

Article

Comparative Evaluation of YOLO- and Transformer-Based Models for Photovoltaic Fault Detection Using Thermal Imagery

Mahdi Shamisavi , Isaac Segovia Ramirez *  and Carlos Quiterio Gómez Muñoz 

HCTLab Research Group, Electronics and Communications Technology Department, Universidad Autónoma de Madrid, 281049 Madrid, Spain; mahdi.shamisavi@uam.es (M.S.); carlosq.gomez@uam.es (C.Q.G.M.)

* Correspondence: isaac.segovia@uam.es

Abstract

Photovoltaic systems represent one of the most reliable and widely used technologies for electricity generation from renewable energy sources, although their performance is affected by the occurrence of faults and defects that lead to energy losses and efficiency reduction. Therefore, detecting and localizing defects in photovoltaic panels is essential. A wide variety of image analysis techniques based on aerial thermal imagery acquired by drones have been widely implemented for proper maintenance operations, requiring a comprehensive comparison among these approaches to assess their relative performance and suitability for different scenarios. This study presents a comparative evaluation of several vision-based approaches using artificial intelligence for photovoltaic defect detection. YOLO- and Transformer-based models are analyzed and benchmarked in terms of accuracy, inference time, per-class performance, and sensitivity to object size. Experimental results demonstrate that both YOLO- and Transformer-based models are computationally lightweight and suitable for real-time implementation. However, Transformer-based architectures exhibit higher detection accuracy and stronger generalization capabilities, while YOLOv5 achieves superior inference speed. The RF-DETR-Small model provides the best balance between accuracy, computational efficiency, and robustness across different defect types and object scales. These findings highlight the potential of Transformer-based vision models as a highly effective alternative for real-time, on-site photovoltaic fault detection and predictive maintenance applications.

Keywords: photovoltaic panels; vision artificial intelligence; YOLO; transformers; thermal image; photovoltaic defects



Academic Editor: Alan Brent

Received: 3 November 2025

Revised: 10 December 2025

Accepted: 26 December 2025

Published: 5 February 2026

Corrected: 25 June 2026

Copyright: © 2026 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Nowadays, generating power from renewable resources has become a fundamental focus in research and industry due to the importance and critical role of electricity. Solar photovoltaic (PV) panels are among the most widely used devices for electricity generation, accounting for approximately three-quarters of all renewable energy generation [1]. Several factors, such as environmental conditions and technical deterioration, can lead to a reduction in their efficiency and overall productivity [2]. PV faults can be categorized into five main groups: structural defects, electrical defects, thermal defects, overlying defects, and degradation defects. Table 1 provides an overview and brief explanation of each defect type. These defects and faults can lead to a decrease in electricity generation and, in severe cases, may even cause fire incidents due to hotspot faults. Therefore, it is crucial to focus on maintenance as well as on detecting and localizing faults that may occur during

manufacturing, transportation, installation, or operation to improve conversion efficiency and reduce production costs [3]. These concerns have motivated researchers in this field to develop new approaches and solutions for detecting, localizing, and addressing these issues effectively, or propose a controller that can manage switching between different electrical power sources when PV faults lead to reduced output [4].

Table 1. Different types of PV defects, their causes, effects, and corresponding subcategories.

Defect Types	Description	Subcategories
Structural defects	These defects are common and originate both during manufacturing and the subsequent operation of the solar panels. Causes include mechanical stress during assembly, transportation, and handling, as well as temperature changes and exposure to external agents.	• Microcracks and major cracks • Delamination • Surface damage • Bubbles and deformations
Electrical defects	These faults affect the performance of the PV module and are detectable mainly by electroluminescence and thermal techniques. These faults are usually caused by the presence of stray currents in ungrounded PV systems, interconnection breakage at the string, and poor soldering and thermo-mechanical stress [5,6].	• Potential induced degradation • Busbar interconnection failure
Thermal defects	These defects generate temperature anomalies and are detected with thermographic cameras. Module shading, mismatched cells, diode failure, cell cracks, failed or resistive soldering connections, damaged packaging, etc., lead to the occurrence of hotspots [7].	• Hotspots
Overlying defects	These faults are usually caused by the shadow of clouds, solid things like tall buildings, trees, poles, sand, and dust storms. The overlaid PV cell heats up and leads to a hotspot [6,8]. Losses in energy production can reach 15–20% due to this type of defect [9].	• Partial shading • Dust and dirt accumulation
Degradation defects	These defects usually develop over time due to prolonged exposure to adverse environmental conditions such as humidity, UV radiation, and extreme temperature changes. Corrosion degradation affects the junctions between cells and structural components, which can result in electrical and mechanical failures [5,10,11].	• Yellowing or discoloration • Light-induced degradation

PV fault detection methods can generally be categorized into three main groups: electrical parameter measurement, image processing approaches, and artificial intelligence (AI). The first group detects faults based on changes in electrical characteristics, while image processing approaches identify faults through visual analysis. AI and deep learning techniques automatically detect and classify faults using trained models [12]. Another important benefit of this technique is that it falls under Non-Destructive Testing approaches, which help improve product quality, ensure public safety, and prevent further fault development [13]. Condition monitoring equipment, such as cameras or sensors, embedded in Unmanned Aerial Vehicles (UAVs), is widely implemented due to their capability to provide spatially resolved information, which is required for accurate fault localization, advanced diagnostics, and maintenance planning in large-scale PV installations.

AI and deep learning approaches are used to analyze large data volumes, reducing analytic periods and enhancing the accuracy of the results. With recent advancements in GPU technology, it has become feasible to run deep learning models in real time, making visual-based fault detection more practical and efficient. The continuous growth in terms of size and complexity of AI models requires greater computational resources. It is essential to develop models that can run efficiently on edge devices with limited resources, while

remaining reliable for long-term deployment, especially as the size of PV plants continues to grow. Another main challenge in this field is based on the limited availability of labeled datasets for training, and capturing and labeling images of PV panels based on various fault types is both time-consuming and labor-intensive.

In recent years, many studies have focused on detecting defects in PV systems using vision-based AI models and different deep learning algorithms. One of the most widely implemented architectures is Convolutional Neural Networks (CNNs), which can automatically extract features at different levels, making it easier to classify and localize faults using latent features [14]. With the emergence of CNNs in deep learning, the use of these methods for fault detection has grown rapidly, with You Only Look Once (YOLO) being one of the most growing networks in the current state of the art. A new YOLO architecture called ST-YOLO was proposed in [15], which uses an attention mechanism. This model was trained on images with a resolution of 640×512 and achieved a Mean Average Precision (mAP)@0.5 of 96.6%. The proposed architecture was developed based on YOLOv8, and while its model size was reduced by about 15%, it still achieved higher accuracy. Another study [16] proposed a new loss function that combines Efficient Intersection over Union (EIOU) and Focal-F1 to handle dataset imbalance. The authors also introduced a cascade detection network to improve performance on both small- and large-scale defects. Their approach increased the mAP of YOLOv5s from 0.793 to 0.865 and YOLOv5x from 0.802 to 0.867, respectively. Lei et al. [17] proposed a hybrid Deeplab–YOLO framework for detecting hotspots in PV panels. In this approach, MobileNetV2 (a lightweight CNN architecture) was integrated into the Deeplabv3+ model to perform efficient PV panel segmentation, while a lightweight MobileNetV3 backbone replaced the standard structure of YOLOv5. They also added a small defect prediction head and used the EIOU loss function to improve both accuracy and speed. As a result, the model achieved a 2.61% improvement in panel segmentation accuracy and a 0.7% increase in hotspot detection accuracy. Another study [18] introduced a modified version of the YOLOv5 model by adding an attention mechanism, a bidirectional feature pyramid network, GhostConv layers, and the Gaussian Error Linear Unit activation function to boost both accuracy and inference speed. This improved model achieved a 2% increase in accuracy and a 20.3% improvement in speed compared to the original YOLOv5s. A review study [19] compared the performance of several models, including Support Vector Machine, Faster R-CNN, YOLOv5, YOLOv9, YOLOv10, and YOLOv11, for automatic defect detection in solar panels. Among these models, YOLOv11 achieved the highest accuracy, with a mAP of 92.7%, outperforming the others.

