MoRE-DETR: Adaptive Recursive Reasoning with Vision-Language Model for Enhancing Remote Sensing Object Detection

Anonymous authors
Paper under double-blind review

Abstract

Remote sensing object detection suffers from severe foreground-background imbalance and substantial variations in object scale and density, making foreground tokens differ greatly in detection difficulty. Existing DETR-like detectors refine queries uniformly and thus under-refine hard objects, while VLM-assisted variants either retain the VLM during inference or generate coordinates as text with weak geometric supervision. To address these issues, we propose MoRE-DETR, a DETR-based framework that couples adaptive token computation with VLM-assisted training on a shared token representation. Specifically, an Adaptive Recursive Token Routing (ARTR) module employs an uncertainty-aware Mixture-of-Recursions router to allocate recursive depth according to token difficulty, producing difficulty-adaptive foreground tokens. These refined tokens are fed to a Visual-Language Aligned Supervision (VLAS) module, which projects them into the semantic space of a multimodal large language model and directly regresses continuous bounding boxes from its hidden states, thereby injecting explicit geometric supervision back into the token representation. VLAS is used only during training and introduces no additional inference overhead. Extensive experiments on multiple remote sensing benchmarks demonstrate state-of-the-art performance. *Code will be publicly available upon acceptance.*

1 Introduction

Remote sensing object detection (RSOD) has attracted increasing research interest due to its importance in a wide range of applications, including urban management, disaster response, and environmental monitoring. However, RSOD remains challenging due to substantial variations in object scales, severe foreground-background imbalance, and the dense distribution of objects in complex aerial scenes. To address these challenges, many object detection methods have been developed, ranging from CNN-based detectors (Ren et al., 2015; Jocher et al., 2023) to more recent Detection Transformer (DETR)-based approaches Zhu et al. (2020); Li et al. (2022); Zhao et al. (2024). Compared with CNN-based detectors, Transformer architectures can capture long-range dependencies and model global contextual information, making DETR-based detectors an increasingly important paradigm for remote sensing object detection.

Existing DETR-based RSOD methods can be broadly grouped into three categories. Spatial feature enhancement methods (Huang et al., 2025c; Hu et al., 2025; Li et al., 2026a; Zhao et al., 2026) design dedicated attention or feature interaction modules to strengthen spatial representations of small and cluttered objects. Frequency-domain fusion methods (Zhi et al., 2025; Zhang et al., 2025; Jia et al., 2026; Chen et al., 2026) introduce spectral transforms to capture global context and suppress background noise. Vision foundation model based methods (Ding et al., 2026; Zhang et al., 2026b) leverage large-scale pretrained representations to improve generalization and semantic understanding. However, as shown in Fig. 1(a), these methods generally do not explicitly distinguish foreground from background tokens or allocate different computation according to detection difficulty, and instead rely on the model to learn which features are informative. Prior studies (Roh et al., 2021; Zheng et al., 2023) have shown that selecting or discarding a subset of tokens can maintain or even improve detection performance, suggesting that not all tokens need equal treatment.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

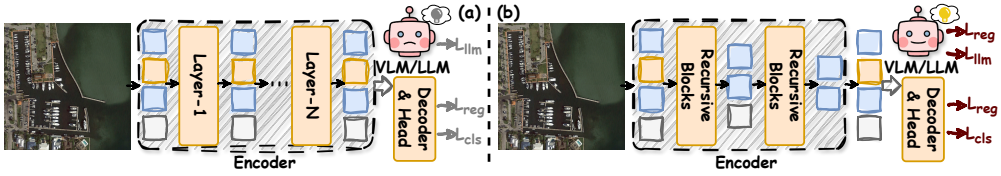

Figure 1: Comparison of state-of-the-art paradigms. (a) Existing RSOD methods treat foreground tokens uniformly and exploit VLMs mainly through text-based cross-entropy supervision. (b) Our approach couples adaptive computation with vision-language alignment, where ARTR refines challenging tokens and VLAS applies hidden-state box regression to them.

