

Full length article

YOLO-TDH: An object detection framework for power transmission line inspection[☆]Baoye Song^a, Shihao Zhao^a, Weibo Liu^{b,*,*}, Yani Xue^b^a College of Electrical Engineering and Automation, Shandong University of Science and Technology, Qingdao 266590, China^b Department of Computer Science, Brunel University of London, Uxbridge, Middlesex, UB8 3PH, United Kingdom

ARTICLE INFO

Keywords:

Vision-based inspection
Power transmission line
Lightweight
YOLO
Object detection

ABSTRACT

Coupled with vision-based inspection techniques, unmanned aerial vehicles (UAVs) have been extensively applied to power transmission line inspection. UAV images typically cover a wide field of view, they often contain complex backgrounds in power transmission line inspection, which make the accurate detection and localization of small-size defects challenging, especially on resource-constrained devices. To tackle this issue, this paper proposes YOLO-TDH, a novel lightweight object detection framework derived from YOLOv8. The proposed YOLO-TDH incorporates three key components to improve the detection of small-size defects: (1) an Enhanced Feature Integration Module (EFIM), which strengthens multi-scale feature extraction and captures fine-grained details required to distinguish defective components from normal ones; (2) an Enhanced Feature Fusion Module (EFFM), which optimizes feature information flow and reduce the loss of critical details for small-size defects; and (3) a novel Transformer-based decoder, which models global context and alleviates ambiguity among overlapping components. The ShapeloU loss is employed to improve the bounding box regression accuracy. Experiments on a dedicated transmission line dataset show that YOLO-TDH, Insulator dataset for targeted fault verification, and the public VisDrone2019 benchmark to validate generalization capability. The results demonstrate that YOLO-TDH consistently outperforms existing state-of-the-art lightweight methods in terms of Precision, Recall, mAP@0.5, and mAP@0.5:0.95. Results show that the proposed YOLO-TDH achieves a proper balance between diagnostic accuracy and computational efficiency, providing a robust solution for real-time, fine-grained health monitoring in resource-constrained scenarios.

1. Introduction

Power transmission systems form the backbone of modern energy infrastructure and consist of diverse components, including large-scale components (e.g., insulators and warning spheres), small-scale ones such as dampers and spacers, and other line fittings. The service condition of these components, together with the presence of external hazards (e.g., bird nests), can significantly affect the operational reliability and safety of the grid [1,2]. Routine inspection is therefore essential for the timely identification of incipient defects and potential failure risks [3]. In recent years, unmanned aerial vehicles (UAVs) have been extensively applied to power transmission line inspection due to their flexibility, high efficiency, and ability to acquire high-resolution images over long-distance corridors even in complex environments, which offers a practical alternative to conventional manual patrols [4,5].

Despite the practical advantages of UAVs in power transmission line inspection, automated analysis remains largely dominated by task-specific pipelines, where each pipeline is specifically designed for a particular component type or defect category. Existing studies mainly focus on detecting specific targets (e.g., foreign object intrusion, insulator faults, or ultra-small bolt defects) [6–10]. Although these methods have shown promising performance on their target tasks, they are not well suited to the joint assessment of multiple component and defect categories within the same scene [11].

In practice, a transmission span may contain several co-occurring conditions, whereas a detector designed for only a narrow set of targets cannot provide a complete assessment of line condition. For example, a model may successfully detect insulator damage while failing to identify a missing damper in the same span because the latter is without the scope of the designed detection task. This limitation is particularly

[☆] This work was supported in part by the Natural Science Foundation of Shandong Province of China under Grant ZR2023MF067 and the National Natural Science Foundation of China under Grant 62233012.

* Corresponding author.

E-mail addresses: songbaoye@sdust.edu.cn (B. Song), shihao.zhao@sdust.edu.cn (S. Zhao), Weibo.Liu2@brunel.ac.uk (W. Liu).

<https://doi.org/10.1016/j.aei.2026.105273>

Received 19 March 2026; Received in revised form 12 June 2026; Accepted 28 August 2026

Available online 15 September 2026

1474-0346/© 2026 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

restrictive in inspection scenarios that require simultaneous recognition of component categories, defect patterns, and environmental hazards. Therefore, single-target detection is insufficient for practical power transmission line inspection. A seemingly natural idea is to develop a unified and fine-grained inspection framework can detect heterogeneous targets within a single model rather than addressing each inspection objective in isolation.

Note that joint detection of multiple components in UAV-based inspection remains technically challenging, largely because of the severe scale variation inherent in transmission infrastructure [12]. This challenge is further amplified by the wide field of view of UAV imagery, in which small-size targets often occupy only a few pixels and are easily affected by background clutter, structural overlap, and line occlusion [13]. Under resource-constrained deployment, the problem becomes even more difficult, since lightweight detectors tend to lose fine spatial details during repeated down-sampling, which degrades the small-object detection performance [14]. Meanwhile, transformer-based detectors (e.g., DETR and RT-DETR) offer stronger long-range context modeling, but such design can also increase the computational cost, making real-time deployment on edge devices challenging [15, 16]. These challenges motivate the development of a unified detection framework that can balance global context modeling and computational efficiency for high-precision multi-scale object detection.

To address these issues, this paper proposes YOLO with Transformer Decoder Head (YOLO-TDH), a unified detection framework built on YOLOv8 [17–19] for power transmission line inspection. By integrating an Enhanced Feature Integration Module and a Transformer-based Decoder into the YOLOv8 architecture, the proposed YOLO-TDH is specifically designed to improve feature representation under complex backgrounds with large scale variation and target overlap. Unlike task-specific detectors that focus on a single component or defect type, YOLO-TDH is put forward for the joint detection of component categories and condition states, including normal components, defect patterns, and environmental intrusions (e.g., bird nests). In this way, proposed framework moves beyond isolated single-task detection and enables a more complete assessment of power transmission line condition in complex inspection scenes using UAVs.

The main contributions of this paper can be summarized as follows.

- (1) A novel Deep Fusion Network (DFNet) is proposed as the backbone of the detector. By incorporating an Enhanced Feature Integration Module (EFIM) with cascaded StarBlocks, DFNet improves fine-grained feature extraction for small defects in complex backgrounds.
- (2) An Enhanced Feature Fusion Module (EFFM), inspired by the Bidirectional Feature Pyramid Network (BiFPN), is developed to achieve bidirectional weighted fusion of multi-scale features. The proposed EFFM effectively integrates spatial details and semantic information, which leads to stronger multi-scale representation capability.
- (3) A hybrid-attention Transformer Decoder Head (TDH) is introduced by combining the IoU-aware Query Selection (IQS), Multi-Head Self-Attention (MHSA), and Multi-Scale Deformable Attention (MSDA). The proposed decoder captures global dependencies while adaptively focusing on salient features across scales, thereby improving the defect localization accuracy.
- (4) A joint loss function is constructed by combining the VariFocal Loss (VFL), the ShapeIoU and a regularization term. This loss function jointly improves the bounding box localization capability by accounting for boundary overlap, center deviation, and shape mismatch.

The remainder of this paper is structured as follows. Section 2 reviews the development of YOLO and the structure of YOLOv8. Section 3 offers a detailed description of our proposed YOLO-TDH model. Experimental results and discussion are given in Section 4. Finally, conclusions summarizing the findings and outlining potential directions for future research are drawn in Section 5.

2. Preliminaries

2.1. Development of YOLO

Since its inception in 2016, the You Only Look Once (YOLO) series has undergone significant evolution, establishing itself as a fundamental framework in object detection. The initial YOLOv1 [20] has introduced a novel regression-based one-stage detection approach, achieving remarkable inference speed at the cost of limited performance on small objects. To address this limitation, YOLOv2 [21] has incorporated anchor boxes to improve detection accuracy. The subsequent YOLOv3 [22] has further enhanced the architecture by integrating Feature Pyramid Networks (FPN) [23] for effective multi-scale prediction. In 2020, two major advancements have emerged: YOLOv4 [24] has significantly boosted both precision and computational efficiency through its Cross Stage Partial (CSP) Darknet backbone and integration of cutting-edge techniques, while YOLOv5 [25] has improved usability and detection performance by implementing the Generalized Intersection over Union (GIoU) loss function and refined training strategies.

The YOLO series has continued its rapid evolution in subsequent years. In 2021, YOLOX [26] has introduced an anchor-free architecture with enhanced data augmentation and training protocols, demonstrating superior performance. The following year has seen two notable advancements: YOLOv6 [27] that has been optimized for edge devices, and YOLOv7 [28] which has focused on real-time performance improvements. Concurrently, Damo-YOLO [29] has extended YOLOv5's capabilities, achieving remarkable results in complex scenes and small object detection. The year 2023 has witnessed further innovations with Gold-YOLO [30], which has employed lightweight architectures and model compression for efficient deployment on low-power devices. Following this progress, YOLOv8 [31] has introduced an improved network structure featuring novel convolutional layers and an optimized detection head, significantly boosting both speed and accuracy.

In 2024 and early 2025, the iteration of the YOLO family has accelerated further. Researchers have initially enhanced YOLOv5 by integrating programmable gradient information, resulting in the highly efficient YOLOv9 [32]. Subsequently, YOLOv10 [33] has pioneered a Non-Maximum Suppression (NMS)-free training approach using consistent dual assignments, effectively reducing inference latency. The evolution has continued with YOLOv11 [34], which has implemented architectural enhancements to optimize the trade-off between parameter count and feature extraction capability. Most recently, YOLOv12 [35] has marked a paradigm shift by integrating attention mechanisms into the real-time framework, utilizing Area Attention to overcome the receptive field limitations of traditional Convolutional Neural Networks (CNNs) while maintaining high inference speeds.