With the rise of Transformer models, they have also been applied to PV defect detection because of their strong performance in object detection tasks. Wang et al. [20] introduced PDFormer, a Transformer-based framework designed specifically for photovoltaic defect detection. PDFormer improves Vision Transformers (ViTs) through three main strategies: data-level augmentation, structural enhancements, and supervised learning techniques. It uses a method called QuadMix to mix positive and negative samples, including a new attention-based module for feature reweighting, and applies multi-teacher knowledge distillation using both ViT and CNN models. Experimental results showed that PDFormer reached an accuracy of 98.08%, outperforming other existing methods on the photovoltaic dataset. Ramadan et al. [21] introduced a complete system for PV fault detection that included capturing UAV images, enhancing them using an unsharp mask filter, applying data augmentation, and finally using a Vision ViT model to classify the images into anomaly and non-anomaly classes. As shown in Figure 1, the proposed model in this research outperformed other existing models in terms of accuracy.

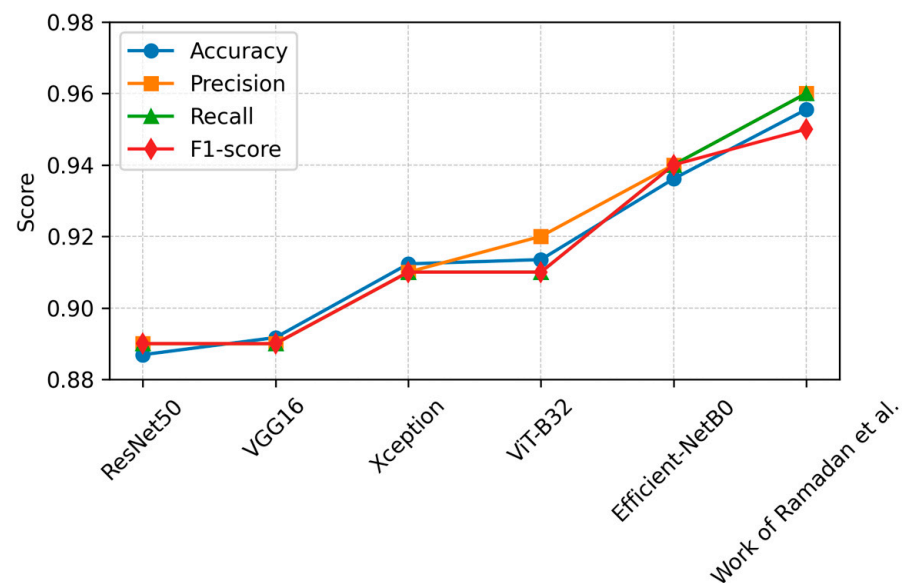


Figure 1. Comparison results of the proposed system with other well-known models. Adapted from [21].

Another study introduced Detecting defects of Photovoltaic solar cells with Image Transformers (DPiT), a Transformer-based network for detecting defects in PV panels [22]. DPiT combines convolution layers for better positional and spatial understanding, a cross-window multi-head self-attention module for stronger window relation modeling, and a multi-scale aggregation block to merge low- and high-level features, to improve its performance. Experiments on the ELPV dataset showed that DPiT outperforms the Swin Transformer, reaching 91.7%, which is the highest accuracy level achieved, with only a small increase in computational costs. Figure 2 illustrates that the proposed model in this study outperforms the other architectures.

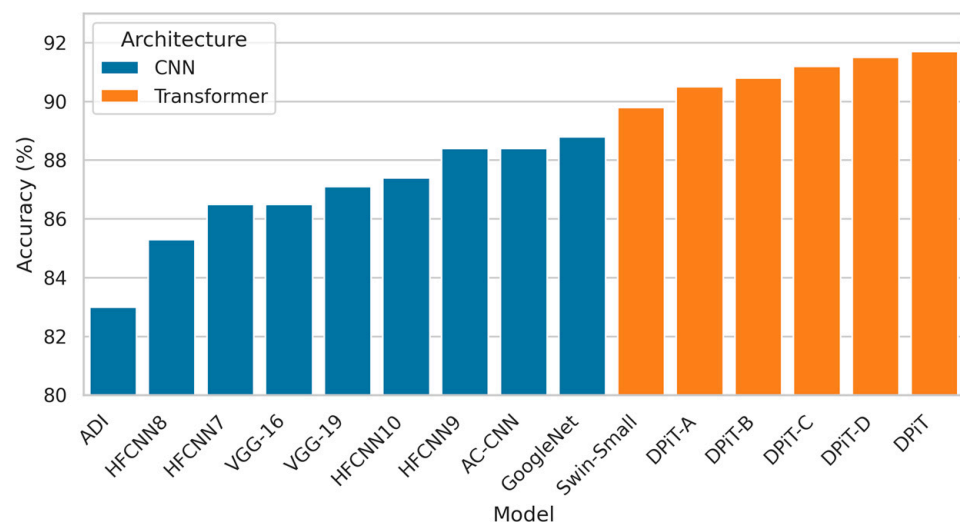


Figure 2. DPiT model against another CNN and Transformer models. Adapted from [22].

Another study [23] presents a lightweight decoder-only DETR (LD-DETR) model designed for PV defect detection. LD-DETR removes the Transformer encoder and instead uses a lightweight convolution module with an up-sampling layer to preprocess features for the decoder. This modification reduces inference time and makes the model more suitable for real-time tasks. Figure 3 shows the performance comparison between LD-DETR and other state-of-the-art models.

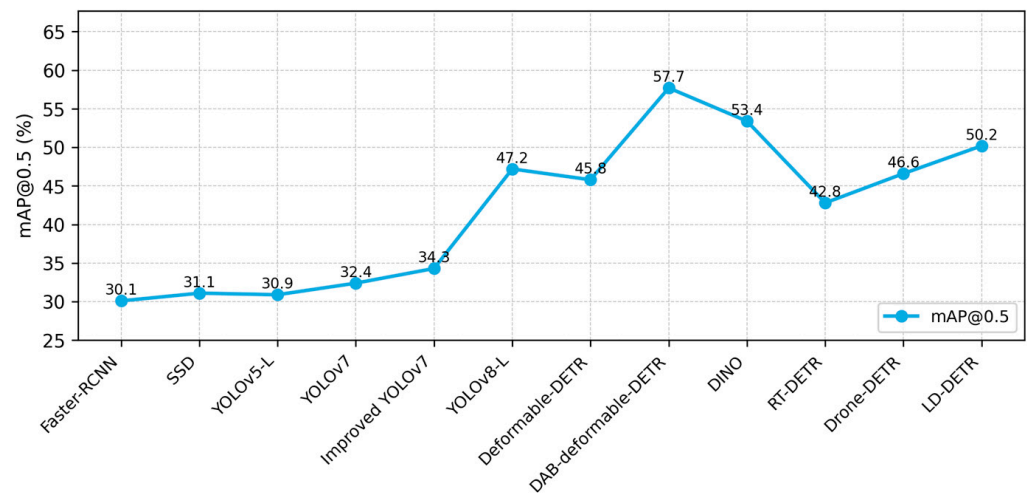


Figure 3. Comparison of mAP@0.5 results between LD-DETR and other state-of-the-art models. Adapted from [23].

Another study compared the latest versions of YOLO models, from YOLOv9 to YOLOv10, using the PVEL-AD electroluminescence dataset [24]. This work combined YOLO's built-in augmentation with CycleGAN-based generative data augmentation to address dataset imbalance. Their findings showed that YOLOv11s outperformed other variants, achieving an mAP@0.5 of 84.12%.

