A Lightweight Normalization-Free Architecture for Object Detection in High-Spatial-Resolution Remote Sensing Imagery

Youyou Li, Yuxiang Fang, Shixiong Zhou, Teng Long, Yicheng Zhang, *Member, IEEE*, Nuno Antunes Ribeiro, and Farid Melgani, *Fellow, IEEE*

***Abstract*—High-spatial-resolution remote sensing imagery applications demand computationally efficient detection methods due to their massive scale and real-time processing requirements. However, fast and reliable object detection in high-spatial-resolution remote sensing imagery is very challenging. Global channel-attention modules are computationally intensive, and batch-normalization layers incur extra computation, increasing latency for megapixel-scale inputs. To address these limitations, we propose Lino-YOLO, a lightweight, normalization-free architecture specifically designed for resource-constrained remote sensing environments. With only 2.64 million parameters, Lino-YOLO integrates three efficiency-oriented components. These include coordinate attention, which enables precise spatial context extraction at minimal computational cost; dynamic tanh, which eliminates the need for batch normalization and provides lightweight training stability; and wise-IoU, which optimizes gradient allocation without additional computational overhead. Our lightweight design achieves remarkable efficiency, demonstrated by a 7.6 ms inference time on an RTX 4090 GPU, while delivering superior accuracy—93.7% mAP@0.5 on the NWPU-VHR10 benchmark. Comprehensive evaluation across multiple YOLO families (v8–v11) demonstrates consistent improvements, with gains ranging from 6.6 to 11.0 percentage points over their respective baselines. Further evaluation on the challenging DIOR dataset reveals consistent improvements across all categories, confirming the effectiveness and robustness of our approach. This exceptional performance-to-size ratio makes Lino-YOLO ideal for large-scale, real-time remote sensing tasks, where both speed and accuracy are critical.**

***Index Terms*—High-spatial-resolution (HSR) remote sensing imagery, Lino-YOLO, multicategory object detection.**

I. Introduction

HIGH-SPATIAL-RESOLUTION (HSR) remote sensing imagery is now produced at an unprecedented rate through satellite constellations, such as WorldView [1], Gaofen [2], and Jilin [3], as well as widespread uncrewed aerial vehicles [4]. These massive datasets support critical applications including urban monitoring, disaster assessment, and traffic management [5], [6], [7]. Nevertheless, the sheer volume of data creates unique challenges: to effectively support time-sensitive applications, such as emergency response and surveillance, processing frameworks must efficiently handle terabytes of high-resolution imagery while simultaneously ensuring real-time or near real-time performance [8], [9], [10].

To meet these demands, recent research has focused on developing object detection frameworks that are both lightweight and accurate—ensuring deployment feasibility on edge devices without compromising detection reliability. One prominent approach involves adapting established architectures, such as faster R-CNN through compression techniques, yielding lightweight variants tailored for optical remote sensing data [11]. Likewise, various YOLO-based models, such as YOLOv5 and its derivatives, have been extended to tackle the challenges of remote sensing scenarios characterized by small object sizes, high object density, and complex backgrounds [12], [13]. For example, YOLO-DRS integrates bioinspired spatial modules to improve multiscale feature extraction, enhancing accuracy across diverse imaging conditions.

In parallel, methods such as Light-YOLOv4 employ pruning and quantization to significantly reduce computational load while maintaining robust detection accuracy [14]. Other efforts, including HRLE-SARDet and UniConDet, introduce innovative representations and unified learning paradigms to improve detection performance on SAR and multigrained tasks, respectively [15], [16]. In addition, recent work such as CGK-FSOD leverages pretrained generative priors to enhance generalization in few-shot settings [17]. Despite these substantial achievements, current methods still exhibit several limitations. For

Received 28 May 2025; revised 1 August 2025; accepted 4 September 2025. Date of publication 12 September 2025; date of current version 29 September 2025. This work was supported in part by the Sichuan Science and Technology Program under Grant 2023NSFSC0753 and in part by the Fundamental Research Funds for the Central Universities under Grant 25CAFUC04065. *(Corresponding author: Teng Long.)*

Youyou Li, Yuxiang Fang, and Shixiong Zhou are with the School of Air Traffic Management, Civil Aviation Flight University of China, Deyang 618307, China (e-mail: liyouyou@cafuc.edu.cn; fangyuxiang@cafuc.edu.cn; zsx@cafuc.edu.cn).

Teng Long is with the Informatics Institute, University of Amsterdam, Amsterdam 1098, The Netherlands (e-mail: t.long@uva.nl).

Yicheng Zhang is with the Institute for Infocomm Research at the Agency for Science, Technology and Research, Singapore 138632 (e-mail: zhang_yicheng@i2r.a-star.edu.sg).

Nuno Antunes Ribeiro is with the Aviation Studies Institute, Singapore University of Technology and Design, Singapore 487372 (e-mail: nuno_ribeiro@sutd.edu.sg).

Farid Melgani is with the Department of Information Engineering and Computer Science, University of Trento, 38123 Trento, Italy (e-mail: melgani@disi.unitn.it).

The source code presented in this study is available on GitHub at https://github.com/a211400/Lightweight-Normalization-free-Architecture-for-Object-Detection.

Digital Object Identifier 10.1109/JSTARS.2025.3609658

instance, they often inadequately incorporate positional context, hindering their ability to discriminate densely clustered or irregularly shaped objects. In addition, computational efficiency remains a concern, as models frequently depend heavily on normalization layers and IoU-based variants, such as GIoU and CIoU, both of which can negatively impact memory efficiency and training effectiveness in complex aerial imagery scenarios [18], [19], [20]. Therefore, addressing these challenges by effectively leveraging positional attention mechanisms, reducing dependency on normalization layers, and employing dynamic IoU-based loss functions is highly desirable.

These remaining challenges—in computational efficiency, spatial precision, and training stability—highlight an unmet need for a unified, lightweight detector optimized for HSR remote sensing. This clearly motivates our proposed architecture, Lino-YOLO, which integrates coordinate attention (CA) [21] for spatial precision, dynamic tanh (DyT) [22] for inference efficiency, and wise-IoU (WIoU) loss [23] for balanced gradient propagation.

Specifically, existing remote sensing object detection methods still encounter several key limitations. Traditional channel-attention mechanisms heavily depend on global pooling operations, inherently neglecting crucial positional information necessary for effectively distinguishing densely clustered or irregularly shaped objects in aerial images. This limitation can be effectively mitigated by CA, which efficiently encodes positional context without substantially increasing computational complexity [24], [25]. Moreover, widely adopted normalization layers, such as batch normalization, impose significant computational overhead and memory usage, hindering real-time performance on megapixel-scale remote sensing imagery. This issue is directly addressed by DyT, which innovatively removes normalization layers, significantly enhancing computational and memory efficiency [22]. In addition, conventional IoU-based loss functions in detectors, such as YOLO (e.g., IoU, GIoU, CIoU) primarily measure spatial overlap but allocate gradients uniformly across samples, regardless of localization quality. This often results in inefficient optimization and degraded performance, especially in dense scenes [26], [27]. To address this, the proposed method incorporates WIoU [23], which introduces an uncertainty-aware gradient reallocation mechanism. By dynamically downweighting low-quality samples and emphasizing reliable ones, WIoU improves convergence stability and detection accuracy under complex spatial distributions. While these advancements provide substantial improvements individually, integrating these approaches cohesively remains an open research direction, motivating our proposed method.

Together, these enhancements—CA for improved spatial accuracy, DyT for greater inference efficiency, and WIoU for balanced gradient propagation—form the core innovations of Lino-YOLO, a lightweight, normalization-free YOLO architecture specifically optimized for object detection in high-resolution remote-sensing imagery. Evaluated using the NWPU-VHR10 [28], [29], [30] and DIOR datasets [31], Lino-YOLO demonstrates superior accuracy, computational efficiency, and robustness compared to existing methods. Our main contributions are threefold as follows.

1) *Unified Lino-YOLO architecture:* We propose Lino-YOLO by integrating CA, DyT, and WIoU into a unified plug-in module. This module efficiently captures fine-grained positional information, removes costly normalization layers, and ensures balanced regression gradients, resulting in a compact yet highly effective detector particularly suited for small and densely packed targets.
2) *Backbone-agnostic modularity:* The proposed module is easily adaptable across various YOLO architectures (v8 [32], v9 [33], v10 [34], and v11 [35]), consistently enhancing performance in both mAP and computational efficiency without requiring additional hyperparameter tuning.
3) *State-of-the-art performance and efficiency:* Comprehensive experimental validation on the NWPU-VHR10 and DIOR datasets demonstrates that Lino-YOLO consistently surpasses established baseline methods, achieving superior accuracy, faster inference times, and robustness across complex, multiscale remote-sensing scenarios.

The rest of this article is organized as follows. Section II elaborates on the Lino-YOLO methodology, including CA, DyT, and WIoU. Section III provides detailed descriptions of the datasets used. Section IV outlines implementation specifics. Section V presents comprehensive experimental results. Section VI concludes this article. Finally, Section VII discusses key insights and future research directions.

II. Methods

In this section we introduce Lino-YOLO, short for a lightweight normalization-free YOLO architecture for object detection in HSR remote sensing imagery that can be integrated into recent member of the YOLO family. Through theoretical analysis, we discovered that the combination of CA and DyT consistently improves detection performance across recent YOLO variants, leading us to propose this CA+DyT scheme as a generalizable enhancement for remote sensing object detection. While YOLOv11 is chosen as the primary baseline due to its recent introduction and strong performance among current YOLO variants, our extensive experiments (see Section IV) demonstrate that the proposed scheme significantly improves detection accuracy across YOLOv8, YOLOv9, YOLOv10, and YOLOv11. The CA module injects a lightweight attention mechanism that performs two 1-D global encodings (along height and width, respectively) and reweights the feature map through elementwise multiplication, thus capturing long-range dependencies while preserving precise spatial cues critical for remote sensing imagery. Complementing this, the statistic-free DyT operator replaces conventional batch normalization, eliminating computational overhead while improving training stability and detection accuracy. Finally, the WIoU loss reallocates gradient gain toward medium-quality anchors, reducing regression error and further boosting overall performance. A schematic overview of the complete architecture is provided in Fig. 1.

CA is explicitly integrated following the SPPF block within the YOLO backbone. By doing so we exploit two key merits of CA—its lightweight design and its ability to capture long-range dependencies with negligible computational overhead—to boost

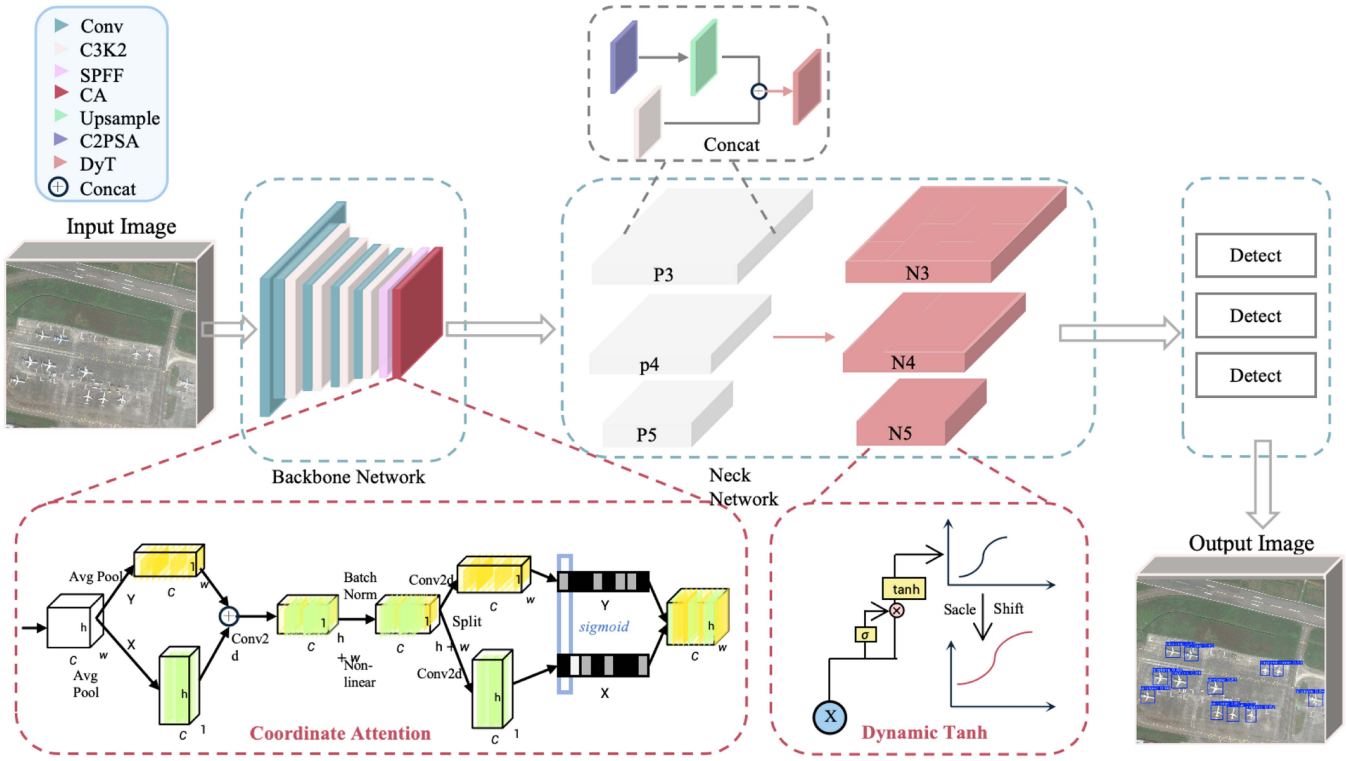


Fig. 1. Architecture of YOLOv11 augmented with CA and DyT modules, illustrating how CA is embedded into the backbone and DyT is integrated into the detection head.

both efficiency and detection accuracy. Concretely, CA performs two 1-D global-pooling operations along the height and width directions, concatenates the resulting descriptors and uses them to reweight the original feature map, thus reinforcing salient object cues while preserving precise spatial information.