2.2. Components of YOLOv8

Although newer iterations have introduced novel mechanisms, based on the widespread adoption and the established balance between accuracy and efficiency, this work has selected YOLOv8 as our baseline model for further optimization. As illustrated in Fig. 1, the YOLOv8 architecture has maintained the conventional three-component design: an enhanced backbone network (Backbone) for more effective feature extraction, an optimized feature fusion network (Neck) for improved multi-scale feature integration, and a redesigned detection head (Head) that has boosted both localization and classification performance. The aforementioned components will be described in the following section.

Specifically, the backbone network performs initial multi-scale feature extraction from input images, utilizing multiple convolutional layers combined with cross-stage partial bottleneck (C2f) modules. To further enhance feature extraction capability, it incorporates Spatial Pyramid Pooling Fusion (SPPF) modules at specific stages. Through these operations, the backbone effectively captures both local and

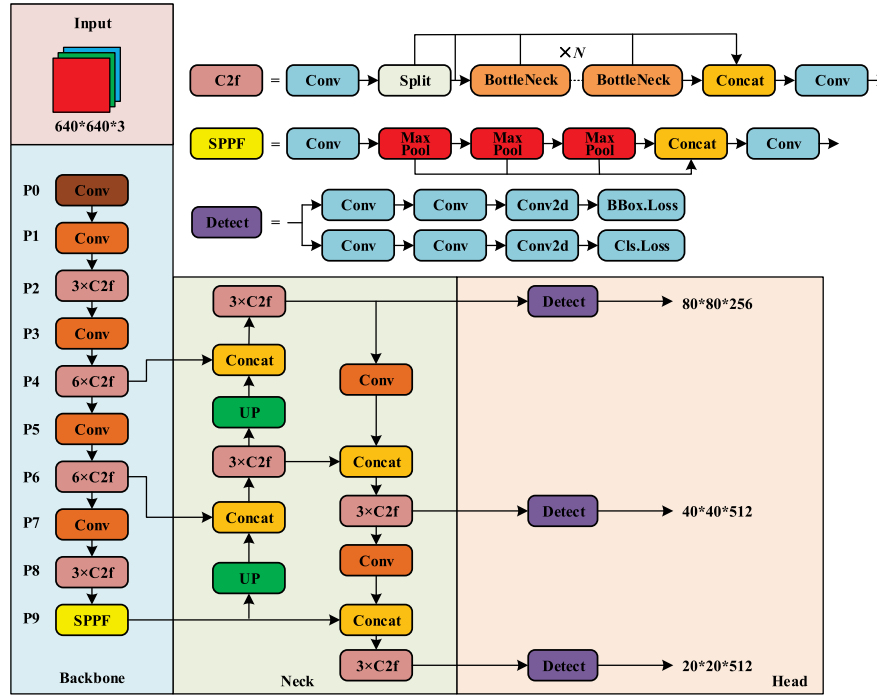


Fig. 1. Structure of YOLOv8.

global characteristics, producing multi-scale feature maps that are subsequently passed to the neck network for refinement.

The neck network serves to boost the model's perceptual ability for multi-scale targets by fusing features extracted at different resolutions. YOLOv8 adopts a refined PAN-FPN architecture that streamlines up-sampling processes to reduce computational demands. By integrating C2f modules throughout the fusion stage, this design improves feature quality and expressive power, ensuring an effective combination of multi-scale features while maintaining fast detection speeds.

Finally, the detection head processes the fused multi-scale features to generate classification results and bounding box predictions. YOLOv8 employs a decoupled head module that separates classification and regression tasks into independent branches, which significantly reduces optimization conflicts. Specifically, the network splits into a regression branch for bounding box coordinates and a classification branch for class probabilities. This decoupled design simultaneously optimizes both classification accuracy and localization precision, enhancing overall detection performance.

3. The proposed YOLO-TDH

Although YOLOv8 has made remarkable progress in object detection, its performance in detecting small targets within complex environments, particularly in high-precision scenarios like transmission line equipment inspection, remains challenged by notable missed detections and false alarms. To address these limitations, this paper proposes the You Only Look Once with Transformer Decoder Head (YOLO-TDH), as illustrated in Fig. 2. Subsequent sections will detail the architectural refinements and implementation methodology of YOLO-TDH, which aims to deliver a more robust and efficient solution for small target detection in complex transmission line environments, thereby substantially enhancing detection performance and practical utility.

3.1. Backbone of YOLO-TDH

The backbone of YOLO-TDH features a hierarchical architecture comprising three core components: standard convolutional layers, proposed Enhanced Feature Integration Modules (EFIM), and SPPF modules. Owing to its deep feature fusion capabilities, we designate this architecture as the Deep Fusion Network (DFNet).

Input data first undergoes preliminary feature extraction through initial convolutional layers. Cascaded EFIM modules then perform multi-stage feature refinement, where each stage stacks convolutional layers with EFIM units to iteratively enhance feature representations. The stacking structure in deeper network segments effectively strengthens spatial and semantic representations, enabling precise capture of fine-grained features in complex environments. Finally, the SPPF module integrates cross-scale features, significantly improving multi-scale adaptability and feature extraction robustness for transmission line equipment inspection. We now present a comprehensive analysis of the proposed EFIM, highlighting its critical components.

The EFIM module shown in Fig. 3, which serves as a pivotal component within YOLO-TDH's backbone network, is designed to enhance feature representation through multi-level extraction and fusion. Architecturally, it sequentially executes: feature transformation via convolutional layers (Conv), parallel processing through channel-wise splitting (Split), hierarchical refinement using stacked StarBlock modules (StarBlock×N) [36], and integrated feature representation via concatenation (Concat).

The EFIM's operational pipeline begins with channel expansion: a 1×1 convolution processes the input feature map (with channel dimension c), doubling its channels to $2c$ for enhanced feature representation. This expanded tensor then undergoes channel-wise splitting, producing two distinct c -channel sub-maps (y_0 and y_1). The second sub-map y_1 serves as input to N sequentially connected StarBlock modules, which iteratively generate new feature maps ($y_2, y_3, \dots, y_{N+1}$) through progressive refinement. The complete set of $N+2$ sub-maps (total channel dimension $(N+2) \times c$) is then channel-concatenated and compressed back to dimension c via a final 1×1 convolution, producing the EFIM's output.

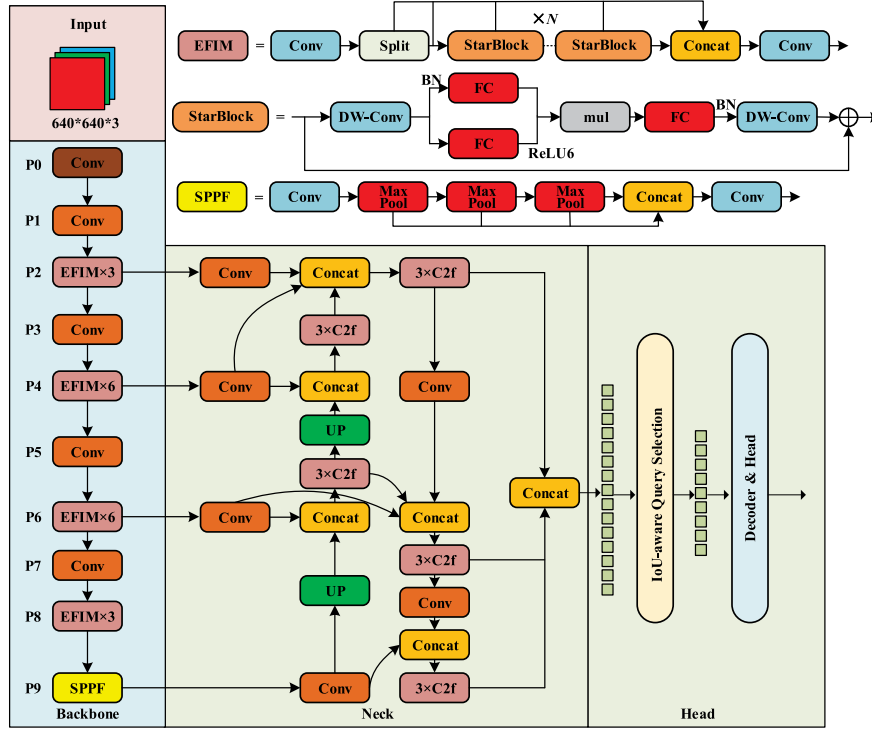


Fig. 2. Structure of YOLO-TDH.

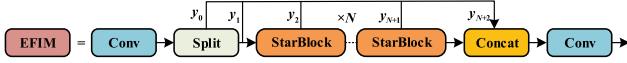


Fig. 3. Structure of the EFIM.

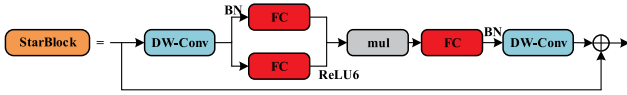


Fig. 4. Structure of the StarBlock.

Further, the StarBlock serves as the core component of the EFIM module in DFNet, achieving deep feature enhancement through multi-stage transformations. As shown in Fig. 4, the module combines depthwise separable convolution (DW-Conv), ReLU6 activation, elementwise multiplication, and residual connections in an integrated architecture. The processing begins with depthwise separable convolution, which splits into two phases: depthwise convolution using 7×7 kernels for channel-wise spatial feature extraction, followed by pointwise convolution with 1×1 kernels for cross-channel fusion and dimensional expansion. This decomposition strategy maintains the ability to capture spatial structure while significantly reducing computational complexity compared to standard convolution.

