-trained ResNet-50 backbone trained on the ImageNet dataset was utilized for all the detection models employed in the experimental section. The training process was performed on two Nvidia Tesla V100 GPUs, each containing 32 GB of memory. Apart from the PD-TR model, the other DL models and their associated hyperparameters were implemented using open-source code provided by the original research papers.

The transformer-based models implemented in the experimental results section maintained a consistent configuration with 6 encoding and decoding layers. Each model contains 8 attention heads with a hidden feature dimension of 256. Specifically, the DINO and PD-TR models utilized 900 query slots, while the remaining models utilized 300 query slots. The hyperparameters and training strategies for PD-TR closely followed those of DINO, differing only in the use of cIoU instead of GIoU for BB regression loss, the incorporation of BatchFormerV2, the adoption of LeakyReLU activation instead of ReLU activation, and the utilization of the LAMB optimizer as opposed to the AdamW optimizer. The PD-TR model underwent 35 training epochs, employing the LAMB optimizer with an initial learning rate of $1e-3$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and an epsilon value of $1e-6$. A standard learning rate scheduler, as recommended by Zhang et al. (2022), was applied with a polynomial weight decay formula of $\eta_t = \eta_0 \times (1 - t/T)$.

4.6. Evaluation metrics

In this study, the plant disease detection framework's performance is assessed and compared evaluation metrics such as mean average precision (mAP), precision, and recall. These metrics rely on three fundamental values of the confusion matrix: true positive (TP), false positive (FP), and false negative (FN). Precision measures the proportion of correct positive predictions among all predicted positive instances, whereas recall is the proportion of accurate positive predictions to all actual positive instances in the dataset. These metrics can be mathematically described as follows:

$$\begin{aligned} \text{Precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}} \\ \text{Recall} &= \frac{\text{TP}}{\text{TP} + \text{FN}} \end{aligned} \tag{8}$$

The widely-used mAP is employed as an evaluation metric in object detection tasks (Zhao et al., 2019). Specifically, we utilize mAP@0.5, which evaluates the object detection accuracy at a confidence threshold of 0.50. This evaluation assesses the model's ability to identify objects by generating the precision–recall curve under the 0.50 confidence threshold and computing the average precision (AP) as the area under this curve. Ultimately, the mAP is determined by averaging the AP results across all object classes present in the dataset. This can be described as follows.

$$\text{mAP} = \frac{1}{n} \sum_{i=1}^{n} \text{AP}_i \tag{9}$$

where n is the total number of disease classes in the dataset and AP_i is the AP for the ith object class, which is computed using the precision–recall curve for that class.

5. Experimental results

5.1. Class imbalance analysis

This section investigates the effects of the imbalanced raw dataset on the performance of the proposed PD-TR model. Its performance was compared with a version of the PD-TR model trained on augmented data (Section 4.1), addressing the imbalanced distribution.

Table 3 reveals that the model trained on augmented data consistently outperforms the model trained on imbalanced data across all three metrics. The mAP increases from 54.8 to 56.3, indicating a better overall balance between precision and recall across all classes. This suggests that the model is better able to detect instances of both majority and minority classes after augmentation. This demonstrates the effectiveness of data augmentation in addressing class imbalance, enhancing model performance, and improving model generalizability.

5.2. PD-TR performance evaluation

Fig. 7 illustrates a sharp decrease in loss values for training and validation bounding box regression, reaching approximately 0.066 and 0.05, respectively, by the 10th epoch. Following this initial drop, the

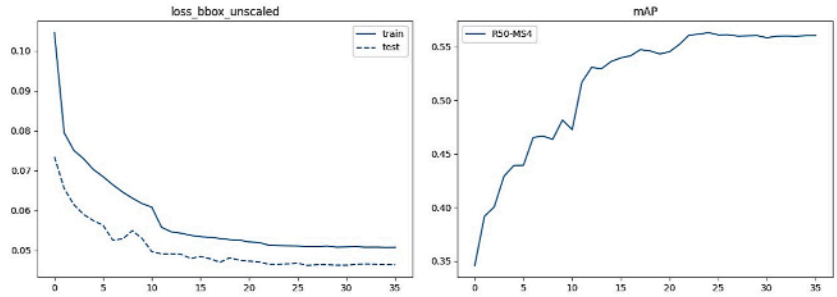

Fig. 7. Bounding box loss and mAP values visualization of the proposed PD-TR model on the plant disease detection dataset.