The main contributions to this article can be summarized as follows:

- A comprehensive benchmarking of state-of-the-art real-time object detection models is developed in this work. Architectures from both the YOLO and Transformer families are trained and evaluated on the PV defect dataset using a unified training protocol and ensuring consistency across experiments. Model performance is compared in terms of mAP@0.5, mAP@0.5:0.95, precision, recall, F1-score, and inference time.
- A per-class evaluation supported by mAP@0.5 heatmaps is conducted to identify the most and least detectable fault categories. The analysis of detection accuracy across object size groups (small, medium, and large) provides insights into the strengths and limitations of each model regarding defect scale. Evaluating the best model on unseen data from another PV plant helps verify its functionality and shows whether it can be used effectively in real-world applications.

This paper is divided into five sections: Section 2 presents the materials and methods, where YOLO and Transformer architectures are explored and discussed. Section 3 describes the experiments and results, including the dataset details and comparison analyses. Section 4 provides discussions and interpretations of the findings, and finally, Section 5 concludes the article.

2. Materials and Methods

In recent years, various models and architectures have been proposed for real-time object detection with low inference time. Two of the most well-known architectures are analyzed in this study: YOLO and Transformers.

2.1. YOLO

YOLO detects both the class and the bounding box of objects in a single stage [25]. YOLO uses the power of CNNs to extract features at different depths, allowing it to localize and classify all objects in an image with just one observation [26]. In the YOLO algorithm, the input image is first divided into an $S \times S$ grid. For each grid cell, the model predicts B bounding boxes along with class and confidence information. Each grid cell is

responsible for detecting objects whose center falls inside it. Each predicted bounding box includes the following parameters: P_c is the confidence of there being an object. B_x , B_y , B_w , and B_h represent the coordinates of the center of the bounding box and the width and height of it, respectively. The output of the YOLO architecture is a tensor with the size of $S \times S \times (B \times 5 + n)$, where n represents the number of classes in the dataset, in which C_1 , C_2 , $\dots$ represent the class names. It is notable that the class information n is shared in each grid cell, so it is not multiplied by B , and all bounding boxes in a grid cell have the same class information. Figure 4 illustrates the image divided into 9×9 grids, and 3 bounding boxes are predicted for each grid.

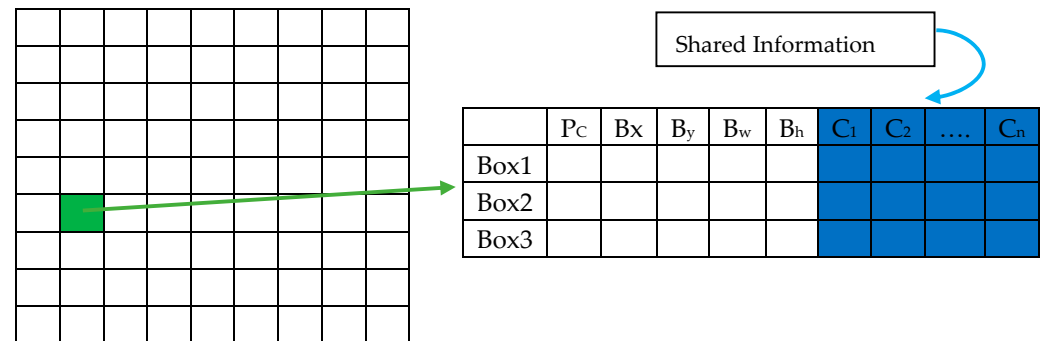


Figure 4. The image is divided into 9×9 grids, and each grid includes information for 3 bounding boxes ($B = 3$). The blue part represents the shared class information.

Due to the variety in object sizes, YOLO uses anchor boxes, which are predefined bounding boxes obtained from the ground-truth object boxes. They help the model detect objects more accurately because the prediction starts based on these predefined anchor boxes. After predicting B bounding boxes for each cell, the confidence score is calculated by multiplying the confidence score by the Intersection over Union (IoU) between the predicted box and the ground-truth box. In the next step, the class score is calculated, showing the confidence of the object belonging to a specific class and how well it matches the corresponding ground-truth box. The predicted bounding boxes are compared to a predefined threshold, and those below the threshold are discarded. In the final step, the Non-Maximum Suppression algorithm is applied to select the best-fitting bounding box, remove the others, and reduce the number of overlapping bounding boxes.

2.2. Transformer

Transformers were first used in natural language processing [27], and due to their good performance, they were later applied to vision tasks. The main component of a Transformer is self-attention, which allows the model to focus more on specific parts of the input compared to others. The Transformer includes two main parts: the encoder and the decoder. The encoder takes the inputs and generates their encodings, and then the generated representation, along with the previous output of the decoder, is fed into the decoder to generate a new output [27].

The self-attention module in the Transformer receives three vectors, namely query, key, and value, with d_k being the dimension, which are generated from the input by multiplying it with the corresponding weight matrices. The output of the module is then produced as a weighted sum of the values, where the weights are generated by a compatibility function between the query and the corresponding key, as shown in the equation below.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (1)$$

Due to the success of Transformers in natural language processing tasks, they were also used in vision tasks. Transformers are implemented in two different ways: one combined with CNNs and another alone, which is called a pure Transformer [28]. At the first step, the input image is divided into N patches, each with a resolution of $p \times p$. Since the Transformer expects a vector of size D , a trainable linear projection is used to flatten each input patch into a patch embedding. Then, a learnable embedding is applied to the sequence of patch embeddings, which serves as the image representation at the output of the encoder. On top of that, a 1D positional embedding is added to the patch embeddings to maintain positional information. ViT uses only an encoder, and its output is fed to an MLP that makes the final prediction.

Transformers are also used in object detection tasks, both in the detection and backbone parts. The pioneer model in this area is Detection Transformer (DETR). DETR uses a conventional CNN to extract features from input images, and then flattens these features and adds positional encoding before passing them to the encoder. After that, the decoder receives a small fixed number of learnable positional embeddings along with the output of the encoder. Finally, the output of the decoder is passed to a feed-forward network to detect objects and predict bounding boxes [29].

3. Experiments and Results

In this section, different models were trained and evaluated using two of the most widely implemented architectures: YOLO and Transformer. The most used versions of these algorithms, according to the state-of-the-art models, are trained on the same dataset for 100 epochs using an NVIDIA A30-1-6C MIG 1g.6 GB GPU. The best fine-tuned model from the training process was then used to validate performance on the validation set.

The Stochastic gradient descent (SGD) optimizer, with an initial learning rate of 0.01, and a Cosine Decay scheduler were employed in training the YOLO models. The batch size was fixed at 8 throughout the training process. In training the RT-DETR and RT-DETRv2 models, the AdamW optimizer was employed with a weight decay of 0.0001 and an initial learning rate of 0.0001, while a MultiStepLR learning-rate scheduler was used to control the learning-rate decay. The batch size was configured as 2 for RT-DETR and 1 for RT-DETRv2 to ensure stable training under the associated memory constraints. For the RF-DETR-Nano and -Small variants, training was performed using the AdamW optimizer with a learning rate of 0.0001. The batch sizes were set to 2 for the Nano model and 1 for the Small model. The data split was kept identical for all models, where about 10% of the dataset was reserved for validation. This validation subset remained the same across all training procedures. Additionally, the random seed was fixed to 0 to ensure reproducibility of the experiments.

The models are evaluated using various metrics to assess their detection accuracy and their ability to predict ground-truth instances. Additionally, inference times of each model are compared to define their runtime efficiency.

3.1. Dataset

The Thermal Solar PV Anomaly Detection Dataset [30] is used for experiments, including both training and validation. This dataset contains eight classes representing different types of faults in solar panels: MultiByPassed (MBP), MultiDiode (MD), MultiHotSpot (MHS), SingleByPassed (SBP), SingleDiode (SD), SingleHotSpot (SHS), StringOpenCircuit (SOC), and StringReversedPolarity (SRP). It is worth noting that the dataset includes 7500 grayscale images with a resolution of 640×640 pixels, generated using various data augmentation techniques. Figure 5 provides an overview of the number of images, number

of instances, and the mean bounding box area for each class; it also illustrates that an imbalance exists in this dataset, and its effect is evaluated in the following section.