Given an input feature map $\mathrm{X} \in \mathbb{R}^{\mathrm{C}\times\mathrm{H}\times\mathrm{W}}$, where C is the number of channels, and H and W are the height and width of the feature map, respectively. First, global average pooling is performed on each channel along the horizontal (channel) direction to obtain a direction-aware feature map $\mathrm{Z_h} \in \mathbb{R}^{\mathrm{C}\times\mathrm{H}}$

$$\mathrm{z_h^c}\left(\mathrm{h}\right) = \frac{1}{\mathrm{W}} \sum_{\mathrm{i}=0}^{\mathrm{W}-1} \mathrm{x_c}\left(\mathrm{h}, \mathrm{i}\right). \tag{1}$$

Among them, $\mathrm{z_h^c}(\mathrm{h})$ is the pooling result of the cth channel at height h.

Next, global average pooling is performed on each channel along the vertical direction (height direction) to obtain another direction-aware feature map $\mathrm{Z_w} \in \mathbb{R}^{\mathrm{C}\times\mathrm{H}}$

$$\mathrm{z_w^c}\left(\mathrm{w}\right) = \frac{1}{\mathrm{H}} \sum_{\mathrm{j}=0}^{\mathrm{H}-1} \mathrm{x_c}\left(\mathrm{j}, \mathrm{w}\right). \tag{2}$$

Among them, $\mathrm{z_w^c}(\mathrm{w})$ is the pooling result of the cth channel at width w.

Further, Z_h and Z_w are concatenated along the spatial dimension, and then feature fusion is performed through a shared 1×1 convolution transformation function F_1

$$f = \mathrm{ReLU}\left(\mathrm{F_1}\left(\left[\mathrm{Z_h}, \mathrm{Z_w}\right]\right)\right) \tag{3}$$

$$\mathrm{ReLU}\left(\mathrm{x}\right) = \begin{cases} \mathrm{x}, & \text{if } \mathrm{x} > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Among them, $[Z_\mathrm{h}, Z_\mathrm{w}]$ represents the splicing operation along the spatial dimension, ReLU is the nonlinear activation function

$$f = [f_h, f_w], \quad f \in \mathbb{R}^{\frac{C}{r}\times(H+W)}$$

where f is the intermediate fused feature obtained from concatenating the direction-aware tensors f_h and f_w along the spatial dimension. r is the reduction ratio, which is used to control the size of the block.

Divide f into two independent tensors $\mathrm{f_h}$ and $\mathrm{f_w}$ along the spatial dimension

$$\mathrm{f_h} \in \mathbb{R}^{\mathrm{C/r}\times\mathrm{H}} \tag{5}$$

$$\mathrm{f_w} \in \mathbb{R}^{\mathrm{C/r}\times\mathrm{W}}. \tag{6}$$

Apply the 1×1 convolution transformations F_h and F_w to f_h and f_w respectively to obtain two attention maps g_h and g_w

$$\mathrm{g_h} = \sigma\left(\mathrm{F_h}\left(\mathrm{f_h}\right)\right) \tag{7}$$

$$\mathrm{g_w} = \sigma\left(\mathrm{F_w}\left(\mathrm{f_w}\right)\right). \tag{8}$$

Among them, σ is the sigmoid activation function, which is used to limit the output to the range of (0, 1), $\mathrm{g_h} \in \mathbb{R}^{\mathrm{C}\times\mathrm{H}}$ and $\mathrm{g_w} \in \mathbb{R}^{\mathrm{C}\times\mathrm{W}}$. Finally, enhance the feature representation

through the elementwise multiplication operation, and apply the generated attention maps g_h and g_w to the input feature map X

$$y_c(i,j) = x_c(i,j) \times g_h^c(i) \times g_w^c(j). \tag{9}$$

Among them, $y_c(i,j)$ is the value at the c channel position (i, j) in the output feature map $Y \in \mathbb{R}^{C\times H\times W}$.

In summary, the CA mechanism captures the global information of the feature map in the horizontal and vertical directions through two 1-D global pooling operations, while retaining the position information in each direction. Then, through two independent convolution transformations, it generates attention maps along the horizontal and vertical directions, respectively. Each attention map captures the long-range dependencies of the input feature map in the corresponding direction. Finally, apply the attention maps to the input feature map to highlight the features of the target object and suppress the features of the background and other irrelevant regions, thereby improving the detection and recognition ability of the model.

A. Dynamic Tanh

Official YOLO variants insert a batch normalization layer after most convolutional layers to compute minibatch mean and variance, normalizing each layer's inputs to stabilize training and mitigate internal covariate shift. The main operation steps are as follows.

For the minibatch of input data, calculate the mean and variance of each feature channel. Let the input data be X_i, where i represents the sample index, j represents the feature channel index, and the minibatch size is m. Then, the mean μ_j and variance σ_j^2 are calculated as follows:

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} X_{ij} \tag{10}$$

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m} (x_{ij} - \mu_j)^2. \tag{11}$$

Subsequently, normalize the data within each feature channel using the computed mean and variance to achieve zero mean and unit variance

$$\widehat{x}_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}}. \tag{12}$$

Among them, ϵ is a very small constant used to prevent division by zero errors.

To maintain the representational ability of the network, the BN layer applies an affine transformation using learnable scaling parameter γ_j and offset parameter β_j after normalization

$$y_{ij} = \gamma_j \cdot \widehat{x}_{ij} + \beta_j. \tag{13}$$

Through the above-mentioned analysis, it can be found that BN relies heavily on minibatch statistics, making its normalization effect sensitive to batch size and introducing discrepancies between training and inference phases, potentially degrading model performance. When the minibatch is small, the statistical information is estimated inaccurately, resulting in poor normalization effects. In addition, BN uses different statistical information during the training and inference phases, which uses minibatch statistics during training and global statistics during inference. This inconsistency leads to a decline in the model's inference performance.

To address these issues, this paper introduces DyT, replacing BN in YOLO's neck architecture. By learning an appropriate scaling factor α and utilizing the boundedness of the tanh function to mimic the behavior of BN, it stabilizes the training process without calculating the activation statistics and significantly improves the model's computational efficiency and detection accuracy. Its calculation is as follows.

For an input tensor $x \in \mathbb{R}^{B\times T\times C}$, scale the input tensor using a learnable scalar parameter α and perform a nonlinear transformation through the tanh function

$$X = \tanh(\alpha x). \tag{14}$$

Among them, B is the batch size, T is the number of tokens, and C is the embedding dimension of each token.

Then, apply the learnable parameters γ and β for an affine transformation

$$\mathrm{DyT}(x) = \gamma \cdot \tanh(\alpha x) + \beta. \tag{15}$$

The theoretical basis for DyT replacing BN lies in its ability to address internal covariate shift—a phenomenon where the distribution of layer inputs changes during training—without relying on batch statistics. BN mitigates this by explicitly normalizing activations to zero mean and unit variance, but DyT achieves similar stabilization through the intrinsic properties of the tanh function combined with dynamic scaling.

The tanh function, defined as $\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$, is bounded between –1 and 1, which naturally constrains the output range of activations and prevents explosion or vanishing gradients that can destabilize training. The learnable scaling parameter α dynamically adjusts the input to tanh, controlling the "steepness" of the nonlinearity. For small α, the function approximates a linear transformation near zero ($\tanh(\alpha x) \approx \alpha x$), while larger α pushes activations toward saturation, effectively bounding extreme values.

This dynamic scaling ensures training stability by adaptively normalizing the activation distribution. Consider the derivative of DyT, which governs gradient flow during backpropagation

$$\frac{\partial}{\partial x}\mathrm{DyT}(x) = \gamma \cdot \alpha \cdot \mathrm{sech}^2(\alpha x) \tag{16}$$

where $\mathrm{sech}^2(\alpha x)$, where $\mathrm{sech}(z) = 1/\cosh(z)$ is the hyperbolic secant function, is always positive and bounded between 0 and 1, ensuring smooth and controlled gradient propagation. Unlike BN, where gradients depend on batch variance (potentially noisy for small batches), DyT's gradients are deterministic and independent of batch size, leading to consistent behavior across training and inference.

Furthermore, DyT implicitly centers activations around zero due to tanh's odd symmetry ($\tanh(-z) = -\tanh(z)$), mimicking BN's mean subtraction without explicit computation. For inputs following a Gaussian distribution $x \sim \mathcal{N}(\mu, \sigma^2)$, the effective normalization can be approximated by choosing $\alpha \propto 1/\sigma$, scaling the input variance to match tanh's linear regime

(roughly $|z| < 1$). This adaptive bounding reduces covariate shift by preventing activations from growing unbounded, as evidenced by empirical studies showing lower variance in layer outputs compared to unnormalized networks [22]. In essence, DyT provides a statistic-free alternative that maintains BN's stabilization benefits—bounded activations, centered distributions, and controlled gradients—while eliminating computational overhead and batch dependencies, making it suitable for high-resolution remote sensing tasks where efficiency is critical.

In summary, DyT achieves effective feature transformation through dynamic scaling, a nonlinear tanh transformation, and an affine operation. Compared to traditional normalization methods that rely on batch statistics, this scaling-based approach offers several advantages: it simplifies computation, reduces dependency on batch dimensions, and maintains computational efficiency, thereby enhancing training stability and detection accuracy. However, a potential drawback of replacing normalization with pure scaling is the loss of statistical normalization properties, potentially impacting the model's robustness under varying input distributions. Nonetheless, the simplicity and effectiveness of DyT's scaling operation generally lead to strong performance with high generalizability [22].

B. Wise-IoU

The WIoU loss function uses a dynamic nonmonotonic focusing mechanism to evaluate the quality of anchor boxes based on the outlier degree. It effectively allocates gradient gains, reduces the competitiveness of high-quality anchor boxes, and weakens the harmful gradients generated by low-quality samples [23]. This allows the model to focus on learning ordinary-quality anchor boxes, thereby improving the overall performance and generalization ability of the detector. It significantly reduces the regression error and effectively improves the model accuracy. The definition of WIoU is as follows:

$$\mathrm{L_{WIoU}} = \mathrm{L_{IoU}} + \mathrm{R_{WIoU}}. \tag{17}$$

Among them, $\mathrm{L_{IoU}}$ is the loss function based on IoU, and the calculation formula is

$$\mathrm{L_{IoU}} = 1 - \mathrm{IoU} = 1 - \frac{\mathrm{W_i H_i}}{\mathrm{S_u}}. \tag{18}$$

Among them, $\mathrm{W_i}$ and $\mathrm{H_i}$ are the width and height of the overlapping part, and $\mathrm{S_u}$ is the sum of the areas of the predicted box and the ground-truth box minus the overlapping area.

$\mathrm{R_{WIoU}}$ is the penalty term for WIoU, which is used to solve the gradient vanishing problem and balance the influence of samples of different qualities. Its definition is as follows [23]:

$$\mathrm{R_{WIoU}} = \beta \cdot \left(\frac{\mathrm{x} - \mathrm{x_{gt}}}{\mathrm{W_g}}\right)^2 + \beta \cdot \left(\frac{\mathrm{y} - \mathrm{y_{gt}}}{\mathrm{H_g}}\right)^2. \tag{19}$$

Among them, β is the dynamically adjusted focusing coefficient, which is used to measure the outlier degree of the anchor box. The calculation formula is

$$\beta = \frac{\mathrm{L^*_{IoU}}}{\mathrm{L_{IoU}}}. \tag{20}$$

Among them, $\mathrm{L^*_{IoU}}$ is the exponential moving average of $\mathrm{L_{IoU}}$, which is used to dynamically update the focusing coefficient.

The dynamic nonmonotonic focusing mechanism controls the influence of anchor boxes of different qualities by adjusting the gradient gain.

- For high-quality anchor boxes (with a small $\mathrm{L_{IoU}}$), a small gradient gain is assigned to reduce their excessive influence on training.
- For low-quality anchor boxes (with a large $\mathrm{L_{IoU}}$), a small gradient gain is also assigned to reduce the interference of the harmful gradients they generate on model training.

Specifically, the gradient gain r is defined as follows:

$$\mathrm{r} = \left(\frac{\mathrm{L^*_{IoU}}}{\mathrm{L_{IoU}}}\right)^{\gamma}. \tag{21}$$

Among them, γ is the hyperparameter that controls the change of the gradient gain.