Injecting vision-language model (VLM) priors into object detection has recently emerged as a promising and increasingly explored direction, owing to the rich visual-semantic knowledge acquired from large-scale pretraining. In natural-image detection, existing methods either exploit vision-language alignment for open-vocabulary localization (Li et al., 2024; Wei et al., 2024; Pan et al., 2025; Huang et al., 2025d) or adapt pretrained VLMs through fine-tuning (Li et al., 2026c; Wu et al., 2026). In the field of closed-set RSOD, several recent works have further explored the integration of VLM priors into object detectors (Li et al., 2025a;b; 2026b; Shi et al., 2026). However, these methods still face two major limitations. Some retain VLM-related components during inference, such as text embedding dictionaries, introducing additional computational overhead, while others, as shown in Fig. 1(a), formulate localization as autoregressive text generation, representing continuous box coordinates as discrete tokens and supervising them with a language modeling loss. Such a formulation does not explicitly model the geometric relationships between coordinates, which limits the effectiveness of fine-grained localization supervision.

Motivated by the above analysis, as shown in Fig. 1(b), we propose MoRE-DETR, a DETR-based framework that couples adaptive token computation with VLM-assisted training on a shared token representation. The first module, Adaptive Recursive Token Reasoning (ARTR), introduces the Mixture-of-Recursions (MoR) mechanism Bae et al. (2026) into the DETR encoder: a foreground token selector separates foreground from background, and an uncertainty-aware MoR router allocates recursive computation according to token difficulty, refining challenging tokens while easy ones exit early. The resulting difficulty-adaptive tokens are then passed to the second module, Visual-Language Aligned Supervision (VLAS), which projects them into the semantic space of a large vision-language model and concatenates them with the prompt and serialized ground-truth tokens. The hidden states at dedicated localization tokens are decoded into continuous boxes via a lightweight MLP, and the resulting geometric supervision is back-propagated to the shared tokens, further shaping the representations produced by ARTR. In this way, ARTR determines where computation should be spent, while VLAS determines what supervision those tokens should receive, forming a closed loop between adaptive computation and vision-language alignment. Overall, MoRE-DETR improves small and difficult object detection in remote sensing imagery.

Our contributions can be summarized as three aspects. (1) **A training-time language-supervised MoR framework.** We are the first to introduce the Mixture-of-Recursions (MoR) paradigm into remote sensing object detection, where an uncertainty-aware router refines feature-based routing with semantic uncertainty to identify challenging foreground tokens. (2) **Recursive refinement via token interaction and context aggregation.** Difficult tokens are repeatedly refined through shared Transformer blocks, where self-attention enables mutual interaction and deformable attention aggregates multi-scale context, yielding more discriminative representations while easy tokens exit early. (3) **A supervision alignment mechanism coupled with adaptive token computation.** Unlike autoregressive textual supervision, we regress continuous boxes from VLM hidden states and apply this supervision to the difficulty-adaptive tokens produced by ARTR, so that vision-language alignment and recursive token refinement reinforce each other without inference overhead.

2 Related Work

DETR-based Remote Sensing Object Detection. DETR (Carion et al., 2020) pioneered a new paradigm for object detection by formulating detection as an end-to-end set prediction problem. Recently, an increasing number of DETR-based detectors have been proposed for remote sensing

object detection, which can be broadly categorized into three strategies. The first category (Huang et al., 2025c; Hu et al., 2025; Li et al., 2026a; Zhao et al., 2026) focuses on spatial-domain modeling, designing efficient feature enhancement modules and high-quality query selection mechanisms to improve object representation and localization. The second category (Zhi et al., 2025; Zhang et al., 2025; Jia et al., 2026; Chen et al., 2026) explores multi-domain context modeling, decomposing and filtering features in the frequency domain to preserve informative visual cues for objects with diverse scales and appearances. The third category leverages vision foundation models for their strong visual representation capabilities. For example, SKYDET (Zhang et al., 2026b) employs a frozen DINOv3 (Siméoni et al., 2025) backbone with a semantic adapter to obtain multi-scale representations. Despite their effectiveness, most existing methods adopt RT-DETR-style hybrid encoders (Zhao et al., 2024) rather than a purely ViT-style one, and do not explicitly distinguish foreground from background tokens or account for varying detection difficulty, relying on the model to implicitly learn which tokens are informative.