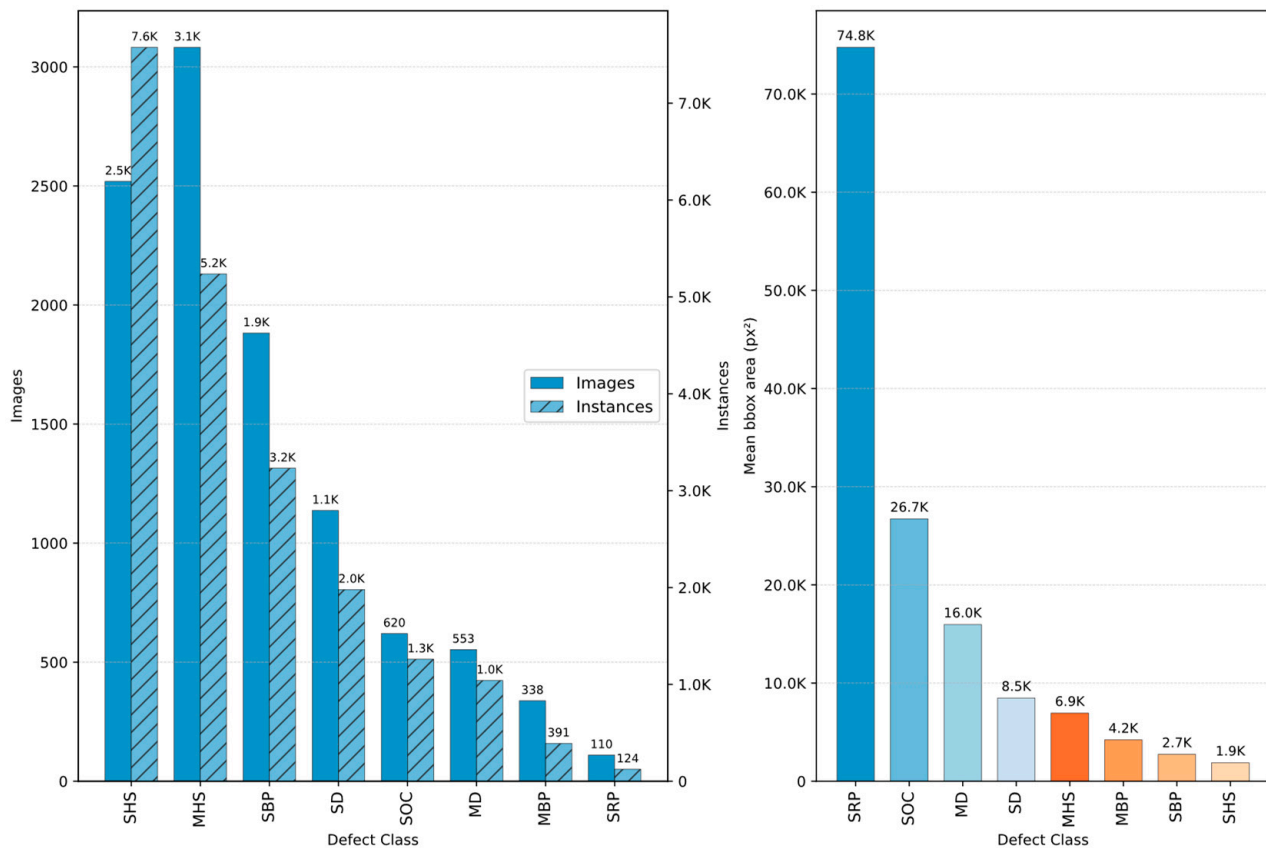


Figure 5. Dataset information: The left plot shows the number of images and instances per class, and the right plot shows the average bounding box area (in pixels $\times$ pixels) per class.

The MHS and SHS classes have the highest number of images and instances, respectively, while the SRP class has the lowest. In terms of area, samples from the SRP class have the largest bounding boxes, whereas those from the SHS class have the smallest.

3.2. Experimental Results

Several quantitative metrics are implemented to evaluate the performance of the proposed object detection models. The mAP is one of the most widely used and effective metrics, which provides a good indication of both precision and recall. Precision quantifies the proportion of correctly predicted bounding boxes, while recall measures the proportion of ground-truth boxes that are successfully detected by the model. These two metrics are independent; if a model predicts boxes with low confidence, it may detect more bounding boxes, which increases recall, but some of these predictions may be false positives, reducing precision.

The metric mAP uses the precision–recall curve and calculates the area under the curve (AUC) as the Average Precision (AP) for each class to consider both precision and recall. The mAP values across all classes are then taken as the mAP. It is also important to note that mAP@0.5 uses a threshold of 0.5 for the IoU to determine whether a predicted bounding box is considered a true positive. On the other hand, mAP@0.5:0.95 calculates mAP using multiple IoU thresholds ranging from 0.5 to 0.95 and then computes the mean across all these thresholds. The mean inference time for a set number of images is calculated to evaluate the inference time of the models. Figure 6 presents the mAP@0.5, mAP@0.5:0.95,

and inference time for state-of-the-art models in the YOLO-Small model and Transformer architectures [31–41].

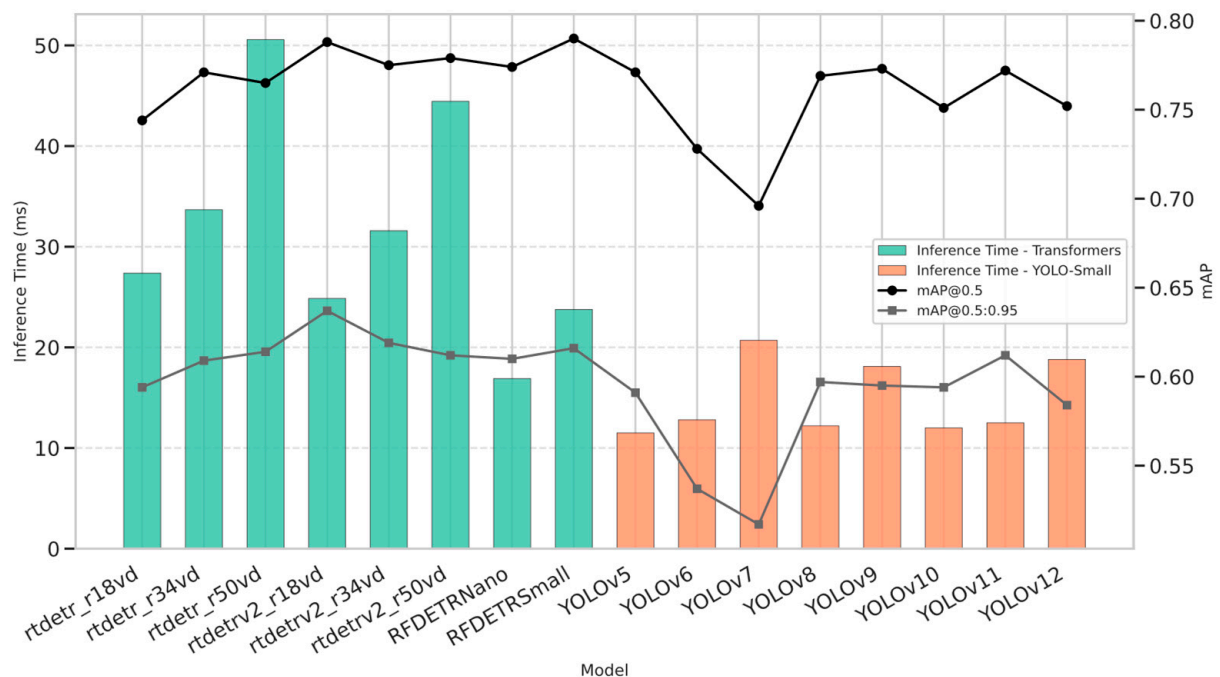


Figure 6. Comparison of accuracy and inference time across different models. The results include inference time, mAP@0.5, and mAP@0.5–0.95 values.