Table 3
Performance of the DINO model for the original imbalance data and augmented data in terms of mAP, precision, and recall.

	mAP	Precision	Recall
Raw data	54.8	0.56	0.49
Augmented data	56.3	0.59	0.53

losses continued to drop gradually and stabilized at 0.054 for training and 0.045 for validation by the end of the 35th epoch. Simultaneously, PD-TR's mAP value rapidly rose to 47 after the 10th epoch, displaying a gradual increase and eventually leveling off at 56.3.

Table 4 presents the detection performance of the PD-TR model on 12 different plant diseases, denoted as d1 through d12. The evaluation includes key metrics like mean mAP, precision, and recall.

Overall, the model demonstrates good performance across the evaluated plant diseases, with an average mAP of 56.3. The mAP scores span from 49.7 to 63.2 across different diseases, indicating the model's effectiveness in generating accurate BB predictions and label predictions. The best performance can be observed in disease d1, where the model achieves an mAP value of 63. This indicates that the model excels in accurately localizing and classifying instances of this particular disease. On the other hand, the worst performance is evident in disease d3, with an mAP of 49. This indicates that the model's predictions for this disease are comparatively less accurate in terms of BB localization and class prediction. These lower metrics for d3 signify that the model encounters challenges in accurately detecting and classifying instances of this particular disease. The poor detection performance of PD-TR for the d3 class can be attributed to various factors. The complex structure of strawberry leaves and fruits, coupled with their diverse color spectrum depending on maturity and growing conditions, can significantly challenge models in distinguishing healthy from diseased areas. This challenge is further compounded by the presence of trichomes, microscopic hair-like structures on the plant surface, which can mimic early-stage powdery mildew symptoms and lead to misidentification for the d3 class.

Precision values, ranging from 0.51 to 0.66, depict the model's ability to minimize false positive predictions, resulting in more accurate positive classifications. Similarly, Recall values, ranging from 0.50 to 0.60, demonstrate the model's ability to capture a significant portion of actual positive instances within the dataset. Moreover, the presented model obtained an average mAP value of 56.3, as depicted in Table 4. The main factor contributing to this performance is the large and challenging real-life characteristics of the plant disease dataset captured using smartphones and cameras.

5.3. Detection results and attention weight analysis

Figs. 8 and 9 provide a comprehensive view of the PD-TR model's performance across twelve distinct plant diseases, displaying both the BB predictions and their corresponding confidence scores. The model demonstrates a high detection performance in identifying each type of plant disease. Moreover, the mean deformable attention weights of the PD-TR model offer valuable insights into its ability to focus on the relevant disease regions.

Of particular note is the model's competitive performance in detecting powdery mildew (d2, d3, d8, d10, d11) and botrytis cinerea (d4, d6, d7, d9), which cause characteristic white powdery or gray spots on leaf and fruit surfaces. The PD-TR model displayed a remarkable capability in identifying multiple instances of these diseases on a single leaf or fruit, displaying high confidence in its predictions. This highlights the robustness and efficacy of the proposed model in addressing complex scenarios. The visualizations of attention weights further reinforce the model's reliability, as they clearly indicate the model's successful focus on disease-specific regions within the plants.

In conclusion, the algorithm presented in this paper accurately detects disease locations and categories, even in scenarios involving multiple diseases and small disease spots within the image. This performance is accomplished while effectively addressing concerns of false positives and missed detections, thus highlighting the model's practical significance in plant disease detection.