VLM-based Remote Sensing Object Detection. Vision-language models (VLMs) have recently shown strong visual understanding and cross-modal representation capabilities, motivating their application to remote sensing object detection. Existing VLM-based approaches can be grouped into three categories. The first (Li et al., 2024; Wei et al., 2024; Pan et al., 2025; Huang et al., 2025d) targets open-vocabulary detection by formulating localization as vision-language grounding, where objects are located according to text prompts. The second (Li et al., 2025a;b; 2026b; Shi et al., 2026) instead applies VLMs to closed-set remote sensing detection, exploiting the prior knowledge acquired from large-scale pretraining to compensate for the limited visual cues in aerial imagery. The third (Li et al., 2026c; Wu et al., 2026) adapts VLMs to the remote sensing domain through parameter-efficient fine-tuning, aiming to bridge the gap between natural-image pretraining and aerial scenes while preserving general visual-language knowledge. Despite their differences, these methods share two limitations. Some retain the VLM at inference, introducing additional computational and deployment overhead, while others formulate localization as autoregressive textual generation, representing continuous coordinates as discrete tokens without explicitly modeling their geometric relationships.

3 METHODOLOGY

3.1 PROBLEM STATEMENT

Remote sensing object detection aims to localize and categorize objects in aerial images. Let the training set be $D_{train} = (x^i_{tr}, y^i_{tr})^{N_{tr}}_{i=1}$, where $y^i_{tr} = (b^i_{tr}, c^i_{tr})$ denotes the bounding-box and class labels of objects in the i-th training image, and N_{tr} denotes the training-set size. Similarly, the testing set is $D_{test} = (x^i_{te}, y^i_{te})^{N_{te}}_{i=1}$, where N_{te} is the cardinality of testing set. There are K categories to be detected, and each image typically contains multiple categories. Our objective is to leverage the potential of vision-language models to enhance the detection accuracy of our DETR-based detector.

Overview. As illustrated in Fig. 2, following Focus-DETR (Zheng et al., 2023), we introduce two intertwined modules that share the same token flow. The Adaptive Recursive Token Reasoning (ARTR) module targets the uneven difficulty among foreground tokens: multi-scale maps are tokenized, a foreground selector retains informative tokens A^0, and an uncertainty-aware Mixture-of-Recursions router decides which tokens undergo additional recursion through shared Transformer blocks, so that harder tokens consume more computation. The resulting difficulty-adaptive tokens are then forwarded to the Vision-Language Aligned Supervision (VLAS) module, which exploits a pretrained VLM to supply auxiliary training supervision. These tokens are mapped into the VLM embedding space and concatenated with the prompt T_P and serialized ground-truth T_Y; the localization hidden states T_{H_i} are decoded into continuous boxes, and the induced geometric supervision flows back to the shared tokens, further shaping ARTR's representations. In this way, ARTR controls how much computation each token receives, whereas VLAS controls the supervision applied to them, yielding a closed loop between adaptive computation and vision-language alignment.

3.2 ADAPTIVE RECURSIVE TOKEN REASONING

Foreground objects in remote sensing images occupy only a small fraction of the image and vary greatly in scale, so tokens differ substantially in detection difficulty. Uniform computation thus wastes capacity on background and easy tokens while leaving ambiguous ones under-refined. To