It can be observed that Transformer-based models generally perform better than YOLO models; however, their inference times are also higher compared to the YOLO architecture. RFDETR-Small and RT-DETRv2_R18vd achieved the highest mAP@0.5 among all models, with inference times of about 23 ms and 24 ms, respectively. In contrast, the fastest inference times were obtained by YOLOv5, YOLOv8, YOLOv10, and YOLOv11. The slowest models were RT-DETR_R50vd and RT-DETRv2_R50vd, with inference times of 50.57 ms and 44.44 ms. Another relevant conclusion is that Real-Time DETR version 2 achieves lower inference time compared to its previous version while maintaining similar accuracy. As shown in Figure 6, among Transformer-based models, RF-DETR variants have the lowest inference times while maintaining a high mAP@0.5. This indicates that these models are good candidates for real-time applications that require both accuracy and fast response. It is concluded that RT-DETRv2_R18vd achieves the highest mAP@0.5:0.95 among all models. This indicates that its predicted bounding boxes align more closely with the ground-truth boxes, showing better localization performance; for applications where localization is prioritized over other factors, this model demonstrates superior performance in precisely identifying faults. Overall, if the goal is to achieve the highest precision and recall without constraints on inference time, Transformer-based models are strong choices, although when time efficiency becomes a constraint, lighter Transformer-based models such as RF-DETR-Nano, or YOLO models like YOLOv5 and YOLOv11, are more suitable options.

Table 2 shows the performance of Transformer-based models on different dataset classes using the AP@0.5 metric. RT-DETR_R34vd achieves the highest AP@0.5 on the MBP class with a value of 0.943, while RT-DETRv2_R50vd records the lowest AP@0.5 on the MD class with a value of 0.464.

Table 2. Performance (AP@0.5) of different models on various dataset classes: MBP, MD, MHS, SBP, SD, SHS, SOC, and SRP.

Model	MBP	MD	MHS	SBP	SD	SHS	SOC	SRP
rt detr_r18vd	0.899	0.467	0.767	0.916	0.768	0.815	0.776	0.538
rt detr_r34vd	0.943	0.483	0.799	0.927	0.803	0.812	0.811	0.583
rt detr_r50vd	0.929	0.475	0.783	0.930	0.796	0.801	0.816	0.586
rt detr2_r18vd	0.918	0.593	0.806	0.931	0.813	0.798	0.838	0.602
rt detr2_r34vd	0.889	0.552	0.823	0.927	0.818	0.814	0.787	0.588
rt detr2_r50vd	0.939	0.464	0.786	0.935	0.819	0.808	0.805	0.675
RFDETR-Nano	0.921	0.560	0.758	0.914	0.732	0.796	0.803	0.704
RFDETR-Small	0.897	0.586	0.778	0.915	0.768	0.791	0.824	0.756

Figure 7 illustrates the mAP@0.5 of different models along with the standard deviation (STD) of AP for each class in the dataset. It is also useful to examine model performance based on the variability of scores across all classes. STD quantifies the dispersion of the performance scores, defining a statistical measure of the variability between classes. If two models have a similar mAP@0.5, the model with a lower STD is considered better, as its scores for all classes are more consistent and closer to the overall mAP@0.5. RF-DETR-Small has the lowest STD, meaning that its scores for all classes are close to its overall mAP@0.5, while the mAP@0.5 itself is also high. Interestingly, although RF-DETR-Nano has a lower mAP@0.5 compared to RT-DETRv2_R18vd, its STD is also lower, making it a more consistent choice when considering accuracy across all classes.

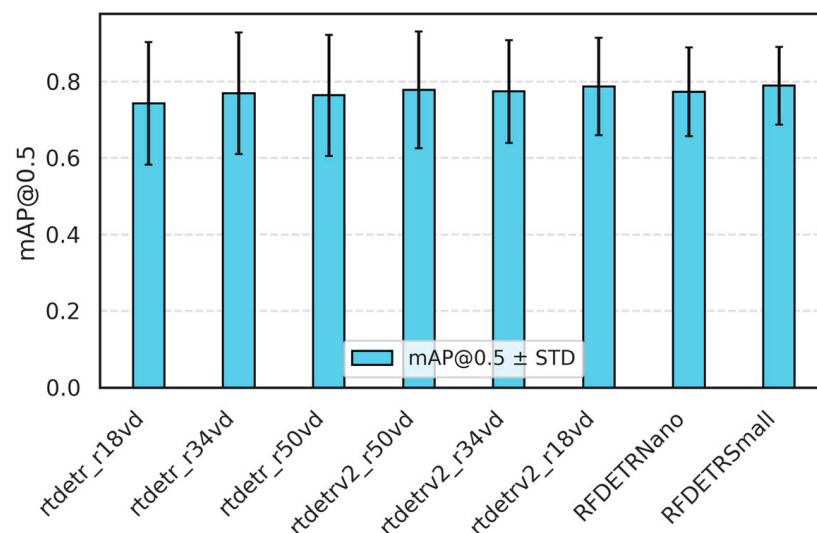
**Figure 7.** mAP@0.5 with STD of AP for different classes. The classes are arranged from highest to lowest STD, left to right.

Figure 8 presents a heatmap of AP@0.5 across different classes. The MD class is the most difficult for models to predict, with an average AP@0.5 of about 0.5. In contrast, the SBP class is the easiest to predict, where the average AP@0.5 across models is around 0.92. The classes can be ranked from hardest to easiest to predict in the following order: MD, SRP, MHS, SD, SHS, SOC, MBP, and SBP; however, this ordering may also be affected by the imbalance present in the dataset, as the MD and SRP classes have a lower number of samples.

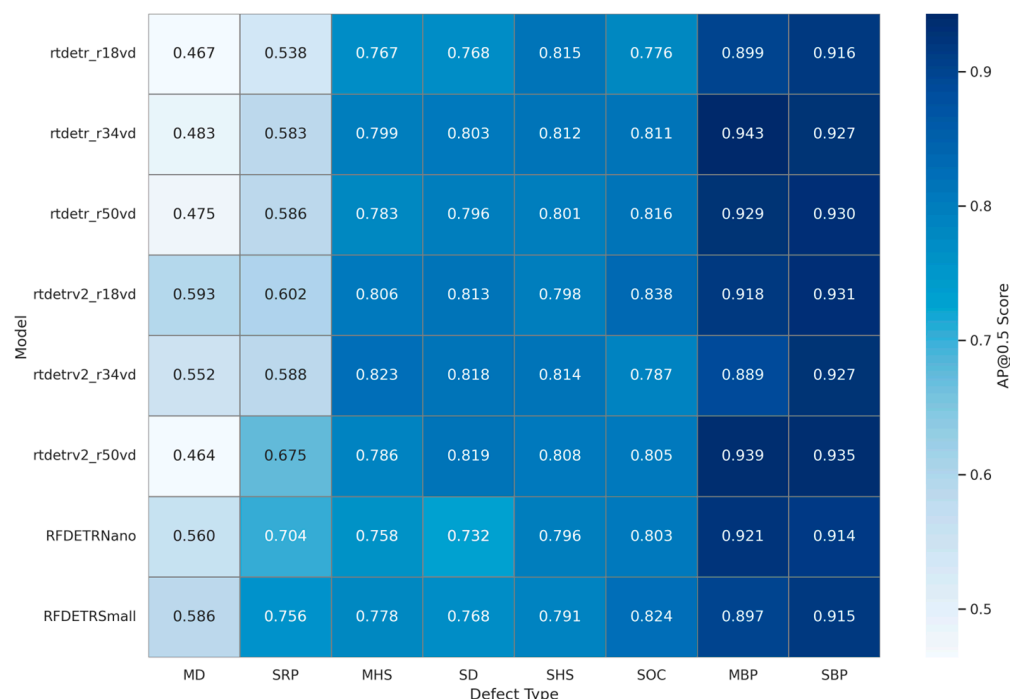


Figure 8. Heatmap of AP@0.5 for different classes for different models.