5.4. Ablation study

In this section, an ablation study was conducted to evaluate the influence of different modules on the performance of the DINO model in plant disease identification. The results of the experiments are explained in Table 5.

PD-TR (1) distinguishes itself from the original DINO model by replacing ReLU activation with LeakyReLU activation, resulting in an enhanced mAP of 53.6 compared to the initial 51.9. Subsequently, PD-TR (2) incorporates the CIoU loss in place of GIoU loss from PD-TR (1), further boosting the mAP to 54. Moving on, PD-TR (3) adopts the LAMB optimizer instead of the AdamW optimizer, leading to a 20% faster convergence and a shorter training process. Ultimately, PD-TR (4) introduces the BatchFormerV2 module to PD-TR (3), resulting in the highest mAP of 56.3. These modifications across network components allowed the PD-TR model to outperform the DINO model by 4.4 in terms of mAP, confirming the effectiveness of these changes in enhancing the model's detection capabilities.

5.5. Comparison with other models

This section is dedicated to evaluating the performance of the proposed PD-TR framework in contrast to other established detection networks, which include Mask-RCNN (He et al., 2017), YOLOv5 (Mathew and Mahesh, 2022), SSD (Liu et al., 2016), DETR (Carion et al., 2020), DAB-DETR (Liu et al., 2022), DN-DETR (Li et al., 2022), deformable DETR (Zhu et al., 2020), and the original DINO model (Zhang et al., 2022). The performance of these models in terms of precision, recall, mAP, and inference speed, is described in Table 6.

Among the models listed in the table, the PD-TR model presents a balanced performance across multiple aspects. It achieves the highest mAP value of 56.3 while maintaining competitive precision and recall

Table 4
Performance of the PD-TR for each class of plant disease in terms of mAP, precision, and recall.

	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	d11	d12	Average
mAP	63.2	62.4	49.7	57.1	59.2	50.5	56.1	54.3	56.6	58.4	56.2	51.6	56.3
Precision	0.66	0.65	0.51	0.6	0.62	0.52	0.58	0.58	0.59	0.62	0.59	0.55	0.59
Recall	0.6	0.57	0.53	0.55	0.54	0.57	0.53	0.52	0.54	0.56	0.51	0.5	0.53

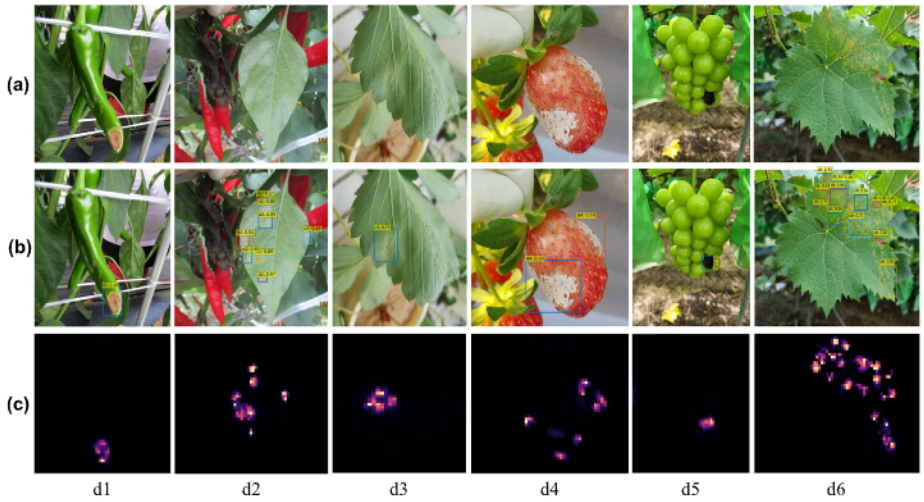

Fig. 8. The PD-TR model's predictions for each plant disease, including (a) the input image, (b) disease detection results, and (c) attention weights visualization. **Note:** The model predicted both the disease class and the model's confidence score in that prediction.