In summary, through the dynamic nonmonotonic focusing mechanism, WIoU reasonably assigns gradient gains, effectively improves the model's attention to anchor boxes of ordinary quality, reduces the influence of low-quality samples on training, and thus improves the positioning accuracy and generalization ability of the object detection model.

C. Synergistic Mechanism of CA, DyT, and WIoU

The combined introduction of CA, DyT, and WIoU produces an architectural variant that raises detection accuracy while keeping computational cost essentially unchanged. Denote by F_b the baseline FLOPs of the chosen YOLO backbone. The extra workload contributed by CA, denoted as F_{CA} can be written as

$$F_{\mathrm{CA}} = \frac{2C\,(H+W)}{r} \tag{22}$$

with C, H, and W the channel, height, and width dimensions, and r the internal reduction ratio. Because the dominant convolutional complexity scales with $C\,H\,W$, the ratio $F_{\mathrm{CA}}/F_b = O\big((H+W)/(rHW)\big)$ is negligible for common resolutions. DyT is an elementwise mapping and therefore adds only

$$F_{\mathrm{DyT}} = O(C) \tag{23}$$

which replaces the batch dependent complexity of batch normalization with a constant-time alternative. WIoU modifies only the training loss and leaves inference unchanged. Consequently

$$F_{\mathrm{total}} = F_b + F_{\mathrm{CA}} + F_{\mathrm{DyT}} = F_b\big(1 + o(1)\big) \tag{24}$$

so the runtime cost remains effectively constant and realtime throughput is preserved. Accuracy gains derive from a sequence of complementary regularizations. CA reweights the input tensor X with spatial maps g_h and g_w, producing $Y = X \odot g_h \odot g_w$. The variance of Y therefore obeys

$$\mathrm{Var}(Y) \le \mathrm{Var}(X)\,\big\|g_h \odot g_w\big\|_\infty^2 \tag{25}$$

indicating that representational energy is concentrated on informative regions. DyT then applies the Lipschitz-bounded transform $z \mapsto \gamma \tanh(\alpha z) + \beta$ with $|\tanh(\alpha z)| \leq 1$; hence

$$\mathrm{Var}\big(\mathrm{DyT}(Y)\big) \leq (\gamma\alpha)^2 \Big[1 - \tanh^2\big(\alpha \mathbb{E}[Y]\big)\Big] \tag{26}$$

which prevents activation explosion and yields batchsize-independent normalization. Finally, WIoU rescales the regression gradient by a factor $r \in [0, 1]$, $\nabla L_{\mathrm{WIoU}} = r\, \nabla L_{\mathrm{IoU}}$, thereby tempering low-quality anchors and preventing the domination of very accurate ones. The composite Jacobian becomes

$$\frac{\partial L}{\partial X} = r\, \gamma\big(1 - \tanh^2(\alpha Y)\big)\, g_h\, g_w\, \nabla L_{\mathrm{IoU}} \tag{27}$$

so CA channels gradients to salient spatial positions, DyT ensures nonsaturating flow, and WIoU reallocates learning capacity to the most informative anchors. This joint conditioning explains the empirically observed superadditive gain in mean average precision (mAP), achieved without perceptible inference-time penalty.

III. Materials

In this study, we employ two representative datasets, namely NWPU-VHR10 [28], [29], [30] and DIOR [31], to rigorously evaluate the performance of our proposed method. Both datasets are widely recognized benchmarks in the field of remote sensing object detection, featuring diverse object categories, varied scales, and complex backgrounds. Their complementary characteristics provide comprehensive scenarios for assessing algorithm robustness, generalization capability, and detection accuracy under realistic and challenging conditions, making them ideal choices for validating the effectiveness of our approach.

A. NWPU-VHR10 Dataset

The NWPU-VHR10 benchmark is one of the most widely used test beds for evaluating multiclass object detection in very-high-resolution remote-sensing imagery [28], [29], [30]. It is composed of 800 RGB scenes collected from Google Earth and Digital Globe with ground-sampling distances ranging from 0.08 to 2 m and side lengths between roughly 300 and 1500 pixels, thereby covering a broad spectrum of spatial scales. Among them, 650 images contain at least one target while 150 are deliberately left free of objects to facilitate the measurement of false-alarm rates. Ten categories—airplane, ship, storage tank, baseball diamond, tennis court, basketball court, ground track field, harbor, bridge, and vehicle—are exhaustively annotated with axis-aligned bounding boxes, yielding 3651 labeled instances. The official protocol follows five-fold cross-validation and reports mAP at an IoU threshold of 0.5, though fixed 70/30 splits are also common in subsequent studies. Pronounced intraclass appearance variation, densely packed small objects such as parking-lot vehicles, and background clutter around ports and sports facilities make NWPU-VHR10 a challenging yet indispensable benchmark for assessing the robustness and generalization capability of modern detection architectures in high-resolution aerial scenarios. Fig. 2 illustrates the object categories in the NWPU-VHR10 dataset, with example images directly cropped from the dataset.

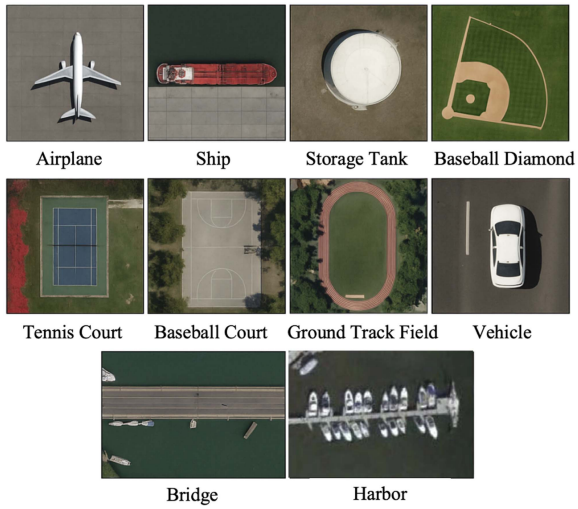


Fig. 2. Illustration showing examples of the ten object categories in the NWPU-VHR10 dataset.

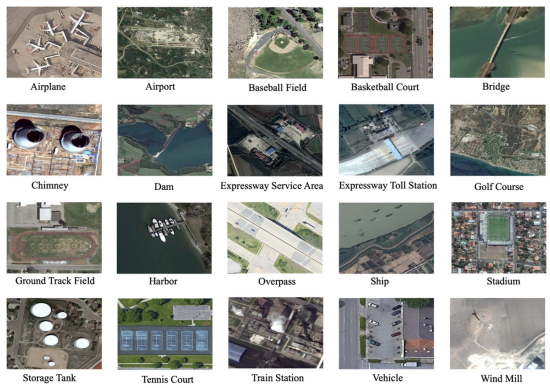


Fig. 3. Illustration showing examples of the 20 object categories in the DIOR dataset.

B. DIOR Dataset

Images in the DIOR dataset are primarily sourced from open remote sensing platforms, such as Google Earth, covering diverse geographic regions and environmental conditions, including urban, rural, mountainous, and coastal areas [36], [37]. These images were collected under varying imaging conditions, weather scenarios, seasons, and image qualities, ensuring data diversity and representativeness. The spatial resolution of the images ranges from 0.5 to 30 m. The dataset comprises a total of 23 463 images, with 192 472 object instances manually annotated using axis-aligned bounding boxes. This study employs a uniformly sampled subset, consisting of one-fifth of the training and validation images from each category. To clearly demonstrate the model's capabilities without relying heavily on extensive labeled datasets and to enhance data efficiency, this study employs a randomly and uniformly sampled subset consisting of one-fifth of the total images from each category, amounting to 2930 images in total [38], [39]. This subset is further divided into

a training set of 2344 images with corresponding bounding box annotations and a validation set of 586 images, also accompanied by bounding box annotations. The dataset includes the following categories: airplane, airport, baseball field, basketball court, bridge, chimney, dam, expressway service area, expressway toll station, golf course, ground track field, harbor, overpass, ship, stadium, storage tank, tennis court, train station, vehicle, and windmill. Fig. 3 illustrates the object categories in the DIOR dataset.

IV. Implementation Details

This section provides a comprehensive description of the hardware and software environment, dataset preprocessing pipeline, training hyperparameters, and evaluation metrics adopted in our experiments.

All experiments were executed on a Linux workstation equipped with six `Nvidia RTX 4090` GPUs. The software stack consisted of `Python 3.11.8`, `PyTorch 2.2.2` compiled with `CUDA 12.6` and `cuDNN 8.9.0.2`.

Training and inference were implemented on top of the official `ultralytics/yolo` code base, which we extended with customized CA, DyT, and WIoU modules.

Following the standard practice for NWPU-VHR10, we adopted the official training-validation split (70/30) and further partitioned the training portion into batches of 36 images. Each image was resized to 640×640 pixels while preserving its aspect ratio, padding the remaining area with zeros when necessary. Models were trained for 300 epochs using Adam with weight decay 0.0005. The initial learning rate was set to 0.01.

Performance was quantified with mAP at IoU thresholds of 0.5 (`mAP@0.5`). We additionally report the number of parameters (`#Params`) and the throughput in frames per second (FPS) measured on the aforementioned 4090 GPU using a batch size of 1 and input resolution of 640×640. For statistical robustness, each reported figure is the arithmetic mean over three independent runs.

V. Results

This section offers a thorough evaluation of Lino-YOLO through five complementary lenses. First, we chart local variance curves and precision–recall curves to benchmark overall convergence and classwise performance. Second, attention heatmaps are visualized across categories to reveal how the CA and DyT modules sharpen spatial focus. Third, an ablation study on the lightweight YOLOv11n backbone isolates the individual and joint contributions of these three components. Fourth, we provide both qualitative illustrations and quantitative metrics against strong state-of-the-art detectors. Finally, we test generalizability by grafting Lino-YOLO onto four recent YOLO families (v8–v11), showing consistent gains in accuracy and speed.

A. Performance Comparison Via Local Variance and PR Curves

This section assesses the model's performance by analyzing the local variance curve and the PR curve. The local variance curve is used to evaluate the model's stability, capturing how consistently the model responds to local variations in the input space. The PR curve, on the other hand, provides insights into the model's classification performance, particularly under imbalanced data distributions.

Fig. 4 visualizes the local variance of training and validation losses across epochs for three model variants: YOLOv11n, YOLOv11n+CA, and YOLOv11n+CA+DyT. Specifically, it compares three distinct loss metrics—box_loss, cls_loss, and dfl_loss—in both training and validation phases. Generally, the local variance for all loss types initially rises sharply, reaches a peak within the first 10 to 20 epochs, and then rapidly declines, stabilizing near zero as training progresses. Notably, YOLOv11n+CA+DyT consistently demonstrates a higher peak variance in early epochs for both training and validation losses compared to the other two configurations, suggesting an initial phase of more significant optimization fluctuations before stabilizing.

Analyzing further, YOLOv11n+CA+DyT exhibits faster variance reduction after the initial peaks compared to the other models, especially evident in the validation set losses. This suggests that despite the higher early instability, the DyT variant effectively stabilizes optimization and converges quickly, possibly due to improved normalization capabilities. Conversely, YOLOv11n consistently shows the lowest peak variance across most loss types but experiences slightly slower variance reduction afterward. The addition of CA alone (YOLOv11n+CA) leads to intermediate variance behavior, indicating enhanced feature extraction capability relative to the baseline, but without matching the rapid stabilization offered by integrating DyT. Overall, these results underscore the beneficial impact of incorporating DyT in conjunction with CA, facilitating more effective and stable training dynamics.

Fig. 5 presents the PR curves for our proposed model compared against five state-of-the-art object detection frameworks on the NWPU-VHR10 dataset. The PR curves provide critical insights into detection performance across different confidence thresholds. Our model (subplot f) demonstrates superior performance with a consistently higher precision across all recall values, maintaining precision above 0.9 even at high recall rates (0.8–0.9). This indicates that our model achieves both high confidence detections and comprehensive object coverage, significantly outperforming the baseline models. In contrast, other models such as YOLOv8n (subplot b) and YOLOv9s (subplot c) show more rapid precision degradation as recall increases, particularly in the 0.7–0.9 recall range.