Remark 1. The proposed DFNet establishes a robust feature extraction framework through three key components: standard convolutional layers for foundational feature encoding, deeply stacked EFIM modules enabling progressive feature refinement, and an SPPF module performing cross-scale feature fusion. This hierarchical architecture demonstrates superior capability in handling multi-scale object detection tasks, particularly in complex visual environments where discriminative feature representation is critical.

3.2. Neck of YOLO-TDH

The Enhanced Feature Fusion Module (EFFM) serves as a critical architectural bridge in YOLO-TDH, connecting the backbone network with the detection head while playing a pivotal role in power transmission line equipment inspection. Specifically designed for feature map refinement, the module processes backbone-extracted features through multi-stage optimization to generate discriminative representations. This systematic enhancement enables reliable detection of transmission line components even under challenging environmental conditions.

The EFFM is adapted from the BiFPN framework [37], specifically customized to process DFNet's structural outputs. The module takes multi-scale backbone features P_2 , P_4 , P_6 , and P_9 as input, corresponding to feature maps with resolutions of 160×160 , 80×80 , 40×40 , and 20×20 , respectively. Each of the features contains distinct hierarchical semantic and spatial information. As shown in Fig. 5, the EFFM integrates these multi-scale features to enable effective cross-level information fusion, significantly enhancing the model's feature representation capability.

Specifically, the EFFM adopts a bidirectional fusion strategy that utilizes: standard convolution (Conv) for feature transformation, up-sampling (Upsample) for resolution enhancement, feature fusion (Concat) for cross-scale integration, and bottleneck-based C2f modules for efficient feature processing. This architecture constructs a multi-scale feature pyramid that simultaneously enhances semantic information and optimizes spatial resolution, as illustrated in Fig. 5.

Initially, feature maps P_4 , P_6 , and P_9 (with resolutions of 80×80 , 40×40 , and 20×20 , and channel dimensions of 256, 512, and 1024, respectively) are standardized to 256 channels via 1×1 convolutions. In the upward path, P_9 undergoes up-sampling (nearest-neighbor interpolation) to achieve 40×40 resolution. The upsampled feature map then engages in BiFPN-style weighted fusion with P_6 , where both inputs share identical resolution (40×40) and channel dimension (256). The fused features are processed through a C2f module to generate P'_6 . Subsequently, P'_6 is upsampled to 80×80 resolution and undergoes

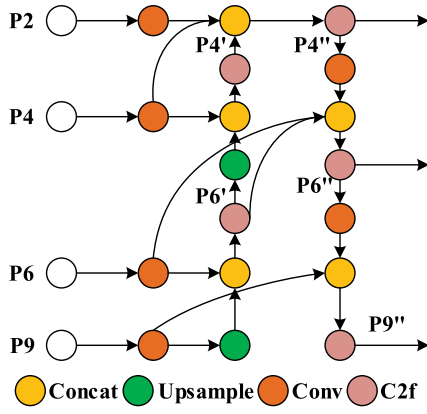


Fig. 5. Neck of YOLO-TDH.

identical BiFPN-weighted fusion with P_4 , followed by C2f-based refinement to produce P_4' . In the downward path, P_2 (160×160 , 128 channels) undergoes 3×3 convolution with stride 2 for downsampling to 80×80 resolution (256 channels). The downsampled feature map then participates in BiFPN-style fusion with both P_4 and P_4' , followed by C2f module processing to generate P_4'' . Subsequently, P_4'' is further downsampled to 40×40 resolution and undergoes BiFPN-weighted fusion with F_6 and F_6' to produce P_6'' . This process repeats as P_6'' is downsampled to 20×20 resolution and fused with P_9 through the same BiFPN procedure, ultimately yielding P_9'' after C2f refinement. The final feature maps P_4'' , P_6'' , and P_9'' (with resolutions of 80×80 , 40×40 , and 20×20 , all with 256 channels) are concatenated to form a unified multi-scale feature map (denoted as F_{multi}), which is subsequently fed into the decoder module for the final detection of power transmission line equipment.

Remark 2. The Neck of YOLO-TDH significantly enhances feature representation quality through its hierarchical architecture, enabling robust multi-scale processing capability essential for detecting various power transmission equipment. The EFFM's bidirectional processing pipeline ensures optimal feature map transformation during propagation from the backbone network to the detection head, achieving both high efficiency and precision in object detection, even when operating in challenging grid environments with occlusions and scale variations.

3.3. Head of YOLO-TDH

The detection head of YOLO-TDH, illustrated in Fig. 6, acts as the critical interface between the EFFM and the final detection outputs, accurately transforming multi-scale feature maps into class probabilities and bounding box coordinates. Our design extends the RT-DETR framework [16] by integrating both Multi-Head Self-Attention (MHSA) [38] and Multi-Scale Deformable Attention (MSDA) mechanisms [39]. This hybrid scheme achieves significant improvements in feature discriminability, particularly in challenging power transmission line scenarios with small-scale targets and cluttered backgrounds, ensuring robust performance against environmental disturbances.

The pipeline of the detection head begins with IoU-aware Query Selection (IQS) [16], which filters the initial object queries to improve localization accuracy. The IQS algorithm operates by evaluating the Intersection over Union (IoU) metric that measures the overlap between predicted and ground-truth bounding boxes. This process is formally defined as:

$$Q_{\text{selected}} = f_{\text{IoU}}(Q), \quad (1)$$

where $Q \in \mathbb{R}^{N \times D}$ denotes the input set of initial queries, N represents the number of queries, and D indicates the feature dimension. The function $f_{\text{IoU}}(\cdot)$ is to identify the top 300 queries that are most indicative

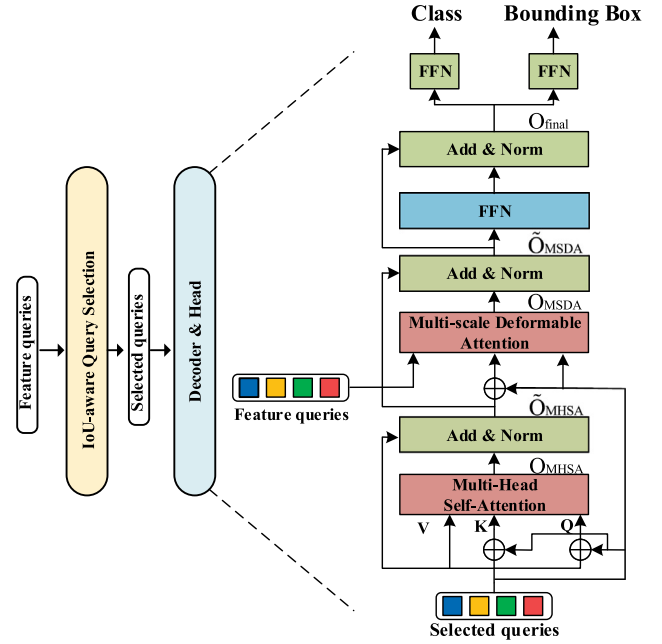


Fig. 6. Head of YOLO-TDH.

of true object locations, thereby improving the robustness of the final predictions.

The selected queries Q_{selected} are fed into the decoder module to represent potential object locations, enabling a multi-query strategy for simultaneous multi-object detection. During decoding, the detection head incorporates both MHSA and MSDA mechanisms to strengthen feature interdependencies. The MHSA mechanism enhances representational capacity by modeling global feature dependencies, which can be computed as:

$$\begin{cases} \text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \cdot V, \\ \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V), \\ O_{\text{MHSA}} = \text{Concat}(\text{head}_1, \dots, \text{head}_h) \cdot W^O, \end{cases} \quad (2)$$

where the matrices of query (Q), key (K), and value (V) are derived from linear projections of Q_{selected} :

$$\begin{cases} Q = Q_{\text{selected}} \cdot W_Q, \\ K = Q_{\text{selected}} \cdot W_K, \\ V = Q_{\text{selected}} \cdot W_V, \end{cases} \quad (3)$$

where W_Q , W_K , and W_V are learnable projection matrices with dimension d_k , W_i^Q , W_i^K , and W_i^V are the weights for the i th of h heads, and W^O is the output projection matrix.

The MSDA mechanism performs adaptive feature sampling across the multi-scale feature map $F_{\text{multi}} \in \mathbb{R}^{H \times W \times C}$ through learnable spatial offsets, enabling dynamic focus on variably-sized object features. The operation is formally defined as:

$$O_{\text{MSDA}} = \sum_{m=1}^M \sum_{p \in \mathcal{P}} w_m \cdot F_{\text{multi}} \cdot (p + \Delta p_m), \quad (4)$$

where M is the number of sampling points for each reference point $p \in \mathcal{P}$, w_m is the learned attention weight of the m th point, and Δp_m is its learnable spatial offset.

The decoder's feature integration pipeline concludes by combining the outputs of MSDA and MHSA through a residual connection followed by layer normalization:

$$\tilde{O}_{\text{MSDA}} = \text{LayerNorm}(\tilde{O}_{\text{MHSA}} + O_{\text{MSDA}}). \quad (5)$$

This integrated result is then passed through a Feed-Forward Network (FFN), which applies a nonlinear transformation to further refine the feature representation:

$$\text{FFN}(x) = \text{ReLU}(x \cdot W_1 + b_1) \cdot W_2 + b_2, \quad (6)$$

where W_1, W_2 , and b_1, b_2 are learnable weights and biases, respectively. The output of the FFN is integrated via a final residual connection and normalization step, producing the refined query features, O_{final} , for that decoder layer:

$$O_{\text{final}} = \text{LayerNorm}(\tilde{O}_{\text{MSDA}} + \text{FFN}(\tilde{O}_{\text{MSDA}})). \quad (7)$$

The final query features, which encode rich semantic and spatial information, are then passed to parallel prediction heads to generate the final class probabilities and bounding box coordinates as shown in the upper right of Fig. 6.