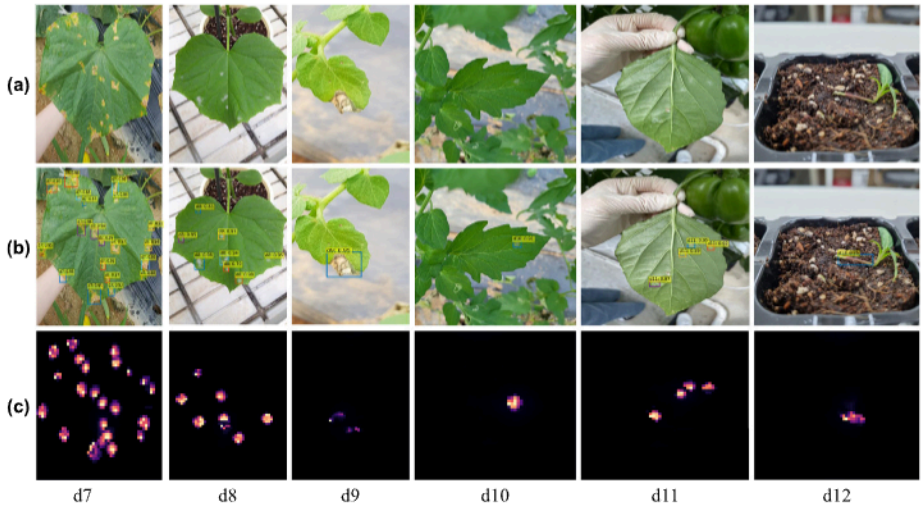

Fig. 9. The PD-TR model's predictions for each plant disease, including (a) the input image, (b) disease detection results, and (c) attention weights visualization. **Note:** The model predicted both the disease class and the model's confidence score in that prediction.

Table 5
Ablation study involving the replacement of various components within the PD-TR model.

	LeakyRELU	CIoU	LAMB	BatchFormerV2	mAP
DINO					51.9
PD-TR (1)	✓				53.6
PD-TR (2)	✓	✓			54.1
PD-TR (3)	✓	✓	✓		54.1
PD-TR (4)	✓	✓	✓	✓	56.3

values of 0.59 and 0.53, respectively, making it a reliable choice for disease detection. Additionally, its relatively fast inference speed of 23 FPS highlights its suitability for real-time applications. It is worth noting that PD-TR is based on the DINO model, which also performed well with mAP value of 51.9.

While models like YOLOv5 and Mask-RCNN exhibit competitive mAP, SSD and DN-DETR lag behind. The DN-DETR model exhibited the lowest mAP value of 49.4 among the transformer-based models, coupled with an inference speed of 24 FPS. In summary, the PD-TR model achieves the highest mAP of 56.3, suggesting its superior ability to accurately detect plant diseases across different classes, while YOLOv5 stands out with the highest inference speed of 37 FPS, making it suitable for real-time applications. The PD-TR model, along with DETR-based models, offers a commendable inference speed of around

Table 6
Comparison of the PD-TR model's performance with the other eight models using the testing plant disease data.

Model	mAP	Precision	Recall	Inference speed (FPS)
SSD (Liu et al., 2016)	48.5	0.51	0.43	30
YOLOv5 (Mathew and Mahesh, 2022)	49.3	0.52	0.48	**37**
Mask-RCNN (He et al., 2017)	50.9	0.54	0.52	20
DETR (Carion et al., 2020)	50.2	0.53	0.49	25
DAB-DETR (Liu et al., 2022)	49.8	0.52	0.51	21
Deformable DETR (Zhu et al., 2020)	50.4	0.52	0.53	22
DN-DETR (Li et al., 2022)	49.4	0.51	0.48	24
DINO (Zhang et al., 2022)	51.9	0.55	0.52	23
PD-TR (Ours)	**56.3**	0.59	0.53	23