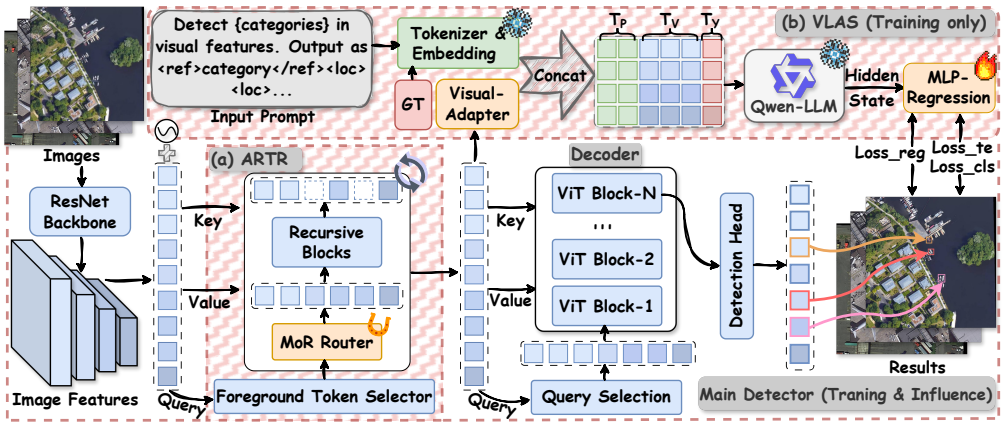

Figure 2: Overview of MoRE-DETR. (a) ARTR recursively refines only the most ambiguous foreground tokens via a foreground selector and an uncertainty-aware MoR router with shared Middle-Cycle blocks. (b) The VLM branch aligns a multimodal large language model's outputs with serialized GT during training to regularize visual features, and is removed at inference.

address this, we propose ARTR, which allocates recursive computation according to token difficulty in two stages. First, a foreground selector filters out background tokens, and an uncertainty-aware MoR router determines which foreground tokens continue recursion by combining feature-based importance with classification uncertainty. Second, a gated recursive token update refines the selected active tokens through shared "Middle-Cycle" blocks Bae et al. (2026), while inactive tokens exit early. We detail these two stages below.

Uncertainty-aware MoR router. We employ the Foreground Token Selector (Zheng et al., 2023) to filter out background tokens and obtain a set of high-confidence foreground tokens $A^0 \in \mathbb{R}^{M\times d}$, where M denotes the number of foreground tokens, and d denotes the dimensionality of each token. The selected tokens are then processed by a set of shared ViT blocks for up to R recursive steps. At each recursion step r, a router selects a subset of tokens to continue the recursion, while the remaining tokens exit early. Conventional MoR routers typically employ a lightweight MLP to estimate token importance solely from feature representations. In remote sensing detection, however, severe scale variation, cluttered backgrounds, and numerous tiny objects make many foreground tokens semantically ambiguous, and such feature-space scores cannot capture this category-level uncertainty, causing ambiguous tokens to exit recursion prematurely and remain under-refined. To incorporate category-level uncertainty into the routing process, we reuse the shared classification head $g(\cdot)$ of baseline to obtain the classification probabilities. Specifically, for the k-th active token A_k^r at recursion step r, its category-wise prediction $p^k \in \mathbb{R}^C$ is obtained as $p^k = \text{Sigmoid}(g(A_k^r))$, We then quantify the classification uncertainty of each token based on the binary entropy of its category-wise predictions. Specifically, we first compute the binary entropy for each category and then average and normalize them to obtain the uncertainty of the k-th token: $u_k = \sum_{c=1}^{C}[-p_c^k \log(p_c^k) - (1-p_c^k)\log(1-p_c^k)]/(C\log 2)$, where $u_k \in [0,1]$ denotes the classification uncertainty of the k-th token, with larger values indicating greater semantic ambiguity. The obtained uncertainty is then used to refine the original MLP-based routing score:

$$s_k = \text{MLP}_1\left(A_k^r\right) + \lambda u_k, \tag{1}$$

where λ is a learnable balancing coefficient. We select the tokens with routing scores s_k greater than a predefined threshold to the $(r+1)$-th recursion step, while the remaining tokens exit early. In this way, tokens with higher semantic uncertainty are more likely to receive additional recursive processing, allowing the model to allocate greater computation to semantically ambiguous foreground regions.