3.4. Loss function of YOLO-TDH

The final output features from the decoder head, O_{final} , are processed by two parallel prediction heads: a classification head and a bounding box regression head. The classification head computes class probabilities, $P_{\text{class}} \in \mathbb{R}^{N \times C}$, for C categories via a linear layer followed by a softmax activation:

$$P_{\text{class}} = \text{softmax}(O_{\text{final}} W_{\text{class}} + b_{\text{class}}), \quad (8)$$

where $W_{\text{class}} \in \mathbb{R}^{N \times C}$ and $b_{\text{class}} \in \mathbb{R}^C$ are learnable parameters. The model is trained by VariFocal Loss (VFL) [40], which mitigates class imbalance by down-weighting the loss from well-classified examples and focusing on hard-to-classify samples.

The bounding box regression head predicts the coordinates, $B_{\text{box}} \in \mathbb{R}^{N \times 4}$, using a separate linear layer:

$$B_{\text{box}} = O_{\text{final}} W_{\text{box}} + b_{\text{box}}, \quad (9)$$

where $W_{\text{box}} \in \mathbb{R}^{D \times 4}$ and $b_{\text{box}} \in \mathbb{R}^4$ are learnable parameters. The total regression loss, $\mathcal{L}_{\text{box}}$, is a weighted sum of the ShapeIoU loss [41] and a standard L_1 loss [42]:

$$\mathcal{L}_{\text{box}} = \lambda_{\text{IoU}} \mathcal{L}_{\text{ShapeIoU}}(B^{\text{pred}}, B^{\text{gt}}) + \lambda_{L_1} \mathcal{L}_{L_1}(B^{\text{pred}}, B^{\text{gt}}), \quad (10)$$

where B^{pred} and B^{gt} are the predicted and ground-truth boxes, respectively, and λ_{IoU} and λ_{L_1} are balancing coefficients.

The $\mathcal{L}_{\text{ShapeIoU}}$ term is a composite metric designed to enhance regression accuracy by penalizing discrepancies in boundary overlap, center position, and object shape, with its formal definition given by

$$\mathcal{L}_{\text{ShapeIoU}} = \mathcal{L}_{\text{IoU}} + \Delta_{\text{shape}} + 0.5 \times \Omega_{\text{shape}}, \quad (11)$$

where the IoU loss [41], $\mathcal{L}_{\text{IoU}} = 1 - \text{IoU}$, measures the overlap between the predicted box B and the ground-truth box B^{gt} . The shape-sensitive distance term penalizes center point deviations, normalized by the scale and aspect ratio of the ground-truth box:

$$\Delta_{\text{shape}} = h_h \frac{(x - x^{\text{gt}})^2}{c^2} + w_w \frac{(y - y^{\text{gt}})^2}{c^2}, \quad (12)$$

where $c = \sqrt{(w^{\text{gt}})^2 + (h^{\text{gt}})^2}$, and the weights h_h and w_w are functions of the ground-truth box's aspect ratio, expressed by

$$w_w = \frac{2 \times (w^{\text{gt}})^{\gamma}}{(w^{\text{gt}})^{\gamma} + (h^{\text{gt}})^{\gamma}}, \quad (13)$$

$$h_h = \frac{2 \times (h^{\text{gt}})^{\gamma}}{(w^{\text{gt}})^{\gamma} + (h^{\text{gt}})^{\gamma}}, \quad (14)$$

where γ is an adjustable parameter that is set to 1 in this paper.

The shape penalty term, Ω_{shape} , quantifies the discrepancy in aspect ratio between the predicted and ground-truth boxes:

$$\Omega_{\text{shape}} = \left(1 - e^{-\frac{w_w |w - w^{\text{gt}}|}{\max(w, w^{\text{gt}})}} \right)^{\theta} + \left(1 - e^{-\frac{h_h |h - h^{\text{gt}}|}{\max(h, h^{\text{gt}})}} \right)^{\theta}, \quad (15)$$

where θ is an adjustable parameter which is set to 4.



Fig. 7. Example images in the TLD-Det dataset.

Remark 3. It should be pointed out that, by integrating VariFocal loss for classifier optimization and enhanced ShapeIoU loss for bounding box regression, the YOLO-TDH model achieves more precise detection of power transmission line equipment in complex backgrounds, while strengthening its adaptability to scale variations and occlusions in practical applications.

4. Experimental results

4.1. Dataset description

4.1.1. TLD-Det

To facilitate the performance evaluation, a specialized dataset for Transmission Line Device Detection (TLD-Det) is constructed in this study. This dataset is specifically compiled to address the challenges posed by visually ambiguous components and their operational states in overhead power line environments. The dataset is annotated following the YOLO format, and representative samples are illustrated in Fig. 7.

The dataset encompasses 11 distinct object classes, covering key components and their conditions, including three operational states for markers and insulators (intact, suspect, and damaged), the presence of bird nests, and two conditions for dampers and spacers (intact and damaged). The distribution of labels is shown in Fig. 8. For model training and validation, the dataset is split into a training set (5667 images) and a combined test/validation set (628 images).

The image data is acquired via unmanned aerial vehicles (UAVs) equipped with high-definition transmission systems, minimizing image distortions during flight. To ensure feature-learning quality, we implement a rigorous selection pipeline that automatically filters out-of-focus images, resulting in a refined dataset of consistently high-resolution samples. The resulting TLD-Det dataset serves two key purposes: (1) providing reliable training data for automated detection models, and (2) establishing a standardized benchmark for future method evaluation.

4.1.2. Insulator

To further evaluate the model's capability in identifying fine-grained component faults, an additional Insulator Defect Dataset is introduced [43]. As indicated by Fig. 9, this dataset focuses exclusively on insulator conditions, categorized into two distinct classes: normal insulators and defective insulators. The dataset is strictly partitioned into a training set of 2386 images, a validation set of 341 images, and a test set of 682 images.

Although this dataset introduces specific visual challenges—such as extreme aspect ratios and severe class imbalance, as depicted in Fig. 10, it is inherently less complex and lacks the diverse multi-object co-occurrences typical of actual power transmission line inspections. Consequently, rather than performing exhaustive baseline comparisons, we utilize this dataset exclusively for a targeted verification of the proposed YOLO-TDH algorithm.

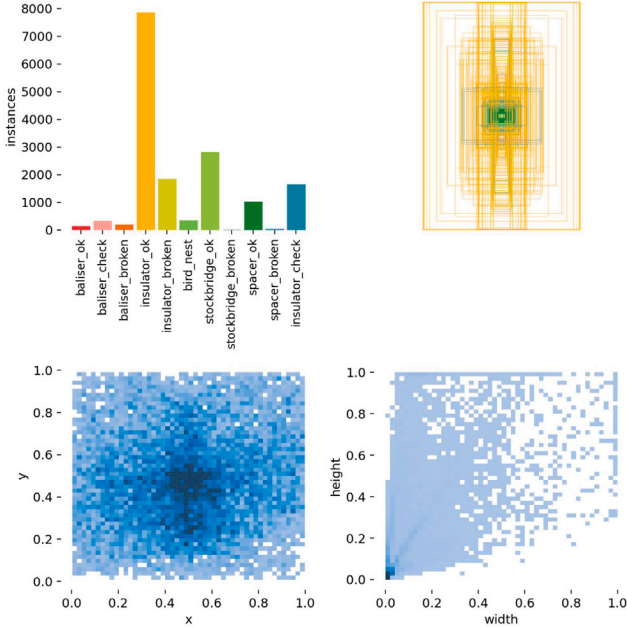


Fig. 8. Label distribution in the TLD-Det dataset.

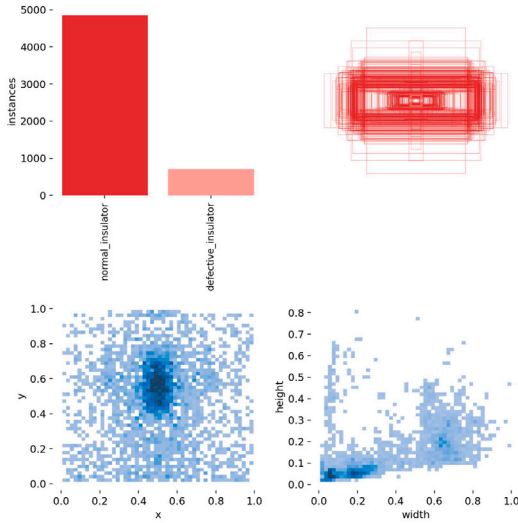


Fig. 9. Label distribution in the insulator dataset.



Fig. 10. Example images in the insulator dataset.

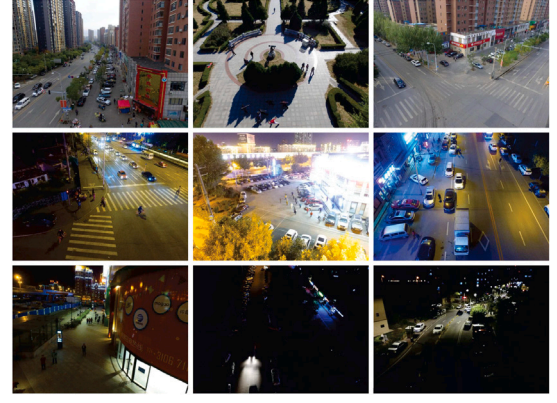


Fig. 11. Example images in the VisDrone dataset.