23 FPS, indicating its potential for further inference speed improvement in the future.

6. Discussion

Our objective in this study was to identify the most efficient and robust model for automated plant disease identification framework. We examined seven distinct DL-based detection networks using a large-scale plant disease dataset and assessed their performance based on three commonly used evaluation metrics: recall, mAP, and precision. Through a series of experiments, models based on transformer structure exhibited superior performance compared to other DL detection networks when trained on a large-scale dataset. These transformer-based structures showcased a remarkable ability to capture intricate features and acquire more refined representations of disease areas. The results agree with recent studies that have also highlighted the efficiency of transformer-based structures like DAB-DETR (Liu et al., 2022), DN-DETR (Li et al., 2022), and DINO (Zhang et al., 2022). Nevertheless, it is worth noting that transformer-based models do have certain weaknesses, such as long training schedules and low inference speeds, which pose challenges when applied to real-time plant disease detection applications. Models like YOLOv5 (Mathew and Mahesh, 2022) and SSD (Liu et al., 2016) are better for scenarios where fast detection is a priority, favoring speed over absolute performance. Nonetheless, all networks examined in this study were capable of outputting predictions within a second during testing, making them suitable for plant disease detection application.

We then proposed the PD-TR model for plant disease detection based on the original DINO model. We implemented some changes to the DINO model (Section 4.3), such as using LeakyReLU activation instead of RELU activation, replacing GIoU with CIoU, incorporating BatchFormerV2, and applying the LAMB optimizer. While these changes were not specifically aimed at improving the plant disease detection topic, they were introduced to further enhance the model's performance and robustness. The ablation study in Section 5.4 revealed that these modifications enhanced the mAP value of PD-TR by 4.4 to 56.3. In particular, these adjustments were easy to deploy and led to better convergence speed and generalization ability without affecting the performance.

In the past decade, many studies on DL-based plant disease identification have demonstrated its superior performance compared to conventional ML algorithms (Xie et al., 2020; Albattah et al., 2022), yet the interpretation of these models' predictions has often been neglected. In agriculture, interpretability plays a pivotal role in enhancing farmer trust. This study concentrates on highlighting the strengths of interpretability in plant disease detection framework and underscores the interpretive potential of transformer structures, which is due to the transformer architecture. The PD-TR model could extract and visualize the extracted deformable attention weights from the decoder (Section 5.3). This distinctive attribute facilitates analysis of the model's outputs, facilitates transparency, and, importantly, contributes to fostering confidence in automated plant disease identification frameworks.

7. Conclusions and future works

This study proposes a comprehensive end-to-end automated plant disease identification framework based on the transformer architecture that can be integrated into practical disease identification applications. The model was trained on a dataset containing 121,446 images representing 12 distinct plant disease classes. Various improvements were proposed to improve the performance of the original DINO model, such as integrating data augmentation and BatchFormerV2 techniques, adopting the CIoU loss, implementing LeakyReLU activation, and using the LAMB optimizer. The outcomes demonstrate the model's ability to achieve a high detection rate and speed in accurately identifying plant diseases. Moreover, the unique attention weights feature inherent in the transformer architecture allows experts to gain a deeper understanding of how the model detects areas of interest associated with diseases.

The proposed framework was effective in detecting twelve distinct plant disease types, achieving an mAP value of 56.3. This outperforms the performance of six other state-of-the-art object detection models, as evidenced by a series of experiments. Furthermore, compared to the original DINO model, the mAP value witnessed an enhancement from 51.9 to 56.3, thanks to the introduced modifications. Notably, the transformer attention weights serve as a valuable tool for comprehending the model's decision-making process by highlighting the potential disease regions that contribute to accurate detection.

While the plant disease dataset used in this study covers twelve disease classes across six different plants, further expansion of the disease types could potentially further improve the detection capability. In addition, the integration of multispectral or hyperspectral data to capture earlier disease stages invisible to the naked eye will potentially enable more timely interventions. Given the complicated structure of transformer-based models, the current limitation in real-time detection requires attention and future research. Thus, the optimization of these models to achieve an optimized balance between robustness and computational efficiency stands as a crucial trend for future research.