The area under the PR curve, which directly correlates with average precision (AP), is visibly larger for our model compared to all competitors. This quantitative advantage is particularly pronounced for challenging object categories in remote sensing imagery, such as small vehicles and harbor structures, where our model maintains high precision even at recall thresholds where other models falter. RT-DETR-l [40] (subplot a), despite its sophisticated transformer architecture, shows competitive but still inferior performance compared to our approach, especially in the high recall region. YOLOv10n and YOLOv11n (subplots d and e) demonstrate incremental improvements over

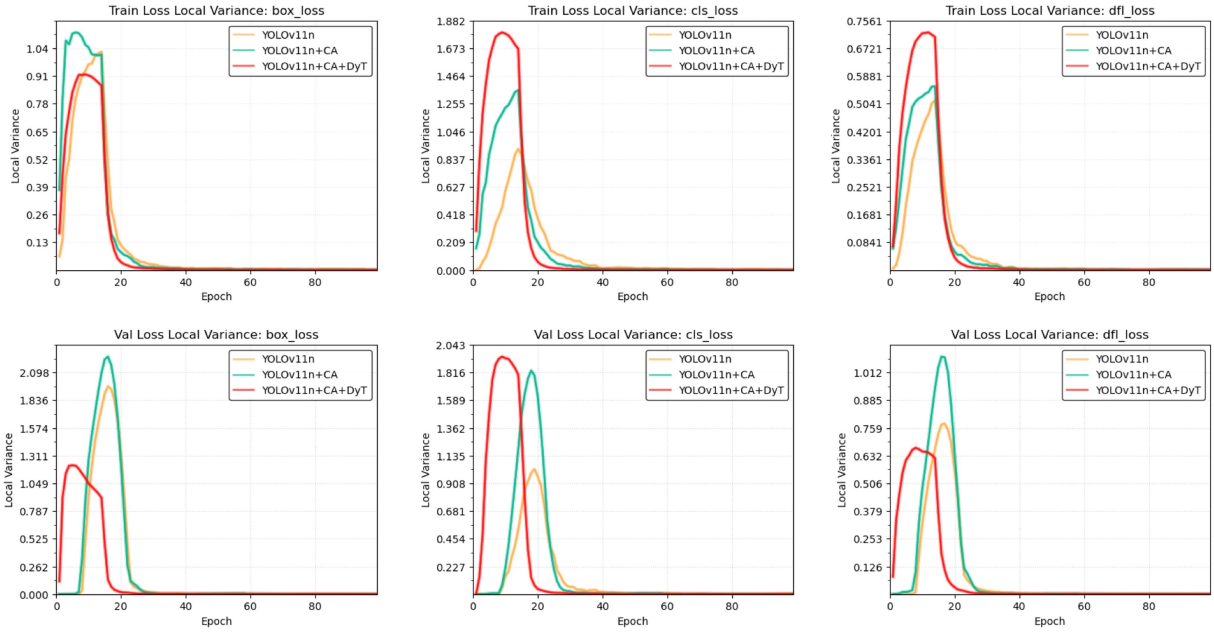


Fig. 4. Local variance comparison before and after model improvement.

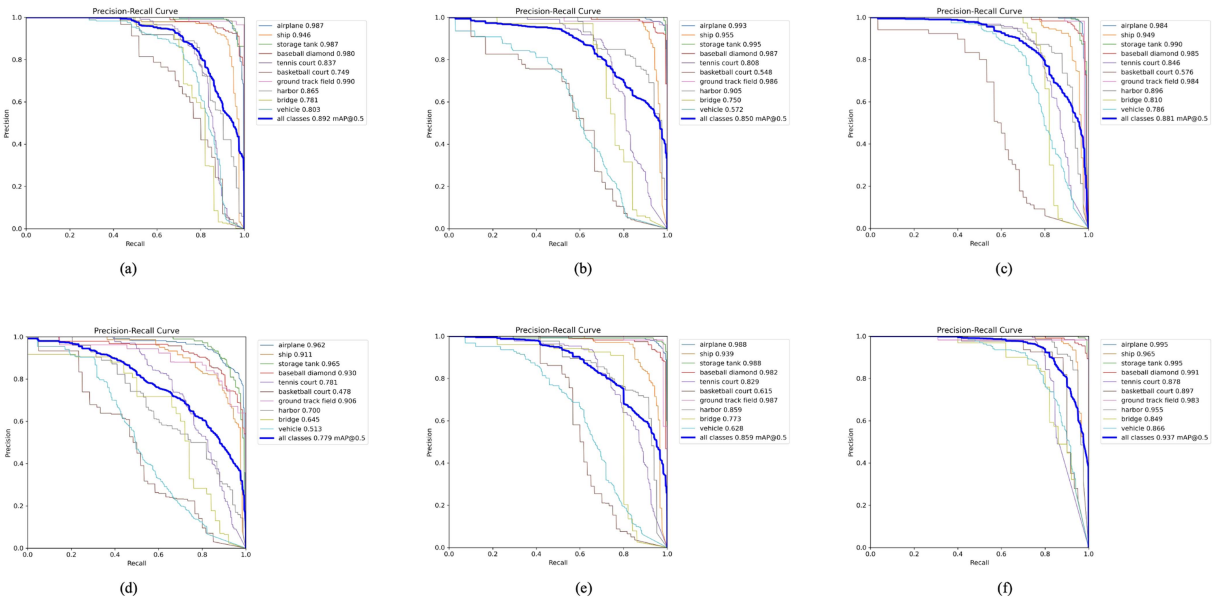


Fig. 5. PR-curve comparison. (a) RT-DETR-l. (b) YOLOv8n. (c) YOLOv9s. (d) YOLOv10n. (e) YOLOv11n. (f) Ours.

earlier YOLO versions but still fall short of the balanced precision–recall tradeoff achieved by our model, which benefits from the synergistic integration of CA, DyT, and WIoU mechanisms.

B. Attention Heatmap Comparison Across Categories

Fig. 6 presents a comparative visualization of attention heatmaps generated by our model across different object categories in the NWPU-VHR10 dataset. These heatmaps provide valuable insights into how our enhanced architecture with CA focuses on discriminative features compared to the baseline model. We extracted and merged the attention heatmaps of six strategic network layers (9, 10, 13, 17, 20, and 23), which represent key points in our architecture where CA modules were integrated.

This is particularly evident in the aircraft, ship, and storage tank categories. Here, the attention is tightly focused on the object boundaries and distinctive structural features.

For smaller objects such as vehicles and tennis courts, our model shows remarkable improvement in attention localization. The baseline model often exhibits diffuse attention patterns that extend beyond object boundaries, while our enhanced model produces sharper, more contained attention regions that closely

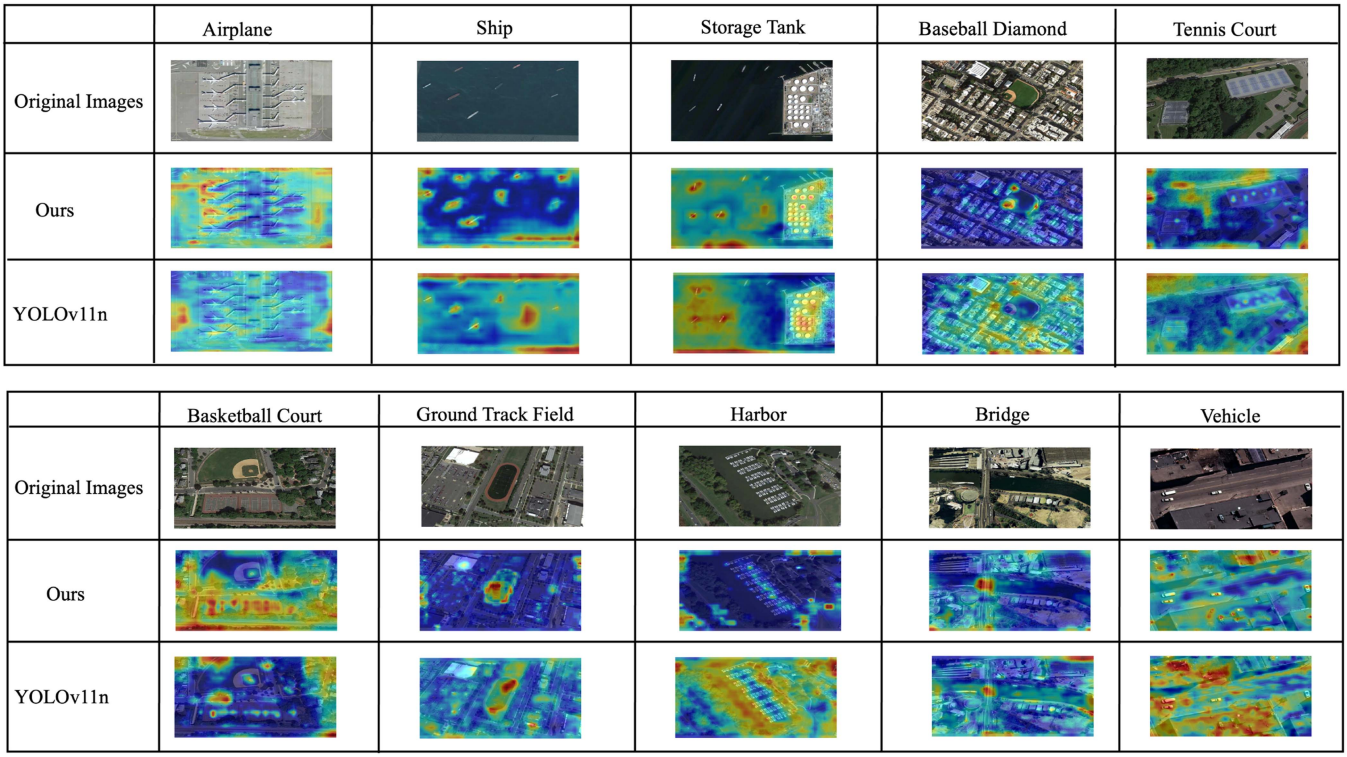


Fig. 6. PR-curve comparison. (a) RT-DETR-l. (b) YOLOv8n. (c) YOLOv9s. (d) YOLOv10n. (e) YOLOv11n. (f) Ours.

TABLE I
ABLATION EXPERIMENT (PART 1)

CA	DyT	WIoU	Airplane	Ship	Storage tank	Baseball diamond
–	–	–	98.5	93.6	98.8	96.6
✓	–	–	98.3 (–0.2)	94.9 (+1.3)	99.0 (+0.2)	98.7 (+2.1)
–	✓	–	99.2 (+0.7)	94.8 (+1.2)	97.8 (–1.0)	98.7 (+2.1)
–	–	✓	99.4 (+0.9)	96.1 (+2.5)	99.2 (+0.4)	98.5 (+1.9)
✓	✓	–	99.3 (+0.8)	96.4 (+2.8)	97.3 (–1.5)	97.9 (+1.3)
✓	–	✓	98.9 (+0.4)	91.6 (–2.0)	99.2 (+0.4)	99.1 (+2.5)
–	✓	✓	99.4 (+0.9)	94.5 (+0.9)	98.5 (–0.3)	98.7 (+2.1)
✓	✓	✓	99.5 (+1.0)	96.5 (+2.9)	99.5 (+0.7)	99.1 (+2.5)

align with the actual object dimensions. This improved focus directly contributes to the enhanced detection performance for small objects, which are particularly challenging in remote sensing imagery.

The harbor and bridge categories demonstrate our model's superior ability to handle complex structures with irregular shapes. The attention maps show that our model effectively captures the elongated structures of bridges and the intricate layouts of harbor facilities, with attention appropriately distributed along the entire structure rather than just concentrating on certain parts.

These visualizations provide direct evidence of how the CA mechanism enhances the model's ability to capture position-sensitive features by decomposing the spatial attention into two orthogonal directions. This directional attention proves particularly beneficial for objects with distinctive horizontal or vertical structures, such as bridges, ships, and buildings, where the attention maps show clear alignment with these directional features. The strategic placement of CA modules after layers 9 and 12 enables our model to refine spatial attention at multiple scales, contributing to the improved detection performance observed across various object categories.

C. Ablation Study in YOLOv11n

The ablation study presented in Tables I and II demonstrates the effectiveness of our proposed components: CA, DyT, and WIoU. The experiment systematically evaluates different combinations of these components across ten object categories from the NWPU-VHR10 dataset.

From Table I, we observe that each individual component contributes positively to the detection performance. When used alone, WIoU achieves the best performance on airplane (99.4%) and ship (96.1%), while DyT performs best on baseball diamond (98.7%). The combination of all three components yields the highest performance across all four categories, with notable improvements in ship detection (96.5%, +2.9%) and baseball

TABLE II
ABLATION EXPERIMENT (PART 2)

Tennis court	Basketball court	Ground track field	Harbor	Bridge	Vehicle	mAP@0.5
83.0	62.9	98.7	85.6	78.5	63.7	86.0
82.2 (−0.8)	54.8 (−8.1)	99.0 (+0.3)	90.0 (+4.4)	79.9 (+1.4)	62.3 (−1.4)	85.9 (−0.1)
81.6 (−1.4)	63.8 (+0.9)	98.9 (+0.2)	86.9 (+1.3)	84.0 (+5.5)	63.0 (−0.7)	86.9 (+0.9)
82.6 (−0.4)	55.2 (−7.7)	97.0 (−1.7)	84.4 (−1.2)	83.4 (+4.9)	62.1 (−1.6)	85.8 (−0.2)
82.7 (−0.3)	70.9 (+8.0)	98.4 (−0.3)	91.0 (+5.4)	80.5 (+2.0)	61.8 (−1.9)	87.6 (+1.6)
87.4 (+4.4)	64.5 (+1.6)	98.7 (0.0)	91.2 (+5.6)	75.5 (−3.0)	65.7 (+2.0)	87.2 (+1.2)
84.3 (+1.3)	65.6 (+2.7)	98.6 (−0.1)	88.5 (+2.9)	81.7 (+3.2)	66.6 (+2.9)	87.7 (+1.7)
87.8 (+4.8)	89.7 (+26.8)	98.3 (−0.4)	95.5 (+9.9)	84.9 (+6.4)	86.6 (+22.9)	93.7 (+7.7)

TABLE III
COMPARISON WITH ADVANCED METHODS ON THE NWPU-VHR10 DATASET (PART 1)

Methods	Inference (ms)	Params (M)	FLOPs (G)	Airplane	Ship	Storage tank	Baseball diamond
RTDETR-l [40]	11.3	32.00	103.5	99.1	96.7	97.8	96.9
YOLOv8n [41]	6.5	2.69	6.8	99.3	95.4	99.5	98.8
YOLOv9s [33]	8.6	7.18	26.7	98.1	95.4	98.2	98.7
YOLOv10n [34]	7.1	2.27	6.5	96.1	91.0	96.5	93.5
YOLOv11n [35]	7.3	2.59	6.3	98.5	93.6	98.8	96.6
Ours	7.6	2.54	6.4	99.5	96.5	99.5	99.1

TABLE IV
COMPARISON WITH ADVANCED METHODS ON THE NWPU-VHR10 DATASET (PART 2)

Methods	Tennis court	Basketball court	Ground track field	Harbor	Bridge	Vehicle	mAP@0.5
RTDETR-l [40]	82.0	71.1	98.1	89.2	68.6	73.8	87.3
YOLOv8n [41]	80.4	54.4	98.6	90.5	74.7	57.6	84.9
YOLOv9s [33]	84.2	62.3	98.6	93.3	89.8	75.6	89.4
YOLOv10n [34]	78.5	47.8	90.5	70.1	70.0	51.5	78.5
YOLOv11n [35]	83.0	62.9	98.7	85.6	78.5	63.7	86.0
Ours	87.8	89.7	98.3	95.5	84.9	86.6	93.7

diamond (99.1%, +2.5%). Table II further confirms this trend across the remaining six categories. The full model with all three components significantly outperforms partial implementations, particularly for challenging categories, such as basketball court (89.7% versus baseline 62.9%) and vehicle (86.6% versus baseline 63.7%). Overall, the complete model achieves a substantial improvement in mAP@0.5, reaching 93.7% compared to the baseline of 86.0%, representing a significant gain of 7.7%.