4.1.3. VisDrone2019

VisDrone2019 is selected as the public benchmark to rigorously validate the generalization capability of the proposed method. Unlike existing public insulator datasets—which often feature close-up shots and simple backgrounds—VisDrone2019 shares the fundamental visual challenges of our TLD-Det dataset, namely high-altitude UAV perspectives, drastic scale variations, extreme small objects, and complex background interference [44]. Evaluating on this comprehensive dataset demonstrates that our architectural innovations are robust across demanding UAV vision tasks rather than overfitted to a specific insulator domain. Experimentally, 6471 training images across all predefined categories are utilized for comprehensive feature learning, while the validation set monitors model convergence. The final evaluation is conducted on the 1610 test images, which provide a rigorous benchmark featuring densely packed and heavily occluded instances that accurately reflect real-world UAV deployment conditions, as depicted in Fig. 11.

4.2. Experiment configuration and evaluation metrics

4.2.1. Experiment configuration

The environmental setup for this experiment is detailed in Table 1. The hardware configuration includes a Windows 10 Professional operating system, an AMD Ryzen 7 5700X 8-core processor, and an NVIDIA GeForce RTX 3060 GPU (12 GB VRAM). The software environment is configured with Python 3.8.18 and the PyTorch 2.0.1 deep learning framework. For accelerated computation, the CUDA 11.8 platform is utilized. The input images are resized to 640×640 pixels, and the batch size is set as 8. The training process employs an initial learning rate of 0.0001 with a momentum of 0.9 and runs for a total of 200 epochs, with a scheduler adjusting the learning rate during training. Online data augmentation strategies, including scaling (0.5), random horizontal flipping ($p=0.5$), and translation (0.1), are also applied.

4.2.2. Evaluation metrics

The performance of the proposed model is comprehensively evaluated using a suite of standard metrics targeting detection accuracy, inference speed, and computational efficiency. Detection accuracy is primarily quantified by Precision (P) and Recall (R). Precision is defined as the ratio of true positive detections to the total number of predicted positives ($P = \frac{TP}{TP+FP}$), while Recall measures the ratio of true positives to the total number of ground-truth instances ($R = \frac{TP}{TP+FN}$). The primary composite metric for performance evaluation is mean Average Precision (mAP). Specifically, we report mAP@0.5, calculated at a fixed Intersection over Union (IoU) threshold of 0.5, and mAP@0.5:0.95, which provides a more stringent evaluation by averaging mAP values across IoU thresholds from 0.5 to 0.95 in increments of

Table 1
Experimental environment.

Parameter	Set up
Epochs	200
Batch Size	8
Initial Learning rate	0.0001
Final Learning Rate	1.0
Momentum	0.9
Image Scale	0.5
Image Flip Left-Right	0.5
Image Translate	0.1
CPU	AMD Ryzen7 5700X 8-Core Processor 3.40 GHz
GPU	NVIDIA GeForce RTX 3060 12G
Framework	Pytorch 2.0.1
Integrated Development Environment	Python3.8.18 CUDA 11.8 CUDNN8.9.4
Operating System	Windows 10 Professional

0.05. The IoU metric itself quantifies the overlap between a predicted and a ground-truth bounding box; a detection is classified as a true positive if its IoU exceeds the specified threshold.

Model efficiency is assessed from three perspectives. Inference speed is measured in Frames Per Second (FPS), where a higher value indicates superior real-time performance. Computational complexity is evaluated using Floating Point Operations (FLOPs), representing the total calculations in a single forward pass; a lower FLOPs count signifies greater computational efficiency. Finally, model size is quantified by the number of trainable Parameters (Params), which directly correlates with the model's memory consumption.

4.3. Ablation experiment

To rigorously evaluate the individual contributions and synergistic effects of the proposed components — namely, the EFIM, EFFM, Decoder, and ShapeIoU — an ablation study has been conducted by sequentially integrating these modules into the baseline architecture, as presented in Table 2. All experimental configurations have been trained and evaluated under identical conditions to ensure fair comparisons. As detailed in Table II, the unmodified baseline model has yielded an mAP@0.5 of 83.7% and an mAP@0.5:0.95 of 59.4%, requiring a computational overhead of 11.13M parameters and 28.5G FLOPs.

The initial integration of the EFIM module has demonstrated a distinct dual advantage: it has enhanced the mAP@0.5 and mAP@0.5:0.95 to 84.2% and 60.0%, respectively, while simultaneously reducing the parameter count to 9.13M and FLOPs to 23.1G. This has indicated that the EFIM efficiently optimizes feature extraction while mitigating computational redundancy. Building upon this, the subsequent incorporation of the EFFM module has further compressed the network. The parameter count has experienced a severe reduction to a minimum of 6.47M, and the FLOPs have incrementally decreased to 22.6G. Concurrently, the model has achieved marginal improvements in both mAP@0.5 (84.4%) and mAP@0.5:0.95 (60.1%). This phase of the ablation study has underscored the pivotal role of the proposed feature fusion mechanisms in forging a highly lightweight architecture.

Following the establishment of this efficient structural foundation, the Decoder module has been introduced. This integration has substantially boosted the model's target resolution and classification capabilities, evidenced by a dramatic surge in Precision to 91.7% and an increase in mAP@0.5 to 84.9%. However, this enhancement has involved a structural and performance trade-off; the parameter count has risen back to 9.07M, and the stricter mAP@0.5:0.95 metric has temporarily regressed to 59.3%.

To rectify this regression and further refine bounding box predictions, the ShapeIoU loss function has ultimately been applied. Because it has been exclusively utilized during the training phase to guide gradient descent, ShapeIoU has introduced zero additional cost during inference, maintaining the parameters at 9.07M and FLOPs at 22.5G. Its application has effectively recovered the mAP@0.5:0.95 to its previous

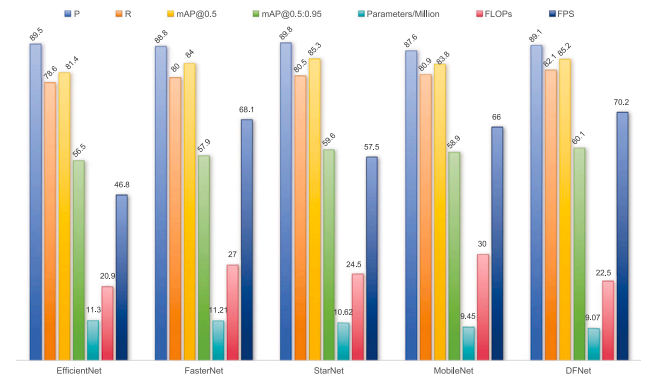


Fig. 12. Comparison of different backbone on TLD-Det dataset.

peak of 60.1% and has pushed the mAP@0.5 to an overall maximum of 85.2%, alongside a notable improvement in Recall to 82.1%.

Collectively, the complete integration of all four components has yielded a highly optimized model that has outperformed the baseline by 1.5 percentage points in mAP@0.5 and 0.7 percentage points in mAP@0.5:0.95. Furthermore, it has achieved a substantial 18.5% reduction in total parameters and a 21.1% decrease in FLOPs. These results have validated the proposed method's capacity to deliver superior detection accuracy while strictly adhering to lightweight computational constraints.

4.4. Comparison experiments

4.4.1. Comparison of different networks

Given the foundational role of the backbone network in overall model performance, we conduct a comparative analysis to evaluate the proposed DFNet against several mainstream lightweight backbones. For this evaluation, the backbone within the YOLO-TDH model is systematically replaced with four prominent lightweight architectures: EfficientNet [45], FasterNet [46], StarNet [36], and MobileNet [47]. Each resulting model variant is subsequently benchmarked on the TLD-Det dataset. The architectural specifications and comparative results are detailed in Table 3 and Fig. 12.

A comprehensive analysis of the experimental results, presented in Table 3, demonstrates the competitive performance of the proposed DFNet backbone, which achieves an excellent balance between accuracy and efficiency. The proposed DFNet achieves a mAP@0.5 of 85.2% and mAP@0.5:0.95 of 60.1%. While slightly lower than StarNet in mAP@0.5, it surpasses StarNet in mAP@0.5:0.95 by 0.5%. This performance is complemented by superior efficiency: DFNet utilizes the fewest parameters (9.07M) among all tested networks and maintains a competitive computational load (22.5G FLOPs). Furthermore, DFNet demonstrates the highest inference speed at 70.2 FPS, which is 22.1%

Table 2
Results of ablation experiments.

EFIM	EFFM	Decoder	ShapeIoU	P(%)	R(%)	mAP@0.5(%)	mAP@0.5:0.95(%)	Params(M)	FLOPs(G)
				84.2	77.0	83.7	59.4	11.13	28.5
✓				84.3	77.6	84.2	60.0	9.13	23.1
✓	✓			85.5	79.0	84.4	60.1	6.47	22.6
✓		✓		86.8	82.1	84.0	57.6	9.96	23.2
✓		✓	✓	89.4	81.4	84.3	58.7	9.96	23.2
	✓	✓		87.4	83.0	84.5	59.1	9.96	24.9
	✓	✓	✓	87.7	81.5	84.6	59.1	9.96	24.9
✓	✓	✓		91.7	81.5	84.9	59.3	9.07	22.5
✓	✓	✓	✓	89.1	82.1	85.2	60.1	9.07	22.5

Table 3
Comparison of different backbones.