CRediT authorship contribution statement

Hanxiang Wang: Writing – original draft, Methodology. **Tri-Hai Nguyen:** Visualization, Investigation. **Tan N. Nguyen:** Data curation, Conceptualization. **Minh Dang:** Writing – review & editing, Methodology.

Declaration of competing interest

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

Data availability

Data will be made available on request.

References

Albattah, W., Nawaz, M., Javed, A., Masood, M., Albahli, S., 2022. A novel deep learning method for detection and classification of plant diseases. Complex Intell. Syst. 1–18.

Banerjee, A., Biswas, S., Lladós, J., Pal, U., 2023. SwinDocSegmenter: An end-to-end unified domain adaptive transformer for document instance segmentation. arXiv preprint arXiv:2305.04609.

Bharati, P., Pramanik, A., 2020. Deep learning techniques—R-CNN to mask R-CNN: a survey. In: Computational Intelligence in Pattern Recognition: Proceedings of CIPR 2019. Springer, pp. 657–668.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., 2020. End-to-end object detection with transformers. In: European Conference on Computer Vision. Springer, pp. 213–229.

Dang, L.M., Wang, H., Li, Y., Min, K., Kwak, J.T., Lee, O.N., Park, H., Moon, H., 2020. Fusarium wilt of radish detection using RGB and near infrared images from Unmanned Aerial Vehicles. Remote Sens. 12 (17), 2863.

Dang, L.M., Wang, H., Li, Y., Nguyen, T.N., Moon, H., 2022. DefectTR: End-to-end defect detection for sewage networks using a transformer. Constr. Build. Mater. 325, 126584.

Dang, M., Wang, H., Li, Y., Nguyen, T.-H., Tightiz, L., Xuan-Mung, N., Nguyen, T.N., 2024. Computer vision for plant disease recognition: A comprehensive review. Bot. Rev. 1–61.

Das, D., Singh, M., Mohanty, S.S., Chakravarty, S., 2020. Leaf disease detection using support vector machine. In: 2020 International Conference on Communication and Signal Processing. ICCSP, IEEE, pp. 1036–1040.

Douarre, C., Crispim-Junior, C.F., Gelibert, A., Tougne, L., Rousseau, D., 2019. Novel data augmentation strategies to boost supervised segmentation of plant disease. Comput. Electron. Agric. 165, 104967.

FAO, 2009. How to feed the world in 2050. https://www.fao.org/fileadmin/templates/wsfs/docs/expert_paper/How_to_Feed_the_World_in_2050.pdf, (Accessed 15 July 2023).

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2961–2969.

Hou, Z., Yu, B., Wang, C., Zhan, Y., Tao, D., 2022. Batchformerv2: Exploring sample relationships for dense representation learning. arXiv preprint arXiv:2204.01254.

Hughes, D., Salathé, M., et al., 2015. An open access repository of images on plant health to enable the development of mobile disease diagnostics. arXiv preprint arXiv:1511.08060.

Jogekar, R.N., Tiwari, N., 2021. A review of deep learning techniques for identification and diagnosis of plant leaf disease. In: Smart Trends in Computing and Communications: Proceedings of SmartCom 2020. Springer, pp. 435–441.

Khan, S., Naseer, M., Hayat, M., Zamir, S.W., Khan, F.S., Shah, M., 2022. Transformers in vision: A survey. ACM Comput. Surv. (CSUR) 54 (10s), 1–41.

Lamichhane, J.R., Dürr, C., Schwanck, A.A., Robin, M.-H., Sarthou, J.-P., Cellier, V., Messéan, A., Aubertot, J.-N., 2017. Integrated management of damping-off diseases. A review. Agron. Sustain. Dev. 37, 1–25.

Li, Y., Wang, H., Dang, L.M., Sadeghi-Niaraki, A., Moon, H., 2020. Crop pest recognition in natural scenes using convolutional neural networks. Comput. Electron. Agric. 169, 105174.