These results validate our design choices and demonstrate that CA, DyT, and WIoU are complementary, with their combination yielding synergistic improvements in detection performance across diverse object categories. Notably, the full model shows consistent improvements across almost all categories, with particularly large gains in challenging objects, such as basketball court (+26.8%), vehicle (+22.9%), and harbor (+9.9%). This suggests that our approach effectively addresses the limitations of existing methods, especially for objects with complex structures or in cluttered environments.

D. Qualitative and Quantitative Comparison With SOTA Methods

Tables III and IV present a comprehensive comparison between our proposed method and several state-of-the-art object detection approaches on the NWPU-VHR10 dataset. The comparison encompasses both efficiency metrics (inference time, parameter count, and computational complexity) and detection accuracy across all ten object categories.

In terms of efficiency (see Table III), our method demonstrates a favorable balance between computational resources and performance. With only 2.54 M parameters (the lowest among all compared methods except YOLOv10n) and 6.4 G FLOPs (second only to YOLOv11n's 6.3 G), our approach maintains competitive inference speed at 7.6 ms. This positions our method as a highly efficient solution for remote sensing object detection, particularly suitable for resource-constrained applications.

Regarding detection accuracy, our method consistently outperforms all competitors across the first four object categories (airplane, ship, storage tank, and baseball diamond), as shown in Table III, with AP scores of 99.5%, 96.5%, 99.5%, and 99.1%, respectively. This trend continues in Table IV, where our approach demonstrates superior performance in the remaining six categories, particularly showing significant improvements in challenging objects, such as basketball court and vehicle compared to lightweight models, such as YOLOv8n and YOLOv10n.

The overall mAP@0.5 of our method reaches 93.7%, substantially outperforming all compared approaches, with the next best being YOLOv9s at 89.4%. This represents a significant improvement of 4.3 percentage points over the strongest baseline. These results validate the effectiveness of our proposed components (CA, DyT, and WIoU) in addressing the unique challenges of remote sensing object detection while maintaining computational efficiency.

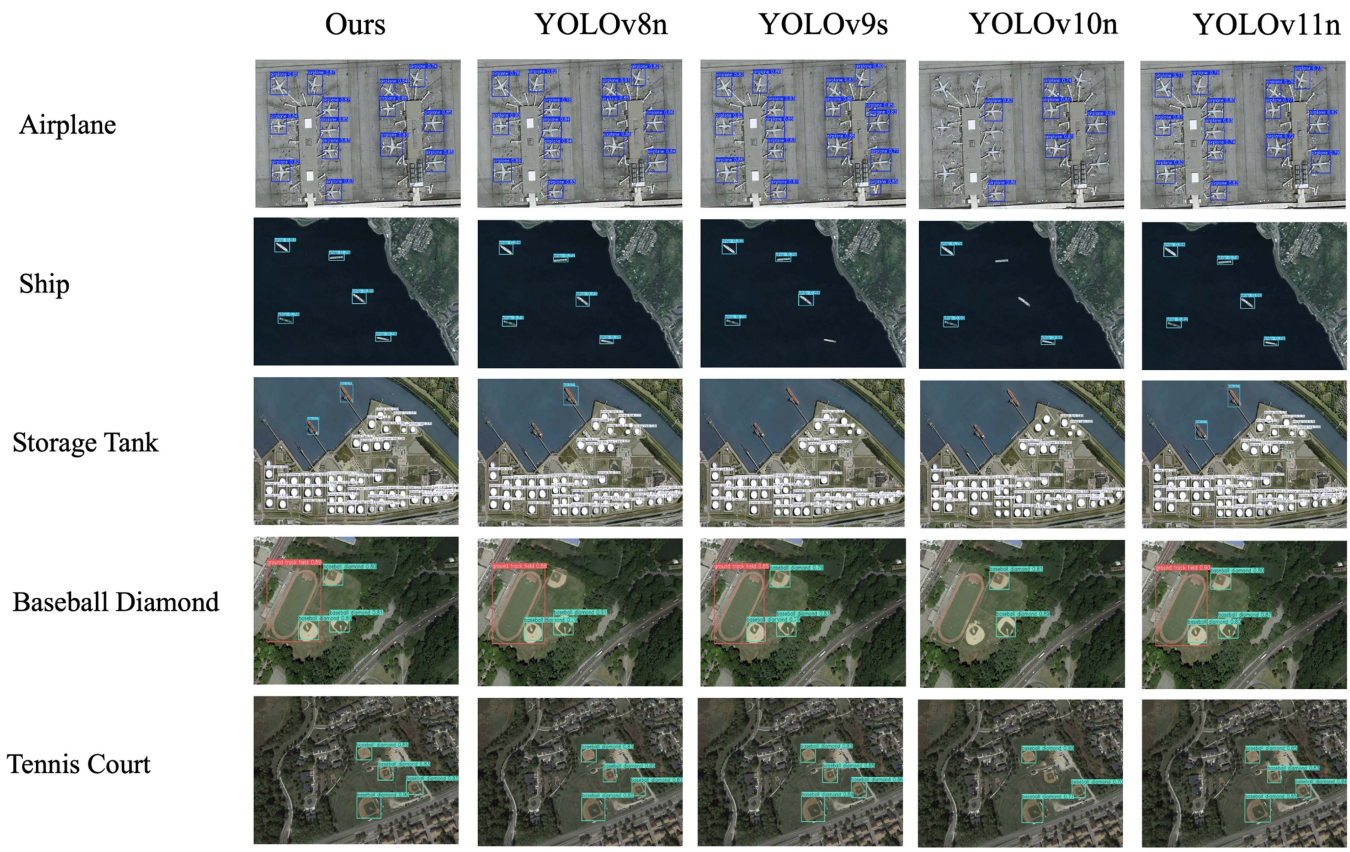


Fig. 7. Multiobject detection results on NWPU-VHR 10 dataset. (Part 1).

We present qualitative detection results on ten remote-sensing object categories using four YOLO backbone versions and our enhanced model. Each scene is processed by our scheme and by the baseline YOLOv8n, YOLOv9s, YOLOv10n, and YOLOv11n variants. The first figure illustrates detections for airplanes, ships, storage tanks, baseball diamonds, and tennis courts on airport ramp and park settings. The second figure shows results on basketball courts, ground track fields, harbors, bridges, and road vehicles in more cluttered urban landscapes.

As shown in Fig. 7, in the airport and park scenes our enhanced model consistently produces bounding boxes that align tightly with aircraft and storage tank outlines, capturing even partially occluded instances with high confidence. Baseline detectors often miss low-contrast airplanes parked under shadows or produce loose boxes around circular storage tanks. Tennis courts and baseball diamonds are detected by all models but ours delivers the cleanest localization by avoiding spurious boxes on nearby pavements and greenery.

The urban and waterfront scenes pose a greater challenge due to small object size and complex backgrounds. As is shown in Fig. 8, our model recovers every basketball court outline, even when court lines are faded or obscured by trees. It encloses the ground track field with precise corners instead of the loose fits seen in the baseline outputs. In harbor scenes our scheme separates adjacent boats and clearly delineates dock edges without merging nearby vessels. Bridge spans are fully covered by coherent boxes, while the baseline models frequently fragment long structures into multiple detections. On busy roads our model locates more vehicles and places tighter boxes around each car, addressing the common issue of missed or merged detections.

These visual comparisons underscore how combining CA, dynamic tanh transformations and WIoU loss leads to stronger recall and sharper localization across architectures. The largest gains appear on small, crowded or irregularly shaped objects where vanilla YOLO struggles most. Consistent improvements on both older and newer backbones demonstrate the general applicability of our enhancements. By refining feature maps and bounding-box regression our approach delivers robust and accurate detection in challenging high-resolution aerial imagery.

E. Generalization Evaluation of the Proposed Scheme Across SOTA YOLO Variants

To assess the effect of integrating CA, DyT, and WIoU loss into the YOLO detection framework, we conducted experiments on the generalization split of the NWPU-VHR10 remote sensing dataset using four backbone versions from eight through 11. We compared the original YOLO model against the enhanced variant in terms of overall mAP and per-class AP across ten object categories. The first figure presents the aggregate mAP improvements achieved by each backbone, while the second

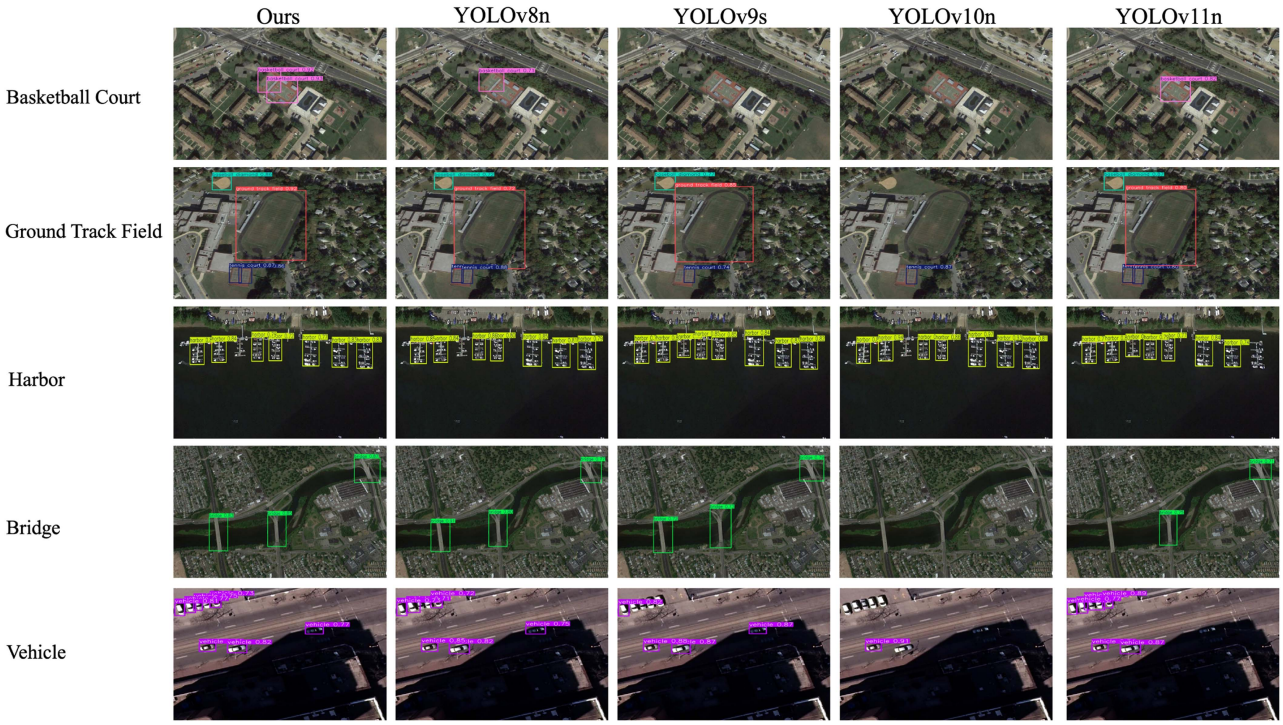


Fig. 8. Multiobject detection results on NWPU-VHR 10 dataset. (Part 2).