Network	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Params (M)	FLOPs (G)	FPS
EfficientNet	89.5	78.6	81.4	56.5	11.30	20.9	46.8
FasterNet	88.8	80.0	84.0	57.9	11.21	27.0	68.1
StarNet	89.8	80.5	85.3	59.6	10.62	24.5	57.5
MobileNet	87.6	80.9	83.8	58.9	9.45	30.0	66.0
DFNet	89.1	82.1	85.2	60.1	9.07	22.5	70.2

Table 4
Comparison of different loss function.

IoU	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
CloU	89.3	81.5	84.9	59.3
EIoU	90.5	81.3	84.6	58.9
GIoU	88.9	81.8	84.0	59.1
Siou	87.6	81.5	84.4	59.3
Diou	88.3	82.4	84.7	60.1
ShapeIoU	89.1	82.1	85.2	60.1

faster than StarNet and 3.1% faster than the next fastest model, FasterNet. Collectively, these results highlight the exceptional balance of the DFNet architecture. Compared to StarNet, DFNet reduces the parameter count by 14.6% and FLOPs by 8.2%. When compared to MobileNet, the model with the second-lowest parameter count, DFNet achieves a 1.4% increase in mAP@0.5 while being 4.0% smaller and requiring 25% fewer FLOPs. These findings highlight the efficacy of DFNet in establishing a new state-of-the-art in efficiency–accuracy trade-offs for this task.

4.4.2. Comparison of different loss functions

A comparative analysis is conducted to validate the efficacy of the ShapeIoU loss function [41] against several widely used alternatives, given the critical role of the loss function in model optimization. For this evaluation, the ShapeIoU component within the YOLO-TDH network is systematically replaced with five other loss functions: CloU [48], EIoU [49], Giou [50], Siou [51], and Diou [52]. Each resulting model variant is then evaluated on the TLD-Det dataset. The results of this comparative experiment are detailed in Table 4 and Fig. 13.

The results demonstrate that the ShapeIoU-equipped model achieves a competitive Precision at 89.1%, surpassing models using Siou (87.6%) and Diou (88.3%). While slightly lower than CloU (89.3%), this high precision indicates a low false positive rate. This performance suggests that ShapeIoU maintains a strong ability to identify components like insulators and wire joints, particularly in challenging scenarios involving occlusions or variable lighting conditions. Notably, its Recall of 82.1% is higher than that of CloU (81.5%) and Siou (81.5%), achieving a superior balance between high precision and recall, and effectively minimizing the risk of missed detections (false negatives). This equilibrium is crucial for ensuring comprehensive detection of multi-scale devices across the transmission line. In terms of mAP, the model with ShapeIoU achieves the highest mAP@0.5 of 85.2%. This surpasses Siou by 0.8% and CloU by 0.3%, while also outperforming EIoU, highlighting its

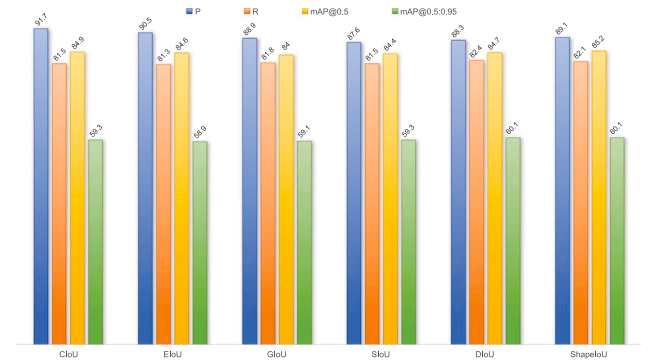


Fig. 13. Comparison of different loss functions on TLD-Det dataset.

strong performance at a moderate IoU threshold. Furthermore, its mAP@0.5:0.95 reaches 60.1%, outperforming EIoU by 1.2% and CloU by 0.8%, while achieving comparable performance to Diou. This result indicates superior robustness and localization accuracy across stricter IoU thresholds. Collectively, these results demonstrate that ShapeIoU more effectively models object boundaries and shapes, making it particularly suitable for detecting small, multi-scale targets (e.g., insulators) against the complex backgrounds characteristic of transmission line environments.

4.4.3. Comparison with different object detectors on TDL-det

To validate the performance of the proposed model, a comparative analysis is conducted against seven state-of-the-art object detection models: YOLOv5s [25], YOLOv6s [27], YOLOv8s [31], YOLOXs [26], Gold-YOLOs [30], Damo-YOLOs [29], and YOLOv10s [33]. The comparative results on the dedicated transmission line device dataset are presented in Table 5.

As demonstrated in Table 5 and Fig. 14, the proposed model achieves the highest detection accuracy among all compared lightweight YOLO variants. Specifically, our model attains a Precision (P) of 89.1%, outperforming the baseline YOLOv8s (84.2%) by 4.9 percentage points and the recent YOLOv10 (81.7%) by a significant 7.4 percentage points. While RT-DETR achieves a slightly higher recall (82.8% vs. 82.1%) and mAP@0.5:0.95 (60.4% vs. 60.1%), our method surpasses it in Precision (89.1% vs. 87.6%) and mAP@0.5 (85.2% vs. 85.0%).

Crucially, the proposed method achieves this high accuracy while maintaining a highly efficient architecture. With only 9.07M parameters and 22.5G FLOPs, our model is substantially more lightweight than

Table 5
Comparison of different object detectors on TLD-Det dataset.

Method	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Params (M)	FLOPs (G)	FPS
YOLOv5s	83.2	75.8	82.4	54.0	5.99	12.7	120.1
YOLOv6s	82.9	77.3	81.5	58.4	16.31	44.3	126.9
YOLOv8s	84.2	77.0	83.7	59.4	11.13	28.5	153.6
YOLOXs	83.2	76.4	82.7	58.0	9.11	23.8	150.2
Gold-YOLO	82.6	74.0	82.7	58.8	20.14	47.0	99.1
Damo-YOLO	81.9	77.5	81.7	58.4	15.33	31.9	109.1
YOLOv10	81.7	78.0	83.9	59.3	7.22	21.4	145.1
RT-DETR	87.6	82.8	85	60.4	19.88	58.6	56.7
Ours	89.1	82.1	85.2	60.1	9.07	22.5	70.2

Table 6
Detection results on the Insulator Defect Dataset.

Class	P(%)	R(%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
All	98.5	91.4	95.9	85.6
normal_insulator	97.4	91.9	96.5	87.9
defective_insulator	99.6	90.9	95.3	83.4

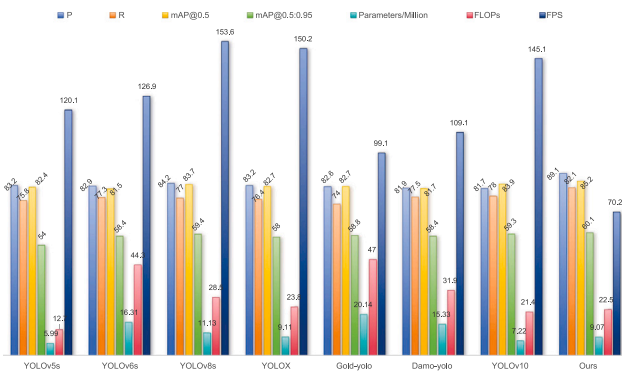


Fig. 14. Comparison of different object detectors on TLD-Det dataset.

other high-performing models. For instance, compared to RT-DETR, which requires 19.88M parameters and 58.6G FLOPs, our model reduces the parameter count by over 50% and the computational load by more than 60%, while still achieving comparable or superior accuracy metrics. Furthermore, our model maintains a competitive inference speed of 70.2 FPS, ensuring its suitability for real-time applications.

In summary, the comparative analysis confirms that the proposed method achieves an optimal balance between detection accuracy and computational efficiency. The significant improvements in precision and mAP, achieved with a minimal parameter footprint, demonstrate the effectiveness of our architectural innovations for accurate and efficient object detection in transmission line inspection scenarios.

4.4.4. Detection results on insulator dataset

As previously discussed, due to its relative simplicity compared to actual multi-object transmission line scenes, we utilize this dataset exclusively for a targeted verification of the proposed YOLO-TDH algorithm in addressing specific visual challenges. The quantitative detection results are summarized in Table 6.

As presented in Table 6, the proposed YOLO-TDH model demonstrates exceptional performance, achieving an overall Precision of 98.5%, a Recall of 91.4%, and a remarkably high mAP@0.5 of 95.9%. Most notably, the results effectively alleviate the concerns regarding the severe class imbalance inherent in the dataset. Despite being the extreme minority class, the defective insulator category achieves an outstanding Precision of 99.6% and an mAP@0.5 of 95.3%. This near-perfect precision indicates that the model produces almost no false positive alarms when identifying critical defects, a characteristic that is highly desirable in practical power grid safety inspections.

Furthermore, the stringent mAP@0.5:0.95 metric for defective insulators reaches 83.4%. This robust localization accuracy strongly proves that the specific architectural designs of YOLO-TDH. The results convincingly validate that YOLO-TDH possesses an excellent capability for fine-grained feature representation and precise bounding box regression, rather than merely overfitting to the majority class.

4.4.5. Comparison with different object detectors on VisDrone2019

To comprehensively evaluate the generalization capability and robustness of the proposed method in complex UAV-view scenarios, we benchmark YOLO-TDH against state-of-the-art lightweight object detectors on the public VisDrone2019 dataset. The comparison includes widely adopted models evaluated in previous sections: YOLOv5s, YOLOv6s, YOLOv8s, YOLOXs, Gold-YOLO, Damo-YOLO, and YOLOv10s. The quantitative results are summarized in Table 7.