The best-performing model for each class can be identified as follows: RT-DETRv2_R18vd performs best on the MD class, while RF-DETR-Small achieves the highest accuracy for SRP. For MHS, RT-DETRv2_R34vd shows superior performance, and RT-DETRv2_R50vd is the best model for SD. The SHS class is best predicted by RT-DETR_R18vd, whereas RT-DETRv2_R18vd leads for SOC. Finally, RT-DETR_R34vd excels in the MBP class, and RT-DETRv2_R50vd performs best for SBP.

Figure 9 illustrates the precision, recall, and F1-score for different Transformer-based models. As shown, RT-DETRv2_R34vd and RF-DETR-Small achieve the highest F1-scores across all models, indicating strong overall detection performance with high precision. In terms of precision, RT-DETR_R50vd, RT-DETRv2_R18vd, and RF-DETR-Small achieve the highest values, meaning their predictions are more accurate, but their recall is lower, indicating that they detect fewer total instances. On the other hand, RT-DETRv2_R34vd has the highest recall, demonstrating its superior ability to detect ground-truth bounding boxes.

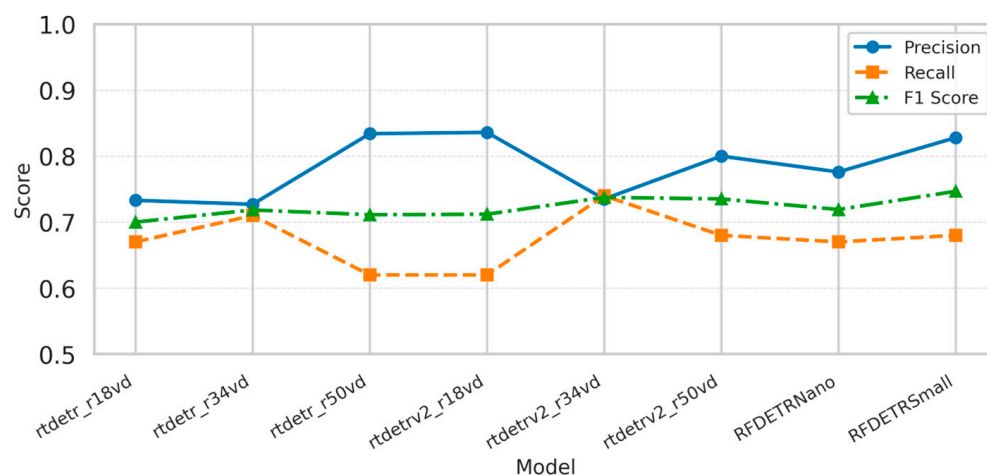


Figure 9. Comparison of precision, recall, and F1-score values across different models.

The performance of the models is analyzed through the detection of objects of different sizes. Object sizes were categorized as follows: areas between 0 and 32×32 pixels were considered small, 32×32 to 96×96 pixels as medium, and above 96×96 pixels as large. RT-DETRv2_R18vd achieves the highest mAP for small objects like the SHS class. This can be attributed to its relatively small backbone, which helps preserve feature quality and reduces the destructive effects of down-sampling. As a result, the feature maps passed to the encoder contain more useful details for detecting small objects. Another important factor is the design of the RT-DETRv2 architecture. In RT-DETRv2, the encoder explicitly decouples intra-scale attention from cross-scale fusion, which prevents small-scale features from being overshadowed or faded during feature fusion. This architectural design allows the model to retain fine-grained information and improves its performance on small object categories, whereas RF-DETR-Small performs the worst in detecting small instances, as shown in Figure 10. For medium-sized objects, RT-DETRv2_R18vd outperforms other models, while RT-DETR_R18vd and RT-DETRv2_R34vd show the lowest accuracy in this category. In the large-object category, RF-DETR-Small demonstrates superior mAP, whereas RT-DETRv2_R50vd achieves the lowest performance. RF-DETR generally performs well on large objects, as its architecture and NAS-driven design provide a large effective receptive field and scale-aware feature fusion, making it better suited for capturing coarse, high-level features required for large-object detection.

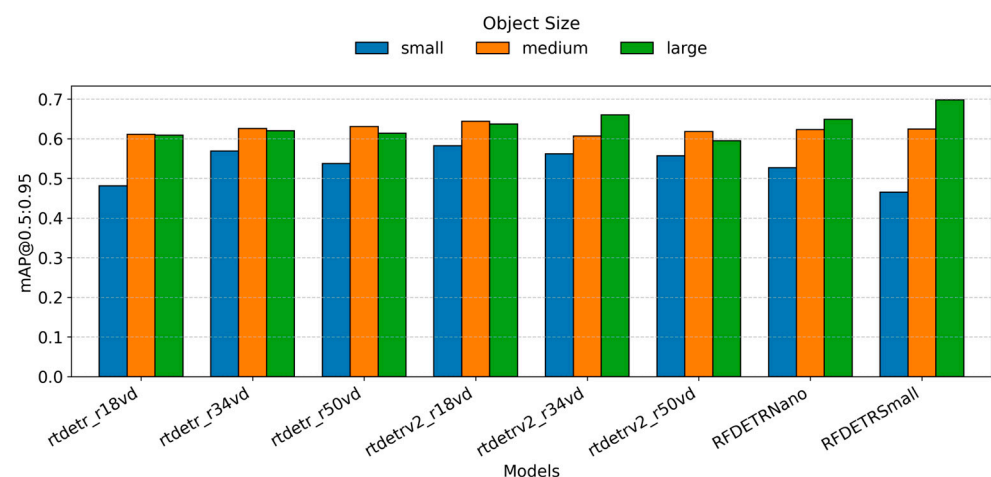


Figure 10. Performance of different models on objects of varying sizes in images.

The reliability of the RF-DETR-Small model has been assessed using a new dataset composed of 84 infrared images from a different PV plant captured by a drone and not included during training or validation phases. The aim is to demonstrate the technical capabilities of the selected network under varying background conditions. The results are promising, with precision, recall, and F1-scores of 84.56%, 79.93%, and 82.18%, respectively, confirming the robustness of the model under unseen operational conditions. In this test, the main objective is fault detection to evaluate the overall detection consistency of the model; for this reason, defect categories were not differentiated, and performance metrics were calculated for two classes: defected and non-defected. The results demonstrate that the model maintains high detection reliability under new conditions, indicating its potential for detecting defects in previously unseen locations. Figure 11 illustrates example predictions from the RF-DETR-Small model on unseen PV images. As shown, the model successfully detects and localizes Single SHS and Multi MHS defects with high accuracy. These conclusions highlight the strength of the selected model as a scalable and efficient solution for automated PV inspection.

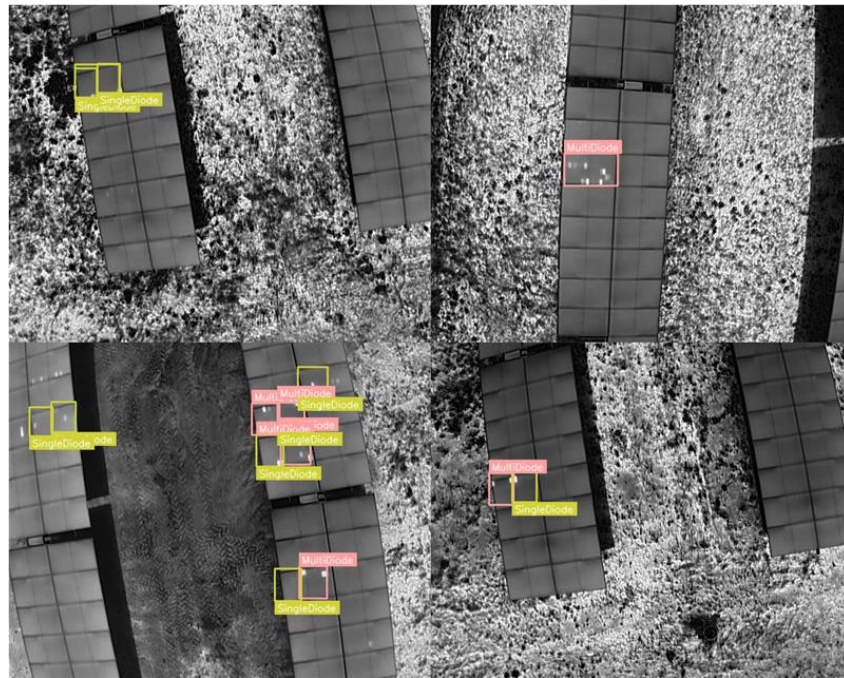


Figure 11. Sample results with the performance of the model on unseen PV plant images.