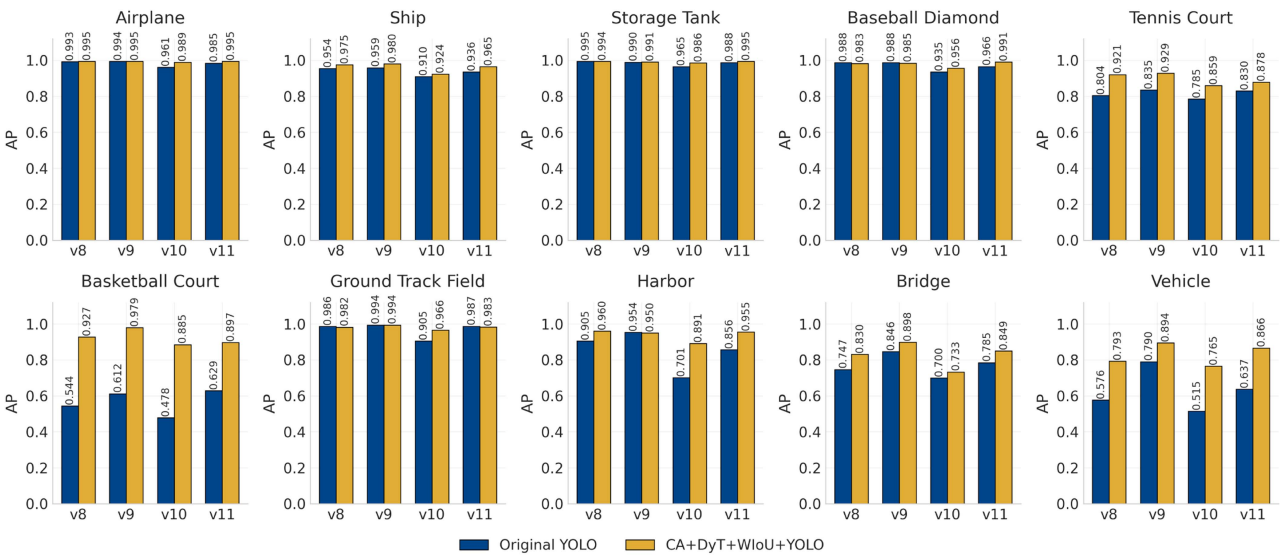


Fig. 9. AP comparison of the baseline YOLO detectors and the proposed CA+DyT+WIoU-enhanced YOLO models across ten remote-sensing object categories.

figure illustrates the AP changes for each category and highlights the areas that benefit most from our enhancements.

In Fig. 9, a breakdown of per-class AP for the ten remote sensing categories—airplane, ship, storage tank, baseball diamond, tennis court, basketball court, ground track field, harbor, bridge, and vehicle—reveals that the most challenging classes receive the largest benefits. Vehicle detection on version ten increases from 51.5% to 76.5%. Harbor and bridge categories each show improvements exceeding 10.0% in AP. By contrast, categories with near-saturating baseline performance, such as airplane, storage tank, and ground track field exhibit more modest gains of around 2%–3%, reflecting a diminishing-returns effect when initial AP is already very high.

As is shown in Fig. 10, the enhanced model significantly outperforms its vanilla counterpart on every backbone. On version eight, mAP rises from 84.9% to 93.6%, an absolute gain of 8.7 percentage points. Version nine's baseline mAP of 89.4% increases to 96.0% after enhancement, yielding a gain of 6.6 points. Version ten, experiences the most pronounced improvement, climbing from 78.5% to 89.5% for an 11.0-point increase. Even the recent architecture, version eleven, improves from 86.0% to 93.7%, achieving a gain of 7.7 points. These

TABLE V
PERFORMANCE COMPARISON OF YOLO MODELS ON CATEGORIES 1–10 IN THE DIOR DATASET. (PART1)

Method	Airplane	Airport	Baseball field	Basketball court	Bridge	Chimney	Dam	Expressway service area	Expressway toll station	Golf course	**mAP@50**
YOLOv8 [32]	74.6	67.0	99.5	45.0	59.6	99.5	59.7	65.3	99.5	46.9	70.5
YOLOv8+	76.2	73.6	99.5	67.4	59.7	99.5	78.6	69.1	99.5	77.0	78.9
YOLOv9 [33]	78.0	58.7	99.5	59.5	54.6	99.5	45.8	57.7	99.5	61.8	73.0
YOLOv9+	80.7	70.3	99.2	70.5	79.8	99.5	80.6	79.7	99.5	75.2	81.9
YOLOv10 [34]	69.1	57.9	99.0	52.8	55.3	93.3	60.0	70.8	99.5	56.7	68.9
YOLOv10+	73.2	63.1	99.5	57.2	54.6	99.5	65.1	75.5	99.5	46.5	75.2
YOLOv11 [35]	71.8	50.0	100.0	41.1	60.0	100.0	60.0	61.2	90.4	42.9	67.8
YOLOv11+	75.4	62.2	99.5	66.4	64.3	99.5	76.2	67.5	99.5	67.0	79.0

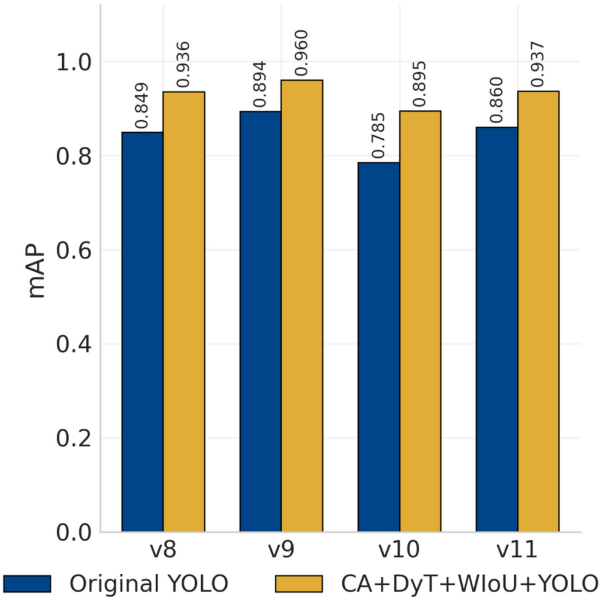


Fig. 10. mAP comparison between the baseline YOLO detectors and the CA+DyT+WIoU-enhanced YOLO models.

results demonstrate that combining CA, DyT transformation, and WIoU loss leads to consistent and substantial performance gains, regardless of backbone age or design.

Overall, our scheme leverages CA to focus on to focus on wider and deeper feature maps, DyT to introduce adaptive nonlinear transformations and WIoU to refine bounding-box regression. This combination markedly improves YOLO's generalization ability at both the global and per-class levels. The enhancements are especially pronounced for detecting small, densely packed or irregularly shaped objects, thereby offering robust performance in complex real-world scenarios.

F. Generalization Evaluation of the Proposed Scheme on the DIOR Dataset

To further validate the generalizability of our proposed enhancements, we conducted comprehensive experiments on the DIOR dataset, which presents additional challenges with its diverse object categories and varying imaging conditions [31]. The DIOR dataset contains 20 object classes, providing a robust testbed for evaluating the transferability of the proposed improvement scheme.

Tables V and VI present a detailed comparison of baseline YOLO models (YOLOv8, YOLOv9, YOLOv10, YOLOv11) against their enhanced counterparts (denoted with "+") across all 20 DIOR object categories. The results demonstrate consistent and substantial improvements across all backbone architectures, with the enhanced models achieving superior performance in both individual class AP and overall mAP (mAP@50).

Examining the per-class performance improvements, several patterns emerge that highlight the effectiveness of our proposed modifications. Categories with complex geometric structures or small object instances show the most pronounced gains. For instance, basketball court detection improves dramatically across all models, with YOLOv9+ achieving 70.5% AP compared to 59.5% for the baseline YOLOv9, representing an 11-point improvement. Similarly, expressway service area detection benefits significantly from our enhancements, with YOLOv9+ reaching 79.7% AP versus 57.7% for the baseline, demonstrating the enhanced model's superior capability in handling irregularly shaped infrastructure objects.

The bridge category consistently shows substantial improvements across all architectures, with YOLOv9+ achieving 79.8% AP compared to 54.6% for the baseline YOLOv9. This 25.2-point improvement underscores how CA helps the model focus on linear structures that span across image regions, while DyT provides adaptive feature transformations that better capture the varying appearances of bridges in different contexts. Golf course detection also benefits markedly, with improvements ranging from 10 to 20 percentage points across different YOLO versions, indicating enhanced capability in detecting large, textured areas with subtle boundaries.

Analyzing the overall mAP@50 improvements, YOLOv9+ demonstrates the most substantial gain, increasing from 73.0% to 81.9% (8.9-point improvement), followed by YOLOv11+ with an improvement from 67.8% to 79.0% (11.2-point improvement). YOLOv8+ and YOLOv10+ also show consistent gains of 8.4 and 6.3 percentage points, respectively. These results confirm that our proposed enhancements provide robust and transferable improvements across different YOLO architectures and datasets, validating the generalizability of the CA, DyT, and WIoU integration for remote sensing object detection tasks.

YOLO models (YOLOv8, YOLOv9, YOLOv10, YOLOv11) are evaluated against their improved counterparts (YOLOv8+, YOLOv9+, YOLOv10+, YOLOv11+) across all categories of the DIOR dataset. Due to the considerable length of the resulting

TABLE VI
PERFORMANCE COMPARISON OF YOLO MODELS ON CATEGORIES 11–20 IN THE DIOR DATASET. (PART2)

Method	Ground track field	Harbor	Overpass	Ship	Stadium	Storage tank	Tennis court	Train station	Vehicle	Wind mill	**mAP@50**
YOLOv8 [32]	72.6	61.4	66.1	96.0	99.5	78.7	87.9	24.5	34.1	72.9	70.5
YOLOv8+	80.9	63.8	68.0	98.3	99.5	79.7	95.9	58.5	44.7	89.1	78.9
YOLOv9 [33]	87.0	57.1	65.3	98.2	97.2	79.2	92.2	41.3	41.8	86.1	73.0
YOLOv9+	90.4	63.4	62.9	98.8	99.5	80.6	97.1	61.0	50.0	98.3	81.9
YOLOv10 [34]	53.6	57.6	56.3	93.8	99.5	74.6	89.8	27.1	34.2	77.1	68.9
YOLOv10+	88.9	52.0	67.8	96.5	99.5	80.3	96.7	68.4	40.7	79.1	75.2
YOLOv11 [35]	63.6	62.1	63.6	96.9	100.0	73.5	80.5	40.0	28.4	70.0	67.8
YOLOv11+	95.9	59.1	65.0	98.4	99.5	80.6	95.8	70.7	42.2	96.1	79.0

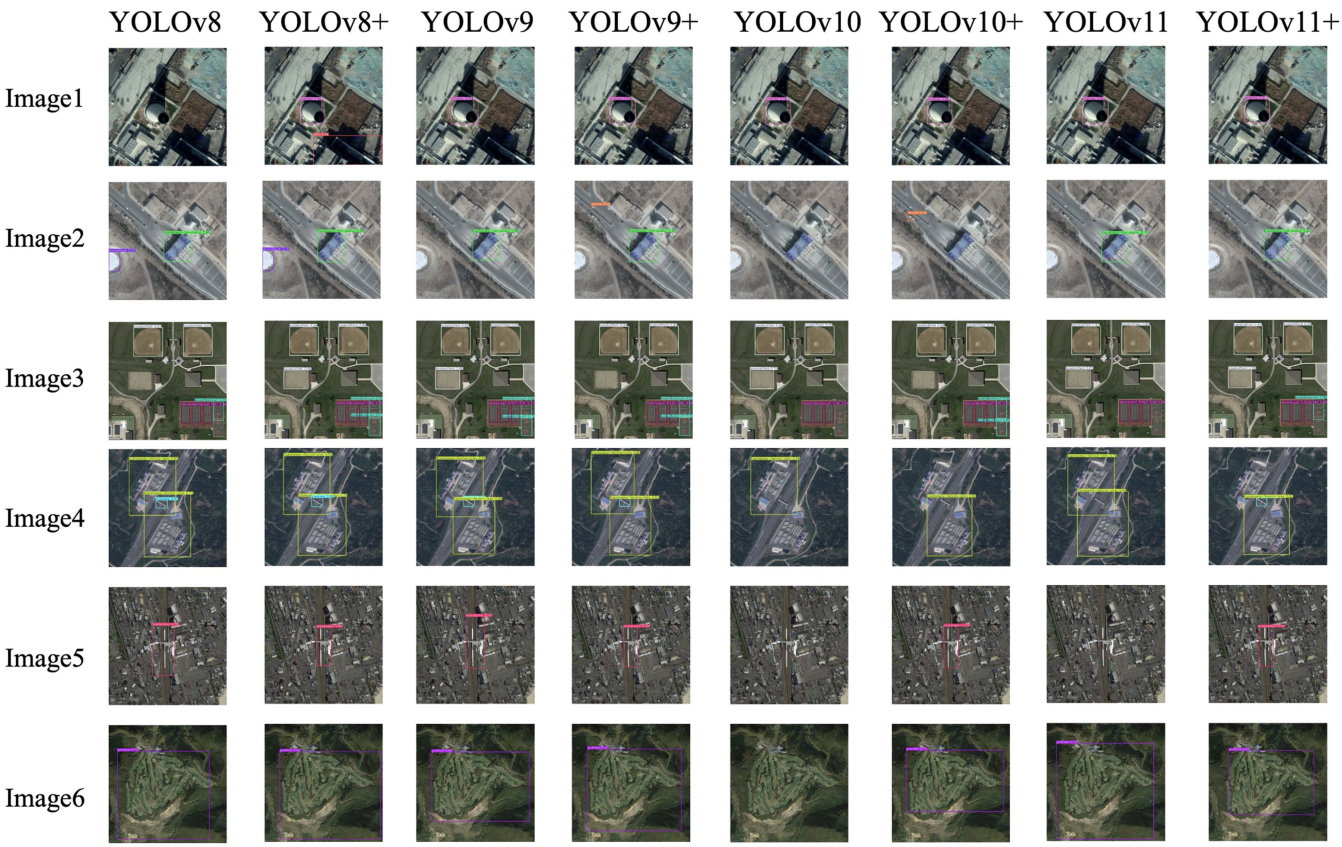


Fig. 11. Performance comparison of YOLO models across all categories of the DIOR dataset. (Part 1).

visualizations, the evaluation is presented in two separate Figs. 11 and 12. Fig. 11 compares YOLO models (YOLOv8, v9, v10, v11) against their improved counterparts (YOLOv8+, v9+, v10+, v11+) across six example images (images 1–6) from the DIOR dataset. Overall, we observe that the enhanced versions consistently exhibit improved detection performance.