As presented in Table 7, YOLO-TDH demonstrates exceptional generalization capabilities on the challenging VisDrone2019 benchmark, which is characterized by dense object clusters and extreme scale variations. Compared to the baseline YOLOv8s, our proposed model achieves a substantial improvement in Precision by 9.6 percentage points (from 44.0% to 53.6%), alongside increases of 2.2% and 3.4% in mAP@0.5 and mAP@0.5:0.95, respectively. Furthermore, YOLO-TDH significantly outperforms other mainstream lightweight YOLO variants (such as YOLOv6s, YOLOXs, Gold-YOLO, and YOLOv10s) across all accuracy metrics, proving its robustness in highly cluttered, non-industrial UAV environments.

It is worth noting that while the Transformer-based RT-DETR model yields marginally higher mAP scores, it does so at a severe computational cost. Specifically, YOLO-TDH operates with less than half the parameters (9.07M vs. 19.88M) and FLOPs (22.5G vs. 58.6G) of RT-DETR. Moreover, YOLO-TDH achieves a high inference speed of 130 FPS, making it nearly 2.6 times faster than RT-DETR (48.9 FPS). This confirms that the YOLO-TDH module successfully achieves an optimal accuracy-efficiency trade-off for real-time UAV deployments.

4.5. Visualization results in different environments

4.5.1. Visualization results in TLD-Det

To provide a qualitative assessment of the model's performance in complex, real-world scenarios, a visual comparison is conducted against seven state-of-the-art methods: YOLOv5, YOLOv6, YOLOX, Damo-YOLO, Gold-YOLO, YOLOv8, and YOLOv10. This evaluation, illustrated in Figs. 15 and 16, is performed on challenging aerial scenes characterized by significant background clutter, multi-scale targets, and dense object arrangements, which present substantial challenges for accurate detection.

This visual analysis highlights the proposed model's superior performance in both detection completeness and localization precision. In terms of detection completeness, the model successfully identifies all key components, whereas competing methods such as Damo-YOLO and Gold-YOLO exhibit significant false negatives by failing to detect smaller targets. With respect to localization precision, multiple models, including YOLOv5, YOLOv6, and YOLOv8, generate imprecise, oversized bounding boxes around an insulator string. In contrast, our model

Table 7
Experimental results on VisDrone Dataset.

Method	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Params (M)	FLOPs (G)	FPS
v5s	33.4	28.1	12.1	23.6	142	5.9	12.7
v6s	43.7	32.3	17.5	30.8	96.2	16.31	44
v8s	44	34.1	18.2	32	118	11.1	28.5
xs	43.6	32.5	17.6	31.1	127	9.11	23.8
gold	44.9	34.4	18.8	32.7	83.3	20.14	47
damo	44.7	33.8	18.6	32.3	101	15.33	31.9
v10s	43.5	34.1	18.3	32.2	132	7.22	21.4
RT-DETR	55	38.7	21.6	37.5	48.9	19.88	58.6
yolo-tdh	53.6	36.8	20.4	35.4	130	9.07	22.5

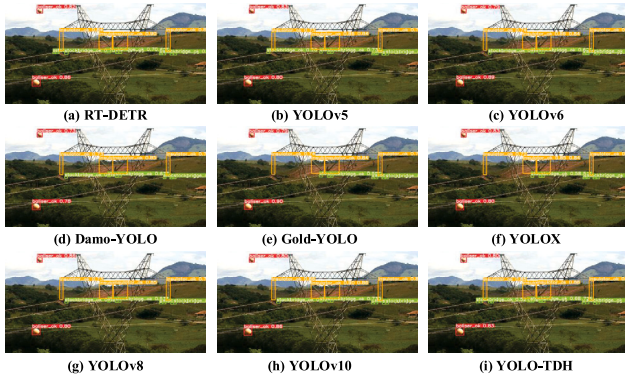


Fig. 15. Comparison of detection results in complex backgrounds.

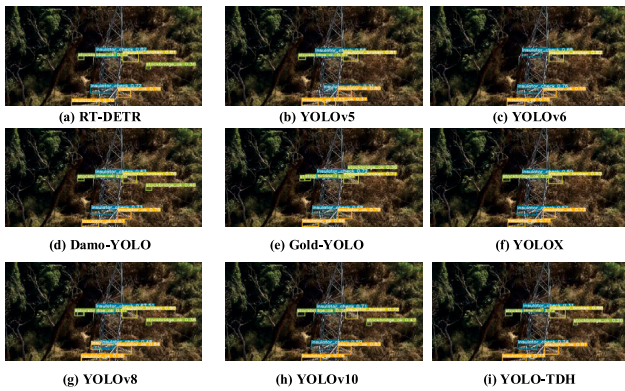


Fig. 16. Comparison of detection results for dense objects.

produces a precise bounding box that tightly conforms to the object's true contours. These qualitative results demonstrate the model's distinct capability to simultaneously mitigate both false negatives and inaccurate localization, confirming its superior reliability for industrial applications that require fine-grained analysis.

4.5.2. Visualization results in insulator

To provide a qualitative assessment of the proposed YOLO-TDH model's performance in fine-grained fault detection, the visualization results on the Insulator Defect Dataset are presented in Fig. 17.

As evidenced by the visual results, the model maintains consistently high prediction confidence scores—typically reaching 0.98 for normal insulators and exceeding 0.91 for fine-grained defects. These visual findings directly corroborate the high precision metrics reported in the prior quantitative analysis.

4.5.3. Visualization results in VisDrone2019

To qualitatively validate the generalization capability of the proposed model in highly complex, non-industrial scenarios, visualization results on the challenging VisDrone2019 public benchmark are



Fig. 17. Detection results in insulator dataset.

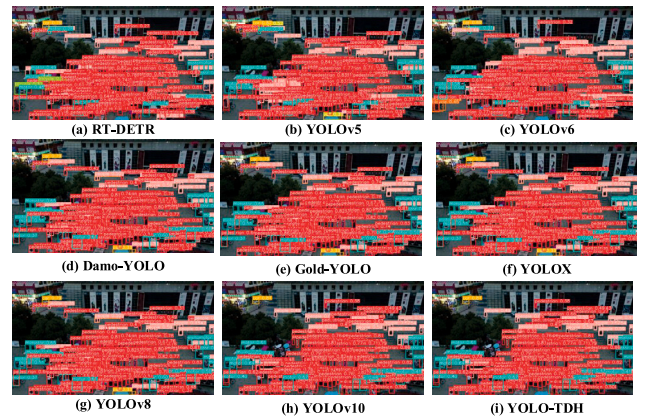


Fig. 18. Detection results in VisDrone dataset.

presented in Fig. 18. This visual comparison benchmarks YOLO-TDH against eight state-of-the-art detectors.

As depicted in the rendered scenes, the selected urban environment presents extreme visual challenges, characterized by severely occluded pedestrians, massive scale variations, and dense object packing under complex lighting conditions. Under such demanding circumstances, several baseline models exhibit noticeable limitations. For instance, methods like YOLOv5, YOLOv6, and Gold-YOLO suffer from evident missed detections (false negatives) for ultra-small targets in the distant background and generate overlapping, imprecise bounding boxes in heavily crowded areas. In contrast, the proposed YOLO-TDH demonstrates superior robustness and perceptual accuracy. It successfully distinguishes individual pedestrians within dense clusters and maintains tight bounding box localization without being overwhelmed by the background clutter.

5. Conclusion

This study has presented YOLO-TDH, a novel object detection model derived from the YOLOv8 architecture and specifically optimized for transmission line device detection. The proposed model, which synergistically integrates the EFIM, the EEFM, a new Decoder, and the ShapeIoU loss, sets a new state-of-the-art on the dedicated TLD-Det dataset. YOLO-TDH has surpassed existing models across key metrics, including Precision, Recall, mAP@0.5, and the more stringent mAP@0.5:0.95, while maintaining a lightweight profile in terms of parameter count and computational load. Furthermore, its strong performance on the insulator dataset and public VisDrone2019 benchmark validates its robust generalization capabilities. These results confirm that YOLO-TDH achieves an exceptional balance between high accuracy and operational efficiency, demonstrating significant potential for robust, real-time deployment in resource-constrained industrial environments.

Future research should further explore the application and optimization of YOLO-TDH in embedded devices to meet the demands of more complex industrial scenarios [5,53,54], thereby promoting its practical deployment and development in intelligent detection systems.

CRedit authorship contribution statement

Baoye Song: Writing – original draft, Resources, Project administration, Methodology, Formal analysis, Data curation, Conceptualization. **Shihao Zhao:** Writing – original draft, Visualization, Software, Methodology, Formal analysis, Data curation. **Weibo Liu:** Writing – review & editing, Validation, Supervision, Methodology, Conceptualization. **Yani Xue:** Writing – review & editing, Visualization, Validation, Software, Formal analysis.