4. Discussion

Several recent state-of-the-art lightweight models from two main families were analyzed and compared: YOLO-based and Transformer-based architectures. This study also examined the performance of the models across different defect types within the dataset. This research provides useful guidance on selecting suitable models depending on whether accuracy or inference time is the main priority. Additionally, it highlights which defect categories are more challenging or easier for current models to detect. By offering a comprehensive evaluation of how each model performs across various defect types and object sizes, this study helps clarify what level of performance can be expected in different scenarios. It is also observed that in cases where two models achieve nearly the same mAP value, the better choice can be the one with the lower STD across classes. A lower STD indicates that the model performs more consistently among all defect categories rather than performing well only on a few of them. All results presented in this study were obtained under the same experimental conditions to ensure fairness and comparability. Therefore, the findings can serve as a reliable reference for future academic research and industrial applications in PV fault detection.

The best-performing model has been tested in images captured from a new PV station to verify that these models can be used in real-world applications for detecting defects from thermal images. The main conclusions are that, although the model was not specifically trained for the new dataset, the model successfully detected hot spot defect type by identifying regions with noticeably higher temperature compared to their surroundings, demonstrating its capability to generalize and operate effectively outside the training environment. These findings support real-time applications by enabling practitioners to choose the most appropriate model based on their specific requirements and constraints.

5. Conclusions and Future Work

This study demonstrates that PV defects can be effectively detected and localized using novel vision-based approaches in a short time and with high accuracy. Two of the most widely used deep learning architectures were compared: YOLO- and Transformer-based models. The analysis demonstrated that both groups could detect defects accurately while

maintaining relatively low inference time, which makes them suitable for deployment on edge devices for real-time monitoring. RF-DETR-Small and RT-DETRv2-R18vd provided the best detection performance among all models. However, when inference time is the main priority, YOLOv5 and YOLOv10 are the most suitable options due to their faster processing times. If both accuracy and inference time are considered important simultaneously, models such as YOLOv5, YOLOv11, and RF-DETR-Small provide a balanced trade-off between performance and speed. RF-DETR-Small has been tested in a new dataset not applied during training, demonstrating high accuracy and reliable results.

Despite the promising results, this comparative study has limitations that merit discussion. First, the classes in the dataset are imbalanced, which means some rare classes appear far less frequently than others. Second, the new dataset from another PV plant contains a limited number of images with classification labels due to constraints in image collection and manual annotation.

These findings support the use of lightweight deep learning models for practical, real-time PV defect detection and maintenance applications. For future work, it is recommended to compare emerging architectures in the areas of few-shot and zero-shot learning with the models evaluated in this study to examine their potential benefits and limitations for PV defect detection. Additionally, it would be valuable to investigate different approaches for addressing dataset imbalance, as improving class balance could lead to higher overall detection accuracy and more consistent performance across defect categories.

Author Contributions: Conceptualization, I.S.R. and C.Q.G.M.; methodology, I.S.R. and C.Q.G.M.; software, M.S.; validation, M.S., I.S.R. and C.Q.G.M.; formal analysis, M.S., I.S.R. and C.Q.G.M.; investigation, M.S., I.S.R. and C.Q.G.M.; resources, M.S., I.S.R. and C.Q.G.M.; data curation, M.S.; writing—original draft preparation, M.S., I.S.R. and C.Q.G.M.; writing—review and editing, M.S., I.S.R. and C.Q.G.M.; visualization, M.S., I.S.R. and C.Q.G.M.; supervision, I.S.R. and C.Q.G.M.; project administration, C.Q.G.M.; funding acquisition, C.Q.G.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the project “ICARUS—Inspección y Control Automatizado con Redes neuronales y UAVs en Sistemas Fotovoltaicos” [ref. SI4/PJI/2024-00233] supported by the Comunidad de Madrid through the direct grant agreement for the promotion of research and technology transfer at the Universidad Autónoma de Madrid.

Data Availability Statement: The original data presented in the study are openly available from the Thermal Solar PV Anomaly Detection Dataset at doi: 10.13140/RG.2.2.12595.54564 or Reference [30].

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

Abbreviations

The following abbreviations are used in this manuscript:

PV	Photovoltaic
UAVs	Unmanned Aerial Vehicles
AI	Artificial Intelligence
CNNs	Convolutional Neural Networks
YOLO	You Only Look Once
DPiT	Detecting defects of Photovoltaic solar cells with Image Transformers
mAP	Mean Average Precision
EIOU	Efficient Intersection over Union
ViTs	Vision Transformers
IoU	Intersection over Union

DETR	Detection Transformer
SGD	Stochastic gradient descent
MBP	MultiByPassed
MD	MultiDiode
MHS	MultiHotSpot
SBP	SingleByPassed
SD	SingleDiode
SHS	SingleHotSpot
SOC	StringOpenCircuit
AUC	Area Under the Curve
AP	Average Precision
STD	Standard Deviation