Specifically, the enhanced YOLO variants (particularly YOLOv9+, YOLOv10+, and YOLOv11+) demonstrate superior detection accuracy for smaller or partially occluded objects in images 1 and 2, with bounding boxes that more precisely align with object boundaries. In images 3 and 4, the improved models significantly enhance the recall of smaller, densely clustered objects, resulting in fewer missed detections and reduced false predictions. Furthermore, the advanced YOLO models exhibit clearer object delineation in complex scenes or terrains with ambiguous boundaries, as evident in images 5 and 6. These enhancements reflect a refined capability to effectively address scale variations and background complexity.

Fig. 12 further extends the comparison with images 7 through 13, representing diverse scenarios and varying levels of detection complexity: Improved YOLO variants (particularly YOLOv10+ and YOLOv11+) achieve better precision in crowded environments. Objects are more consistently and precisely detected, suggesting improvements in addressing object overlap and clutter in images 7 and 8. YOLO improvements manifest clearly in enhanced bounding box accuracy and reduced false positives, particularly in scenarios featuring sparse or linear-structured objects in images 9 and 10. Improved YOLO models again demonstrate increased recall, notably detecting challenging small-scale objects that baseline models miss in images 11 and 12.

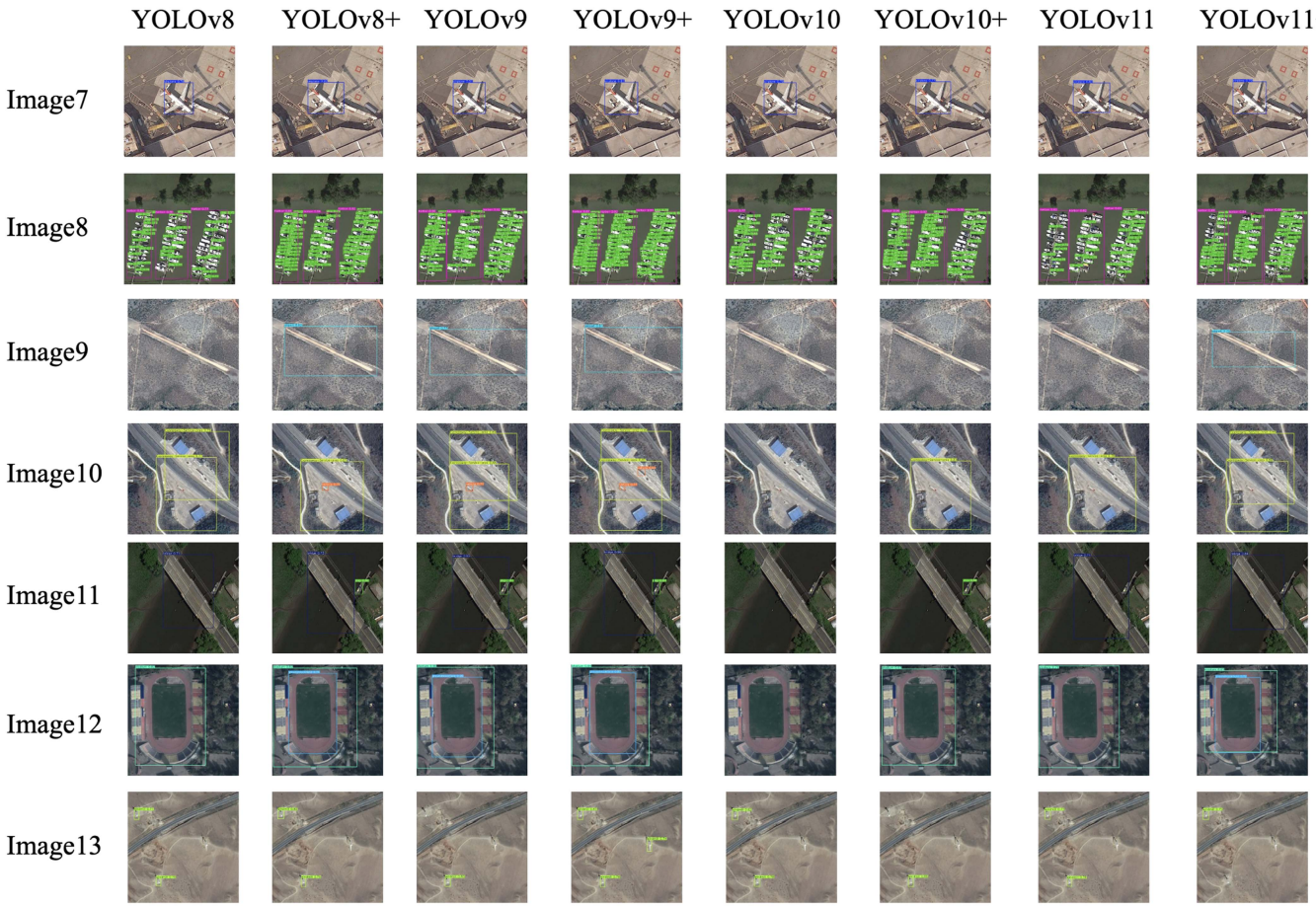


Fig. 12. Performance comparison of YOLO models across all categories of the DIOR dataset. (Part 2).

All enhanced YOLO models display marginal improvements, with better localization and fewer ambiguous detections in image 13.

VI. Conclusion

We have presented Lino-YOLO, a normalization-free, lightweight adaptation of the YOLO family tailored for high-resolution remote sensing. By embedding CA directly into feature maps, replacing batch normalization with a learnable DyT, and refining bounding-box regression via WIoU, Lino-YOLO requires only 2.54 million parameters and processes a 640 $\times$640 image in 7.6 ms on an RTX 4090. On the NWPU-VHR10 benchmark it achieves 93.7% mAP0.5—a 7.7-point improvement over the YOLOv11n baseline—while delivering the largest gains on small, densely packed and irregularly shaped objects.

Our comprehensive evaluation demonstrates Lino-YOLO's effectiveness across multiple dimensions: local variance analysis reveals superior training stability with faster convergence, PR curves show consistently higher precision across all recall values, attention heatmaps confirm enhanced spatial focus on critical object features, and ablation studies validate the synergistic contributions of each component. The backbone-agnostic design enables seamless integration across YOLO families (v8–v11), consistently delivering accuracy and speed improvements. Future work will evaluate Lino-YOLO on larger and more diverse aerial benchmarks, extend its normalization-free paradigm to transformer-based architectures, and explore further compression techniques for edge deployment scenarios. With code and pretrained weights released, we anticipate Lino-YOLO will serve as a robust foundation for efficient, accurate object detection in large-scale Earth-observation systems and real-time remote sensing applications.

VII. Discussion

This work introduces Lino-YOLO, a novel adaptation of the YOLO architecture specifically designed for HSR sensing object detection. Our comprehensive evaluation reveals several key insights about the synergistic interactions between the proposed components and their implications for the broader field of computer vision.

A. Component Interactions and Architectural Innovations

The integration of CA, DyT, and WIoU represents a carefully orchestrated enhancement to the YOLO architecture. Our local

variance analysis demonstrates that while the combination of CA and DyT initially introduces higher optimization fluctuations, it ultimately leads to faster convergence and superior stability. This apparent paradox can be explained by the complementary nature of these components: CA enhances spatial feature representation by embedding positional information directly into feature maps, while DyT provides adaptive normalization that responds dynamically to the varying statistical properties of remote sensing imagery.

The elimination of batch normalization through DyT is particularly significant for remote sensing applications, where batch statistics may not adequately capture the diverse spectral and spatial characteristics of aerial imagery. Our results show that this normalization-free approach not only reduces computational overhead but also improves training stability, as evidenced by the rapid variance reduction after initial epochs. The WIoU component further refines this synergy by providing more accurate bounding box regression, particularly crucial for the small and densely packed objects common in high-resolution remote sensing datasets.

B. Implications for Remote Sensing Object Detection

The 7.7-point mAP@0.5 improvement over YOLOv11n baseline, achieved with only 2.54 million parameters, represents a significant advancement in the efficiency–accuracy tradeoff for remote sensing applications. The PR curve analysis reveals that Lino-YOLO maintains precision above 0.9 even at high recall rates, indicating robust performance across diverse object categories and scales. This is particularly valuable for operational remote sensing systems where both comprehensive object coverage and high confidence detections are essential.

The attention heatmap visualizations confirm that our architectural modifications successfully enhance spatial focus on critical object features, addressing a key challenge in remote sensing where objects of interest often occupy small portions of large images. The backbone-agnostic design ensures that these improvements can be readily adopted across different YOLO variants, providing a flexible foundation for various deployment scenarios.

C. Broader Impact and Innovation Significance

Our work contributes to the growing body of research on efficient deep learning architectures for resource-constrained environments. By integrating DyT, CA, and WIoU into a unified scheme within the YOLO framework, we leverage the normalization-free paradigm introduced by DyT, which challenges conventional assumptions about the necessity of batch normalization in modern CNN architectures. The achieved inference time of 7.6 ms on RTX 4090 hardware demonstrates the practical viability of our combined approach for real-time applications, which is crucial for time-sensitive remote sensing tasks, such as disaster response and environmental monitoring [42].

The consistent improvements across YOLO families (v8–v11) validate the generalizability of our design principles, suggesting that the identified architectural patterns may be applicable to other object detection frameworks beyond YOLO. This cross-family compatibility is particularly valuable for practitioners who need to adapt existing systems without complete architectural overhauls.

D. Limitations and Future Directions

While Lino-YOLO demonstrates significant improvements in remote sensing object detection, several limitations warrant consideration for future research. The current evaluation focuses primarily on datasets with limited object categories—NWPU-VHR10 contains ten classes and DIOR contains 20 classes—whereas real-world remote sensing applications often require detection of more object types [43]. This scalability challenge becomes particularly pronounced when considering the vast diversity of infrastructure, vehicles, natural features, and humanmade structures visible in high-resolution satellite and aerial imagery. In addition, the class imbalance issues inherent in remote sensing data become more severe as the number of categories increases, potentially affecting the effectiveness of existing methods.

Future research directions could explore the transition from Euclidean to hyperbolic geometric spaces for multicategory object detection. This would leverage the superior hierarchical properties and search efficiency of hyperbolic geometry to enhance performance in complex multiclass remote sensing scenarios [44], [45]. This exploration could include developing hyperbolic embeddings for category relationships to better capture the natural taxonomic structures inherent in remote sensing object classes, creating hyperbolic-aware attention mechanisms that exploit the exponential growth properties of hyperbolic space for more efficient feature representation. Such investigations could unlock new paradigms for handling the inherent hierarchical nature of remote sensing data while maintaining the computational efficiency for real-world deployment.

While our experiments demonstrate the effectiveness of the proposed modules (CA, DyT, and WIoU) across various YOLO variants, their modular design suggests potential applicability to other object detection frameworks, such as RetinaNet and CenterNet. CA, as a lightweight spatial attention mechanism, can be readily integrated into the feature extraction backbones of diverse detectors to enhance spatial awareness [21]. Similarly, DyT serves as a normalization alternative that could replace batch normalization layers in any CNN-based architecture, potentially improving efficiency in models, such as RetinaNet, building on its recent success in eliminating normalization in transformer-based systems. WIoU, being a bounding box regression loss, is agnostic to the specific detection paradigm and could optimize localization in keypoint-based detectors, such as CenterNet, with similar IoU variants already explored in dense object detection contexts including RetinaNet and ATSS. Preliminary literature supports this generalization potential, and future investigations will explore these integrations to broaden the impact of our approach.