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

- [1] H. Liang, C. Zuo, W. Wei, Detection and evaluation method of transmission line defects based on deep learning, *IEEE Access* 8 (2020) 38448–38458.
- [2] H. Fathabadi, Impact of high-voltage power transmission lines on photovoltaic power production, *Sol. Energy* 163 (2018) 78–83.
- [3] F. Deng, Z. Xie, W. Mao, B. Li, Y. Shan, B. Wei, H. Zeng, Research on edge intelligent recognition method oriented to transmission line insulator fault detection, *Int. J. Electr. Power Energy Syst.* 139 (2022) 108054.
- [4] Y. Luo, X. Yu, D. Yang, B. Zhou, A survey of intelligent transmission line inspection based on unmanned aerial vehicle, *Artif. Intell. Rev.* 56 (1) (2023) 173–201.
- [5] N. Ekren, Z. Karagoz, M. Sahin, A review of line suspended inspection robots for power transmission lines, *J. Electr. Eng. Technol.* 19 (4) (2024) 2549–2583.
- [6] P. Zhang, Z. Zhang, Y.P. Hao, Z.H. Zhou, B. Luo, T. Wang, Multi-scale feature enhanced domain adaptive object detection for power transmission line inspection, *IEEE Access* 8 (2020) 182105–182116.
- [7] Y. Zhang, J. Zhang, J. Zhou, Urban traffic tiny object detection via attention and multi-scale feature driven in UAV-vision, *Sci. Rep.* 14 (1) (2024) 20614.
- [8] Q. Ma, D. Wu, X. Luo, A review of deep learning-based power load forecasting methods, *Int. J. Netw. Dyn. Intell.* 4 (4) (2025) 100027.
- [9] W. Zhao, M. Xu, X. Cheng, Z. Zhao, An insulator in transmission lines recognition and fault detection model based on improved faster RCNN, *IEEE Trans. Instrum. Meas.* 70 (2021) 5016408.
- [10] P. Luo, B. Wang, H. Wang, F. Ma, H. Ma, L. Wang, An ultrasmall bolt defect detection method for transmission line inspection, *IEEE Trans. Instrum. Meas.* 72 (2023) 5006512.
- [11] D. Wu, W. Yang, J. Li, K. Du, L. Li, Z. Yang, CRL-YOLO: A comprehensive recalibration and lightweight detection model for AAV power line inspections, *IEEE Trans. Instrum. Meas.* 74 (2025) 3533721.
- [12] Y. Wang, Q. Li, B. Chen, Image classification towards transmission line fault detection via learning deep quality-aware fine-grained categorization, *J. Vis. Commun. Image Represent.* 64 (2019) 102647.
- [13] K. Tong, Y. Wu, F. Zhou, Recent advances in small object detection based on deep learning: A review, *Image Vis. Comput.* 97 (2020) 103910.
- [14] Z. Qiu, X. Zhu, C. Liao, D. Shi, W. Qu, Detection of transmission line insulator defects based on an improved lightweight YOLOv4 model, *Appl. Sci.* 12 (3) (2022) 1207.
- [15] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, End-to-end object detection with transformers, in: *Proceedings of the 16th European Conference on Computer Vision*, Glasgow, UK, 2020, pp. 213–229.
- [16] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, J. Chen, Detsr beat yolos on real-time object detection, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, USA, 2024, pp. 16965–16974.
- [17] Y. Zhang, C. Wu, T. Zhang, Y. Liu, Y. Zheng, Self-attention guidance and multi-scale feature fusion-based UAV image object detection, *IEEE Geosci. Remote. Sens. Lett.* 20 (2023) 6004305.
- [18] H. Yi, B. Liu, B. Zhao, E. Liu, Small object detection algorithm based on improved YOLOv8 for remote sensing, *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.* 17 (2024) 1734–1747.
- [19] N. Jain, J. Bedi, A. Anand, S. Godara, A transfer learning architecture to detect faulty insulators in powerlines, *IEEE Trans. Power Deliv.* 39 (2) (2024) 1002–1011.
- [20] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, USA, 2016, pp. 779–788.
- [21] J. Redmon, A. Farhadi, YOLO9000: Better, faster, stronger, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, 2017, pp. 7263–7271.
- [22] J. Redmon, A. Farhadi, YOLOv3: An incremental improvement, 2018, *arXiv:1804.02767*.
- [23] T.Y. Lin, P. Dollar, D. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, 2017, pp. 2117–2125.
- [24] A. Bochkovskiy, C.Y. Wang, H.Y.M. Liao, YOLOv4: Optimal speed and accuracy of object detection, 2020, *arXiv:2004.10934*.
- [25] G. Jocher, C. Liu, H. Adam, L. Yu, P. Rai changyu98, T. Sullivan, Ultralytics/YOLOv5: Initial release, 2020, <http://dx.doi.org/10.5281/zenodo.3908560>.
- [26] Z. Ge, S. Liu, F. Wang, Z. Li, J. Sun, YoloX: Exceeding yolo series in 2021, 2021, *arXiv:2107.08430*.
- [27] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Wei, Y. Nie, Y. Li, B. Zhang, Y. Liang, L. Zhou, X. Xu, X. Chu, X.M. Wei, X.L. Wei, YOLOv6: A single-stage object detection framework for industrial applications, 2022, *arXiv:2209.02976*.
- [28] C.Y. Wang, A. Bochkovskiy, H.Y.M. Liao, YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Vancouver, Canada, 2023, pp. 7464–7475.
- [29] X. Xu, Y. Jiang, W. Chen, Y. Huang, Y. Zhang, X. Sun, Damo-yolo: A report on real-time object detection design, 2022, *arXiv:2211.15444*.
- [30] C. Wang, W. He, Y. Nie, J. Guo, C. Liu, Y. Wang, K. Han, Gold-YOLO: Efficient object detector via gather-and-distribute mechanism, *Adv. Neural Inf. Process. Syst.* 36 (2023) 51094–51112, Louisiana, USA.
- [31] G. Jocher, YOLOv8, 2023, <https://github.com/ultralytics/ultralytics/tree/main>.
- [32] C. Wang, J. Yeh, H. Liao, YOLOv9: Learning what you want to learn using programmable gradient information, in: *Proceedings of the European Conference on Computer Vision*, Italy, 2024, pp. 1–21.
- [33] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, YOLOv10: Real-time end-to-end object detection, in: *Proceedings of the Conference on Neural Information Processing Systems*, Vancouver, Canada, 2024, pp. 107984–108011.
- [34] R. Khanam, M. Hussain, YOLOv11: An overview of the key architectural enhancements, 2024, *arXiv:2410.17725*.
- [35] Y. Tian, X. Sun, P. Chun, S. Duan, G. Ding, YOLOv12: Attention-centric real-time object detectors, 2025, *arXiv:2502.12524*.
- [36] X. Ma, X. Dai, Y. Bai, Y. Wang, Y. Fu, Rewrite the stars, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, USA, 2024, pp. 5694–5703.
- [37] M. Tan, R. Pang, Q.V. Le, Efficientdet: Scalable and efficient object detection, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Virtual, 2020, pp. 10781–10790.
- [38] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale, 2020, *arXiv:2010.11929*.
- [39] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai, Deformable detr: Deformable transformers for end-to-end object detection, 2020, *arXiv:2010.04159*.

- [40] H. Zhang, Y. Wang, F. Dayoub, N. Sunderhauf, Varifocalnet: An iou-aware dense object detector, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8514–8523.
- [41] H. Zhang, S. Zhang, Shape-IoU: More accurate metric considering bound box shape and scale, 2024, [arXiv:2312.17663](https://arxiv.org/abs/2312.17663).
- [42] H. Zhao, O. Gallo, I. Frosio, J. Kautz, Loss functions for image restoration with neural networks, *IEEE Trans. Comput. Imaging* 3 (1) (2017) 47–57.
- [43] Baidu AI Studio, Insulator defect dataset, 2024, Available online: <https://aistudio.baidu.com/datasetdetail/269952>, (Accessed: 21 May 2026).
- [44] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, H. Ling, Detection and tracking meet drones challenge, *IEEE Trans. Pattern Anal. Mach. Intell.* 44 (11) (2021) 7380–7399.
- [45] M. Tan, Q. Le, EfficientNet: Rethinking model scaling for convolutional neural networks, in: Proceedings of the International Conference on Machine Learning, Long Beach, USA, 2019, pp. 6105–6114.
- [46] J. Chen, S. Kao, H. He, W. Zhuo, S. Wen, C.H. Lee, S.H.G. Chan, Run, don't walk: Chasing higher FLOPS for faster neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, Canada, 2023, pp. 12021–12031.
- [47] D. Qin, C. Lechner, M. Delakis, M. Fornoni, S. Luo, F. Yang, W. Wang, C. Banbury, C. Ye, B. Akin, V. Aggarwal, T. Zhu, D. Moro, A. Howard, MobileNetV4-universal models for the mobile ecosystem, 2024, [arXiv:2404.10518](https://arxiv.org/abs/2404.10518).
- [48] Z. Zheng, P. Wang, D. Ren, W. Liu, R. Ye, Q. Hu, W. Zuo, Enhancing geometric factors in model learning and inference for object detection and instance segmentation, *IEEE Trans. Cybern.* 52 (8) (2022) 8574–8586.
- [49] Y.F. Zhang, W. Ren, Z. Zhang, Z. Jia, L. Wang, T. Tan, Focal and efficient IOU loss for accurate bounding box regression, *Neurocomputing* 506 (2022) 146–157.
- [50] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, S. Savarese, 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, California, USA, 2019, pp. 658–666.
- [51] Z. Gevorgyan, SloU loss: More powerful learning for bounding box regression, 2022, [arXiv:2205.12740](https://arxiv.org/abs/2205.12740).
- [52] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, D. Ren, Distance-iou loss: Faster and better learning for bounding box regression, in: Proceedings of the AAAI Conference on Artificial Intelligence, 34, (7) New York, USA, 2020, pp. 12993–13000.
- [53] J. Mao, L. Zhang, Basic theories and methods of target's height and distance measurement based on monocular vision, *Int. J. Netw. Dyn. Intell.* 4 (1) (2025) 100007.
- [54] G. Akkad, A. Mansour, E. Inaty, Embedded deep learning accelerators: A survey on recent advances, *IEEE Trans. Artif. Intell.* 5 (5) (2023) 1954–1972.