Li, F., Zhang, H., Liu, S., Guo, J., Ni, L.M., Zhang, L., 2022. Dn-detr: Accelerate detr training by introducing query denoising. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13619–13627.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C., 2016. Ssd: Single shot multibox detector. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer, pp. 21–37.

Liu, S., Li, F., Zhang, H., Yang, X., Qi, X., Su, H., Zhu, J., Zhang, L., 2022. Dab-detr: Dynamic anchor boxes are better queries for detr. arXiv preprint arXiv:2201.12329.

Liu, X., Min, W., Mei, S., Wang, L., Jiang, S., 2021. Plant disease recognition: A large-scale benchmark dataset and a visual region and loss reweighting approach. IEEE Trans. Image Process. 30, 2003–2015.

Lu, L., Shin, Y., Su, Y., Karniadakis, G.E., 2019. Dying relu and initialization: Theory and numerical examples. arXiv preprint arXiv:1903.06733.

Mathew, M.P., Mahesh, T.Y., 2022. Leaf-based disease detection in bell pepper plant using YOLO v5. Signal Image Video Process. 1–7.

NIA, 2023. AI Hub dataset. https://www.aihub.or.kr, (Accessed 11 May 2023).

Padol, P.B., Yadav, A.A., 2016. SVM classifier based grape leaf disease detection. In: 2016 Conference on Advances in Signal Processing. CASP, IEEE, pp. 175–179.

Panstruga, R., Schulze-Lefert, P., 2002. Live and let live: insights into powdery mildew disease and resistance. Mol. Plant Pathol. 3 (6), 495–502.

Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S., 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 658–666.

SAMIR, 2018. New plant diseases. https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset, (Accessed 11 May 2023).

Singh, D., Jain, N., Jain, P., Kayal, P., Kumawat, S., Batra, N., 2020. PlantDoc: A dataset for visual plant disease detection. In: Proceedings of the 7th ACM IKDD CoDS and 25th COMAD. pp. 249–253.

Soytong, K., Srinon, W., Rattanacherdchai, K., Kanokmedhakul, S., Kanokmedhakul, K., 2005. Application of antagonistic fungi to control anthracnose disease of grape. J. Agric. Technol. 1, 33–41.

Tessera, K.-a., Hooker, S., Rosman, B., 2021. Keep the gradients flowing: Using gradient flow to study sparse network optimization. arXiv preprint arXiv:2102.01670.

Thakur, P.S., Khanna, P., Sheorey, T., Ojha, A., 2021. Vision transformer for plant disease detection: PlantViT. In: International Conference on Computer Vision and Image Processing. Springer, pp. 501–511.

Thapa, R., Zhang, K., Snavely, N., Belongie, S., Khan, A., 2020. The Plant Pathology Challenge 2020 data set to classify foliar disease of apples. Appl. Plant Sci. 8 (9), e11390.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Adv. Neural Inf. Process. Syst. 30.

Williamson, B., Tudzynski, B., Tudzynski, P., Van Kan, J.A., 2007. Botrytis cinerea: the cause of grey mould disease. Mol. Plant Pathol. 8 (5), 561–580.

Xie, X., Ma, Y., Liu, B., He, J., Li, S., Wang, H., 2020. A deep-learning-based real-time detector for grape leaf diseases using improved convolutional neural networks. Front. Plant Sci. 11, 751.

You, Y., Li, J., Reddi, S., Hseu, J., Kumar, S., Bhojanapalli, S., Song, X., Demmel, J., Keutzer, K., Hsieh, C.-J., 2019. Large batch optimization for deep learning: Training bert in 76 minutes. arXiv preprint arXiv:1904.00962.

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.-Y., 2022. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. arXiv preprint arXiv:2203.03605.

Zhao, Z.-Q., Zheng, P., Xu, S.-t., Wu, X., 2019. Object detection with deep learning: A review. IEEE Trans. Neural Netw. Learn. Syst. 30 (11), 3212–3232.

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J., 2020. Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159.