Although the current evaluation focuses on standard computing environments, the lightweight nature of Lino-YOLO, with only 2.54 million parameters and 6.4 GFLOPs, positions it well for deployment on edge devices, such as NVIDIA Jetson TX2

and AGX Xavier, which are designed for real-time AI processing in resource-constrained scenarios, such as remote sensing drones or satellites. Existing frameworks, such as ultralytics for YOLO models, provide straightforward pipelines to export models to TensorRT format, enabling optimized inference on Jetson hardware with minimal latency. For instance, similar YOLO variants have been successfully deployed on Jetson TX2 for real-time object detection, achieving viable frame rates through TensorRT acceleration, despite the device's limited 256 CUDA cores and 8 GB memory. On the more powerful AGX Xavier, with up to 512 CUDA cores and higher TOPS, Lino-YOLO could leverage enhanced parallel processing for even faster inference in high-resolution imagery. This edge deployment capability would demonstrate the model's practical utility for on-device processing in remote areas, reducing reliance on cloud infrastructure; future work will include empirical benchmarks on these platforms to quantify performance metrics, such as FPS and power consumption.

REFERENCES

[1] X. Zhang et al., "Remote sensing object detection meets deep learning: A metareview of challenges and advances," *IEEE Geosci. Remote Sens. Mag.*, vol. 11, no. 4, pp. 8–44, Dec. 2023.

[2] Y. Wang, C. Wang, H. Zhang, Y. Dong, and S. Wei, "Automatic ship detection based on retinanet using multi-resolution Gaofen-3 imagery," *Remote Sens.*, vol. 11, no. 5, 2019, Art. no. 531.

[3] J. Zhang, X. Jia, J. Hu, and K. Tan, "Moving vehicle detection for remote sensing video surveillance with nonstationary satellite platform," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5185–5198, Sep. 2022.

[4] Y. Bazi and F. Melgani, "Convolutional SVM networks for object detection in UAV imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3107–3118, Jun. 2018.

[5] Y. Li, B. He, F. Melgani, and T. Long, "Point-based weakly supervised learning for object detection in high spatial resolution remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 5361–5371, 2021.

[6] Z. Zheng, Y. Zhong, J. Wang, A. Ma, and L. Zhang, "FarSeg++: Foreground-aware relation network for geospatial object segmentation in high spatial resolution remote sensing imagery," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 13715–13729, Nov. 2023.

[7] K. Tang, F. Xu, X. Chen, Q. Dong, Y. Yuan, and J. Chen, "The ClearSCD model: Comprehensively leveraging semantics and change relationships for semantic change detection in high spatial resolution remote sensing imagery," *ISPRS J. Photogrammetry Remote Sens.*, vol. 211, pp. 299–317, 2024.

[8] M. Chi, A. Plaza, J. A. Benediktsson, Z. Sun, J. Shen, and Y. Zhu, "Big data for remote sensing: Challenges and opportunities," *Proc. IEEE*, vol. 104, no. 11, pp. 2207–2219, Nov. 2016.

[9] D. Lunga, J. Gerrand, L. Yang, C. Layton, and R. Stewart, "Apache spark accelerated deep learning inference for large scale satellite image analytics," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 271–283, 2020.

[10] C. Kyrkou and T. Theocharides, "EmergencyNet: Efficient aerial image classification for drone-based emergency monitoring using atrous convolutional feature fusion," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1687–1699, 2020.

[11] A. Magdy, M. S. Moustafa, H. M. Ebied, and M. F. Tolba, "Lightweight faster R-CNN for object detection in optical remote sensing images," *Sci. Rep.*, vol. 15, no. 1, 2025, Art. no. 16163.

[12] P. Mittal, "A comprehensive survey of deep learning-based lightweight object detection models for edge devices," *Artif. Intell. Rev.*, vol. 57, no. 9, 2024, Art. no. 242.

[13] H. Liao and W. Zhu, "YOLO-DRS: A bioinspired object detection algorithm for remote sensing images incorporating a multi-scale efficient lightweight attention mechanism," *Biomimetics*, vol. 8, no. 6, 2023, Art. no. 458.

[14] X. Ma, K. Ji, B. Xiong, L. Zhang, S. Feng, and G. Kuang, "Light-YOLOv4: An edge-device oriented target detection method for remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 10808–10820, 2021.

[15] Z. Zhou et al., "HRLE-SARDet: A lightweight SAR target detection algorithm based on hybrid representation learning enhancement," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5203922.

[16] T. Zhang, Y. Zhuang, G. Wang, H. Chen, L. Li, and J. Li, "A unified remote sensing object detector based on fourier contour parametric learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, 2025, Art. no. 5611225.

[17] T. Zhang et al., "Controllable generative knowledge driven few-shot object detection from optical remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, 2025, Art. no. 5612319.

[18] L. Ge et al., "Regression-guided refocusing learning with feature alignment for remote sensing tiny object detection," *IEEE Trans. Geosci. remote Sens.*, vol. 62, 2024, Art. no. 4408314.

[19] S. Zhang, C. Qu, C. Ru, X. Wang, and Z. Li, "Multi-objects recognition and self-explosion defect detection method for insulators based on lightweight ghostnet-YOLOV4 model deployed onboard UAV," *IEEE Access*, vol. 11, pp. 39713–39725, 2023.

[20] J. Chen, K. Chen, H. Chen, Z. Zou, and Z. Shi, "A degraded reconstruction enhancement-based method for tiny ship detection in remote sensing images with a new large-scale dataset," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5625014.

[21] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 13 713–13 722.

[22] J. Zhu, X. Chen, K. He, Y. LeCun, and Z. Liu, "Transformers without normalization," in *Proc. Comput. Vis. Pattern Recognit. Conf.*, 2025, pp. 14901–14911.

[23] Z. Tong, Y. Chen, Z. Xu, and R. Yu, "Wise-IoU: Bounding box regression loss with dynamic focusing mechanism," *arXiv:2301.10051*, 2023.

[24] A. Brock, S. De, S. L. Smith, and K. Simonyan, "High-performance large-scale image recognition without normalization," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 1059–1071.

[25] Z. Zhang, W. Lu, J. Cao, and G. Xie, "MKANet: An efficient network with sobel boundary loss for land-cover classification of satellite remote sensing imagery," *Remote Sens.*, vol. 14, no. 18, 2022, Art. no. 4514.

[26] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 658–666.

[27] Z. Zheng et al., "Distance-IoU loss: Faster and better learning for bounding box regression," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 07, pp. 12993–13000.

[28] G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," *ISPRS J. Photogrammetry Remote Sens.*, vol. 117, pp. 11–28, 2016.

[29] G. Cheng, J. Han, P. Zhou, and L. Guo, "Multi-class geospatial object detection and geographic image classification based on collection of part detectors," *ISPRS J. Photogrammetry Remote Sens.*, vol. 98, pp. 119–132, 2014.

[30] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.

[31] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, "Object detection in optical remote sensing images: A survey and a new benchmark," *ISPRS J. Photogrammetry Remote Sens.*, vol. 159, pp. 296–307, 2020.

[32] H. Li and J. Wu, "LSOD-YOLOv8s: A lightweight small object detection model based on YOLOv8 for UAV aerial images," *Eng. Lett.*, vol. 32, no. 11, 2024, Art. no. 126440.

[33] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, "YOLOv 9: Learning what you want to learn using programmable gradient information," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2024, pp. 1–21.

[34] A. Wang et al., "YOLOv10: Real-time end-to-end object detection," *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 107984 108011.

[35] R. Khanam and M. Hussain, "Yolov11: An overview of the key architectural enhancements," 2024, *arXiv:2410.17725*.

[36] S. Gui, S. Song, R. Qin, and Y. Tang, "Remote sensing object detection in the deep learning era—a review," *Remote Sens.*, vol. 16, no. 2, 2024, Art. no. 327.

[37] A. A. Aleissaee et al., "Transformers in remote sensing: A survey," *Remote Sens.*, vol. 15, no. 7, 2023, Art. no. 1860.
[38] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers and distillation through attention," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 10347–10357.
[39] Y.-A. Chung, Y. Wang, W.-N. Hsu, Y. Zhang, and R. Skerry-Ryan, "Semi-supervised training for improving data efficiency in end-to-end speech synthesis," in *Proc. ICASSP 2019-2019 IEEE Int. Conf. Acoust., Speech Signal Process.*. IEEE, 2019, pp. 6940–6944.
[40] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Eur. Conf. Computer Vis.*. Springer, 2020, pp. 213–229.
[41] D. Reis, J. Kupec, J. Hong, and A. Daoudi, "Real-time flying object detection with YOLOv8," 2024, *arXiv:2305.09972*.
[42] H. Rehan, "Enhancing disaster response systems: Predicting and mitigating the impact of natural disasters using AI," *J. Artif. Intell. Res.*, vol. 2, no. 1, 2022, Art. no. 501.
[43] D. Lam et al., "xView: Objects in context in overhead imagery," 2018, *arXiv:1802.07856*.
[44] T. Long, P. Mettes, H. T. Shen, and C. G. M. Snoek, "Searching for actions on the hyperbole," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 1138–1147.
[45] C. Lang, A. Braun, L. Schillingmann, and A. Valada, "On hyperbolic embeddings in object detection," in *DAGM German Conf. Pattern Recognit.*. Springer, 2022, pp. 462–476.

Youyou Li received the B.S. degree and the Ph.D. degree in remote sensing from the University of Electronic Science and Technology of China, Chengdu, China, in 2014 and 2021, respectively.

She is currently a Faculty Member with the Civil Aviation Flight University of China, Deyang, China. She is also a Reviewer for several international journals, including IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, INTERNATIONAL JOURNAL OF REMOTE SENSING, etc. Her research interests include target recognition in remote sensing imagery, artificial intelligence algorithms, and aircraft trajectory prediction.

Yuxiang Fang is currently working toward the master 's degree in transportation engineering with the Civil Aviation Flight University of China, Deyang, China.

His research focuses on the application of deep learning techniques to trajectory prediction and target detection, with an interest in exploring innovative approaches to enhance aviation safety and efficiency. He is dedicated to advancing his knowledge and contributing to the field through his studies and ongoing research projects.

Shixiong Zhou is currently working toward the B.Eng. degree in transportation (air traffic management) with the Civil Aviation Flight University of China, Deyang, China.

His research focuses on the field of air traffic management.

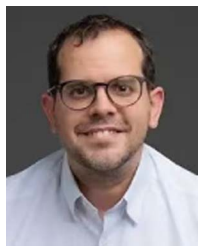
Teng Long received the Ph.D. degree in computer science from the University of Electronic Science and Technology of China, Chengdu, China, in 2020.

He is currently a Researcher with Informatics Institute, Faculty of Science, University of Amsterdam, Amsterdam, The Netherlands, under the joint mentorship of Andrew Yates (Johns Hopkins University) and Nicu Sebe (University of Trento). He contributes to the VIDI (IDEAS: Incremental Dense Representations) project supported by NWO, and the EU-funded ELIAS (European Lighthouse of AI for Sustainability) project. His research interests include hyperbolic learning, multimodal information retrieval, and multimodal foundational learning.

Yicheng Zhang (Member, IEEE) received the Ph.D. degree in electrical and electronic engineering from Nanyang Technological University, Singapore, in 2019.

He is currently a Senior Scientist with the Institute for Infocomm Research (I2R), Agency for Science, Technology and Research, Singapore (A∗STAR). Before joining I2R, he was a Research Associate affiliated with Rolls-Royce @ NTU Corp Lab, Singapore. He has participated in many industrial and research projects funded by National Research Foundation Singapore, A∗STAR, Land Transport Authority, and Civil Aviation Authority of Singapore. He has authored or coauthored more than 90 research papers in journals and pier-reviewed conferences.

Dr. Zhang was the recipient of the IEEE Intelligent Transportation Systems Society (ITSS) Young Professionals Traveling Scholarship in 2019 during IEEE ITSC, and as a Team Member, was the recipient of the Singapore Public Sector Transformation Award in 2020.



Nuno Antunes Ribeiro received the Ph.D. degree in transport systems from the University of Coimbra in 2019. He is an Assistant Professor in engineering systems and design pillar with the Singapore University of Technology and Design, where he is also a Deputy Director of the Aviation Studies Institute. He is Vice-President of the Aviation Applications Section of INFORMS. His research interests include operations research, simulation, machine learning, and econometrics to transportation system management, with a focus on integrating optimization and data-driven methods. His work addresses challenges in air transportation, including airport slot allocation, passenger connectivity, resilient airspace operations, and the impact of convective weather.

Farid Melgani (Fellow, IEEE) received the State Engineer degree in electronics from the University of Batna, Batna, Algeria, in 1994, the M.Sc. degree in electrical engineering from the University of Baghdad, Baghdad, Iraq, in 1999, and the Ph.D. degree in electronic and computer engineering from the University of Genoa, Genoa, Italy, in 2003.

He is currently a Full Professor of telecommunications with the Department of Information Engineering and Computer Science, University of Trento, Trento, Italy, where he teaches pattern recognition and machine learning. He is the Head of the Signal Processing and Recognition Laboratory, University of Trento. He has coauthored more than 300 scientific publications. His research interests include the areas of remote sensing, signal/image processing and artificial intelligence.

Dr. Melgani is an Associate Editor for the *International Journal of Remote Sensing* and IEEE JOURNAL ON MINIATURIZATION FOR AIR AND SPACE SYSTEMS.