Gated recursive token update. To preserve model expressivity, we adopt the "Middle-Cycle" strategy, in which the first and last Transformer blocks remain independent while a subset of intermediate blocks is recursively reused. Let $\mathbf{B}$ denote a Transformer block. The encoder is defined as

$$\text{Encoder} = \{\mathbf{B}_1, \underbrace{\mathbf{B}_2, \ldots, \mathbf{B}_{N-1}}_{N_r}, \mathbf{B}_N\}, \tag{2}$$

where N denotes the total number of Transformer blocks in the encoder, and N_r denotes the number of recursively shared intermediate blocks. R denotes the maximum recursion depth. Under this strategy, the effective depth of the resulting encoder is given by $N_L = N_r \times R + 2$, with the corresponding layer sequence expressed as Layers $= \{\mathbf{B}_1, (\mathbf{B}_2, \ldots, \mathbf{B}_{N-1}), (\mathbf{B}_2, \ldots, \mathbf{B}_{N-1}), \ldots, \mathbf{B}_N\}$. After the router selects the active tokens $A^r \in \mathbb{R}^{M' \times d}$ to proceed to the current recursion step, we apply self-attention to capture the interactions among active tokens. Specifically, the query, key, and value matrices are computed as $Q_h = A^r W_{Q_h}, K_h = A^r W_{K_h}, V_h = A^r W_{V_h}$, where $W_{Q_h}, W_{K_h}, W_{V_h} \in \mathbb{R}^{d \times d_h}$ are linear projection matrices. h denotes the $h-$th attention head; d_h is the dimension of each head, and H is the number of attention heads. The attention $A_h \in \mathbb{R}^{M' \times M'}$ is compute as $A_h = \text{Softmax}(Q_h \times K_h^\top)/\sqrt{d_h}$. Finally, the outputs of all attention heads are concatenated and projected by a linear layer to obtain the enhanced activate tokens:

$$\hat{A}^r = \text{Concat}(A_1 V_1, A_2 V_2, \ldots, A_H V_H) W_O \tag{3}$$

where $W_O \in \mathbb{R}^{d \times d}$ and $\hat{A}^r \in \mathbb{R}^{M' \times d}$. In remote sensing object detection, interactions among foreground tokens alone are insufficient, as they cannot fully exploit the contextual information surrounding the target objects or the complementary information across different scales. To this end, we further introduce deformable attention, allowing each active token to attend to the original multi-scale feature tokens and selectively capture complementary local context and scale-aware information. Let $Q = \hat{A}^r W_Q, K = T_L W_K, V = T_L W_V$ where $W_Q, W_K, W_V \in \mathbb{R}^{d \times d}$. It follows that $\hat{A}^r = \text{MSDeformAttn}(Q, K, V)$. After the recursive Transformer blocks and a feed-forward network (FFN), the selected tokens are updated using a gated residual connection. Specifically, the gate score is computed as $g_s = \sigma((s - \beta)/\tau)$ and the updated tokens are given by

$$A^r = g_s \times \hat{A}^r + (1 - g_s) \times A^r, \tag{4}$$

where $s \in \mathbb{R}^{M' \times 1}$ denotes the routing score produced by the MoR router, β is a learnable threshold, τ is a temperature coefficient, and $\sigma(\cdot)$ denotes the sigmoid function. For tokens with $s_k \geq \beta$, the next recursion state is $A_k^{r+1} = A_k^r$.

Remark. ARTR differs from conventional DETR-based encoders in two aspects. First, it adaptively allocates recursive computation according to token difficulty, allowing challenging foreground tokens to receive more targeted refinement rather than treating all tokens uniformly. Second, it incorporates classification uncertainty into the routing process, providing semantic information beyond feature-based importance and making the router more sensitive to ambiguous foreground tokens.