References

1. Segovia Ramírez, I.; García Márquez, F.P.; Parra Chaparro, J. Convolutional neural networks and Internet of Things for fault detection by aerial monitoring of photovoltaic solar plants. *Measurement* **2024**, *234*, 114861. [\[CrossRef\]](#)
2. Gangwar, P.; Tripathi, R.P.; Singh, A.K. Solar photovoltaic tree: A review of designs, performance, applications, and challenges. *Energy Sources Part A Recovery Util. Environ. Eff.* **2025**, *47*, 5910–5937. [\[CrossRef\]](#)
3. Meribout, M.; Kumar Tiwari, V.; Pablo Peña Herrera, J.; Najeeb Mahfoudh Awadh Baobaid, A. Solar panel inspection techniques and prospects. *Measurement* **2023**, *209*, 112466. [\[CrossRef\]](#)
4. Khan, M.J.; Kumar, D.; Narayan, Y.; Malik, H.; García Márquez, F.P.; Gómez Muñoz, C.Q. A Novel Artificial Intelligence Maximum Power Point Tracking Technique for Integrated PV-WT-FC Frameworks. *Energies* **2022**, *15*, 3352. [\[CrossRef\]](#)
5. Waqar Akram, M.; Li, G.; Jin, Y.; Chen, X. Failures of Photovoltaic modules and their Detection: A Review. *Appl. Energy* **2022**, *313*, 118822. [\[CrossRef\]](#)
6. Yang, B.; Zheng, R.; Han, Y.; Huang, J.; Li, M.; Shu, H.; Su, S.; Guo, Z. Recent Advances in Fault Diagnosis Techniques for Photovoltaic Systems: A Critical Review. *Prot. Control Mod. Power Syst.* **2024**, *9*, 36–59. [\[CrossRef\]](#)
7. Mellit, A.; Tina, G.M.; Kalogirou, S.A. Fault detection and diagnosis methods for photovoltaic systems: A review. *Renew. Sustain. Energy Rev.* **2018**, *91*, 1–17. [\[CrossRef\]](#)
8. Bodnár, I.; Matusz-Kalász, D.; Boros, R.R. Exploration of Solar Panel Damage and Service Life Reduction Using Condition Assessment, Dust Accumulation, and Material Testing. *Sustainability* **2023**, *15*, 9615. [\[CrossRef\]](#)
9. Segovia Ramírez, I.; Das, B.; García Márquez, F.P. Fault detection and diagnosis in photovoltaic panels by radiometric sensors embedded in unmanned aerial vehicles. *Prog. Photovolt. Res. Appl.* **2022**, *30*, 240–256. [\[CrossRef\]](#)
10. Triki-Lahiani, A.; Bennani-Ben Abdelghani, A.; Slama-Belkhodja, I. Fault detection and monitoring systems for photovoltaic installations: A review. *Renew. Sustain. Energy Rev.* **2018**, *82*, 2680–2692. [\[CrossRef\]](#)
11. Alimi, O.A.; Meyer, E.L.; Olayiwola, O.I. Solar Photovoltaic Modules' Performance Reliability and Degradation Analysis—A Review. *Energies* **2022**, *15*, 5964. [\[CrossRef\]](#)
12. Yang, C.; Sun, F.; Zou, Y.; Lv, Z.; Xue, L.; Jiang, C.; Liu, S.; Zhao, B.; Cui, H. A Survey of Photovoltaic Panel Overlay and Fault Detection Methods. *Energies* **2024**, *17*, 837. [\[CrossRef\]](#)
13. Gómez Muñoz, C.Q.; García Márquez, F.P. Future Maintenance Management in Renewable Energies. In *Renewable Energies: Business Outlook 2050*; García Márquez, F., Karyotakis, A., Papaelias, M., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 149–159. [\[CrossRef\]](#)
14. Polymeropoulos, I.; Bezyrgiannidis, S.; Vrochidou, E.; Papakostas, G.A. Enhancing Solar Plant Efficiency: A Review of Vision-Based Monitoring and Fault Detection Techniques. *Technologies* **2024**, *12*, 175. [\[CrossRef\]](#)
15. Xie, H.; Yuan, B.; Hu, C.; Gao, Y.; Wang, F.; Wang, C.; Wang, Y.; Chu, P. ST-YOLO: A defect detection method for photovoltaic modules based on infrared thermal imaging and machine vision technology. *PLoS ONE* **2024**, *19*, e0310742. [\[CrossRef\]](#)
16. Ding, S.; Jing, W.; Chen, H.; Chen, C. Yolo Based Defects Detection Algorithm for EL in PV Modules with Focal and Efficient IoU Loss. *Appl. Sci.* **2024**, *14*, 7493. [\[CrossRef\]](#)
17. Lei, Y.; Wang, X.; An, A.; Guan, H. Deeplab-YOLO: A method for detecting hot-spot defects in infrared image PV panels by combining segmentation and detection. *J. Real-Time Image Process.* **2024**, *21*, 52. [\[CrossRef\]](#)
18. Cao, Y.; Pang, D.; Yan, Y.; Jiang, Y.; Tian, C. A photovoltaic surface defect detection method for building based on deep learning. *J. Build. Eng.* **2023**, *70*, 106375. [\[CrossRef\]](#)
19. Ghahremani, A.; Adams, S.D.; Norton, M.; Khoo, S.Y.; Kouzani, A.Z. Detecting Defects in Solar Panels Using the YOLO v10 and v11 Algorithms. *Electronics* **2025**, *14*, 344. [\[CrossRef\]](#)

20. Wang, J.; Du, H.; Zeng, Y. PDFormer: Efficient Vision Transformer for Photovoltaic Defect Detection. *IEEE Trans. Consum. Electron.* **2025**, *71*, 6602–6611. [CrossRef]
21. Ramadan, E.A.; Moawad, N.M.; Abouzalm, B.A.; Sakr, A.A.; Abouzaid, W.F.; El-Banby, G.M. An innovative transformer neural network for fault detection and classification for photovoltaic modules. *Energy Convers. Manag.* **2024**, *314*, 118718. [CrossRef]
22. Xie, X.; Liu, H.; Na, Z.; Luo, X.; Wang, D.; Leng, B. DPiT: Detecting Defects of Photovoltaic Solar Cells With Image Transformers. *IEEE Access* **2021**, *9*, 154292–154303. [CrossRef]
23. Yang, Y.; Zhang, J.; Shu, X.; Pan, L.; Zhang, M. A Lightweight Transformer Model for Defect Detection in Electroluminescence Images of Photovoltaic Cells. *IEEE Access* **2024**, *12*, 194922–194931. [CrossRef]
24. Haddad, H.; Jerbi, F.; Smaali, I. Defect Detection in PV Electroluminescence Images Using YOLOv9 to YOLOv12 Lightweights Variants. In Proceedings of the 2025 25th International Conference on Digital Signal Processing (DSP), Pylos, Greece, 25–27 June 2025; pp. 1–5. [CrossRef]
25. Terven, J.; Córdova-Esparza, D.-M.; Romero-González, J.-A. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Mach. Learn. Knowl. Extr.* **2023**, *5*, 1680–1716. [CrossRef]
26. Diwan, T.; Anirudh, G.; Tembhurne, J.V. Object detection using YOLO: Challenges, architectural successors, datasets and applications. *Multimed. Tools Appl.* **2023**, *82*, 9243–9275. [CrossRef]
27. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *CoRR*. 2017. Available online: <http://arxiv.org/abs/1706.03762> (accessed on 15 September 2025).
28. 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. *CoRR*. 2020. Available online: <https://arxiv.org/abs/2010.11929> (accessed on 10 September 2025).
29. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. 2020. Available online: <https://arxiv.org/abs/2005.12872> (accessed on 10 September 2025).
30. Darabi, P. ThermoSolar-PV: A Curated Thermal Imagery Dataset for Anomaly Detection in Photovoltaic Modules. June 2025. Available online: https://www.researchgate.net/publication/390740926_ThermoSolar-PV_A_Curated_Thermal_Imagery_Dataset_for_Anomaly_Detection_in_Photovoltaic_Modules?channel=doi&linkId=67fb7d12df0e3f544f41087a&showFulltext=true (accessed on 6 October 2025).
31. Jocher, G. Ultralytics YOLOv5. 2020. Available online: <https://zenodo.org/records/7347926> (accessed on 15 October 2025).
32. Li, C.; Li, L.; Geng, Y.; Jiang, H.; Cheng, M.; Zhang, B.; Ke, Z.; Xu, X.; Chu, X. YOLOv6 v3.0: A Full-Scale Reloading. *arXiv* **2023**, arXiv:2301.05586. [CrossRef]
33. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv* **2022**, arXiv:2207.02696.
34. Jocher, G.; Chaurasia, A.; Qiu, J. Ultralytics YOLOv8. 2023. Available online: <https://github.com/ultralytics/ultralytics> (accessed on 20 October 2025).
35. Wang, C.-Y.; Liao, H.-Y.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616. [CrossRef]
36. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. *arXiv* **2024**, arXiv:2405.14458.
37. Jocher, G.; Qiu, J. Ultralytics YOLO11. 2024. Available online: <https://github.com/ultralytics/ultralytics> (accessed on 12 November 2025).
38. Tian, Y.; Ye, Q.; Doermann, D. YOLO12: Attention-Centric Real-Time Object Detectors. *arXiv* **2025**, arXiv:2502.12524.
39. Zhao, Y.; Lv, W.; Xu, S.; Wei, J.; Wang, G.; Dang, Q.; Liu, Y.; Chen, J. DETRs Beat YOLOs on Real-time Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 16–22 June 2024.
40. Lv, W.; Zhao, Y.; Chang, Q.; Huang, K.; Wang, G.; Liu, Y. RT-DETRv2: Improved Baseline with Bag-of-Freebies for Real-Time Detection Transformer. 2024. Available online: <https://arxiv.org/abs/2407.17140> (accessed on 18 September 2025).
41. Robinson, I.; Robicheaux, P.; Popov, M.; Ramanan, D.; Peri, N. RF-DETR: Neural Architecture Search for Real-Time Detection Transformers. 2025. Available online: <https://arxiv.org/abs/2511.09554> (accessed on 20 September 2025).

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.