3.3 Vision-Language Aligned Supervision

Vision-language models (VLMs) are pretrained on large-scale and diverse data, endowing them with rich visual-semantic priors. Motivated by this, the Vision-Language Aligned Supervision module leverages such priors to enhance remote sensing detection in two stages. The first stage, **Multimodal input construction**, adapts the detector's multi-scale visual features into the VLM's representation space and concatenates them with the detection prompt and serialized ground-truth tokens to form the multimodal input. The second stage, **Auxiliary detection supervision**, extracts the hidden states at the designated localization positions from the VLM output and uses them for bounding-box regression under cross-entropy and box regression losses. We detail these two stages below.

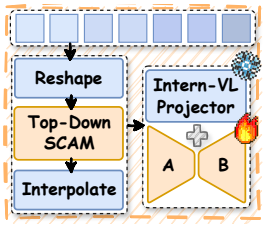

Figure 3: Detailed structure of the visual adapter.

Multimodal input construction. This stage builds scale-aware multimodal inputs for the VLM-assisted branch in three steps. **Multi-scale feature reconstruction and fusion.** The encoder outputs a flattened token set with mixed levels, losing explicit multi-scale information. We reconstruct it into multi-scale features $\{f_1, f_2, f_3, f_4\}$ by level and position. Since high-level features carry richer semantics while shallow ones preserve fine details, we design a top-down SCAM module that propagates high-level semantics to shallow layers. Specifically, each higher-level feature is upsampled and fused with its adjacent lower-level feature:

$$f_{l-1} = \text{SCAM}\left(\text{Concat}\left(f_{l-1}, \mathbf{UP}(f_l)\right)\right), \tag{5}$$

where $f_l \in \mathbb{R}^{H_l \times W_l \times d}$, H_l and W_l are spatial dimensions, $\mathbf{UP}(\cdot)$ denotes upsampling, and $\mathrm{SCAM}(\cdot)$ is the spatial context-aware operation (see Appendix for details). **Visual feature adaptation to the VLM space.** To reduce VLM overhead, we resize the four feature maps to 32×32 via bilinear interpolation, then project the channel dimension to the hidden dimension D of the VLM with a 1×1 convolution: $f_l' = \mathrm{Conv}_{1\times1}(\mathrm{Interpolation}(f_l))$, where $l \in \{1, 2, 3, 4\}$ and $f_l \in \mathbb{R}^{D\times32\times32}$. The maps are reshaped into token sequences and fed into the visual projector of a pretrained VLM. Since this projector is learned from natural images, applying it directly to remote sensing features causes domain mismatch. We therefore attach a LoRA adapter to the projector to adapt it to the detector's feature space while preserving pretrained visual-language alignment. Given a pretrained projection matrix W, its adapted form is $W' = W + \frac{\alpha}{r}BA$, where W is frozen, A and B are trainable low-rank matrices with rank $r \ll D$, and α is a scaling factor. The adapted projector maps the reshaped features into the VLM embedding space, yielding four levels of visual tokens $T_l^v = W' f_l'$. **Ground-truth serialization and scale assignment.** To provide explicit supervision, we serialize ground-truth annotations into a textual sequence, where each object is represented by its category and localization tokens, e.g., "<ref>car</ref><loc><loc><ref>ship</ref><loc>". The prompt and serialized ground truth are fed into the tokenizer and text embedding layer to obtain prompt tokens T_P and ground-truth tokens T_Y. Since the four visual levels correspond to different spatial scales, we assign ground-truth tokens to each level according to the area of their bounding boxes, so that each level is associated with objects of the appropriate scale. Finally, visual, prompt, and scale-specific ground-truth tokens are concatenated to form the multimodal input:

$$T_l^{In} = \mathrm{Concat}\left(T_P, T_l^v, T_l^Y\right), \tag{6}$$

where T_l^v and T_l^Y denote the visual-token and GT-token sequences of the l-th level, and $T_l^{In} \in \mathbb{R}^{N'\times D}$. The resulting four multimodal sequences provide scale-aware visual and textual supervision for the VLM-assisted branch.

Auxiliary detection supervision. During training, the serialized GT tokens are appended to the multimodal input to provide ground-truth context for next-token prediction and to define dedicated slots for box regression. For textual supervision, the language-model head predicts the next token at each target position, supervised by the corresponding GT token, yielding the cross-entropy loss L_{CE}. For localization supervision, we extract the hidden states at the <loc> positions from the VLM output $T_l^{Out} \in \mathbb{R}^{N'\times D}$, denoted as T_{loc}, which encode object-specific localization conditioned on the visual tokens and preceding semantic context. Each T_{loc} is then fed into a lightweight two-layer MLP to regress a continuous box in normalized (c_x, c_y, w, h) format: $\hat{b} = \sigma(\mathrm{MLP}(T_{loc}))$. The overall auxiliary detection loss is

$$L_{\mathrm{VLM}} = L_{\mathrm{CE}} + L_{\mathrm{box}}, \tag{7}$$

where L_{box} denotes the bounding-box regression loss.

Remark. The VLM-assisted branch is used only during training and is removed during inference, introducing no additional inference cost. During training, the target sequence is provided through teacher forcing to guide structured autoregressive prediction. Unlike approaches that autoregressively generate bounding-box coordinates as textual tokens, we directly decode continuous box coordinates from the hidden states at the <loc> positions, providing a more direct and compact localization mechanism.

4 Experiments

4.1 Experimental Setup

Dataset. The experiments conducted on three widely used remote sensing object detection benchmarks. DOTA-1.0(Ding et al., 2022) contains 15 object categories. We crop the original images into 1024×1024 subimages and discard those without object annotations, resulting in 13554 subimages. We use horizontal ground-truth bounding boxes, with 10276 images for training and 3278 images for testing. AI-TOD(Wang et al., 2021) is a large-scale optical remote sensing object detection dataset with 8 categories. It contains 14018 training images and 14018 testing images. VisDrone(Zhu et al., 2022) consists of 8599 drone-captured images covering 10 categories. We use 6471 images for training, 548 for validation, and 1610 for testing.

Table 1: Comparsion results of MoRE-DETR with other popular detection models on DOTA-1.0.

Model	mAP_{50}	AP_{50} per category														
		PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC
CNN Based																
YOLOv8[Online'23]	62.8	86.4	55.8	60.4	56.0	52.8	71.6	78.8	51.9	77.1	38.3	63.0	67.7	78.8	41.9	61.6
YOLOv10[NIPS'24]	60.1	85.6	38.7	54.8	56.5	46.2	70.2	74.5	53.7	77.1	36.2	63.4	64.3	82.3	44.9	53.0
YOLOv11[Online'24]	64.3	86.5	58.1	60.5	55.3	54.2	73.1	75.8	55.8	77.6	39.6	67.0	65.6	81.3	52.9	61.3
YOLOv12[arXiv'25]	63.2	86.5	56.7	60.4	55.4	53.4	72.9	75.2	52.4	77.9	39.6	64.4	64.7	83.0	44.1	60.8
YOLOv13[arXiv'25]	60.1	85.6	38.7	54.8	56.5	46.2	70.2	74.5	53.7	77.1	36.2	63.4	64.3	82.3	44.9	53.0
FBRT-YOLO[AAAI'25]	64.6	89.3	55.6	66.0	57.9	57.4	75.6	77.2	51.4	79.0	43.7	61.2	68.3	81.0	40.6	64.4
Hyper-YOLO[TPAMI'24]	65.0	89.6	61.2	64.6	57.2	56.7	73.1	76.8	52.3	78.5	38.6	66.1	70.1	83.2	42.8	64.2
FCOS[ICCV'19]	64.6	87.6	61.0	59.8	56.5	58.0	72.0	77.5	56.4	78.0	47.1	55.0	68.0	75.9	44.1	52.1
SPAR[AAAI'25]	65.6	91.3	68.9	41.2	44.9	68.2	87.3	89.2	93.9	58.2	73.3	52.3	47.6	80.6	58.0	29.4
Transformer Based																
Deformable DETR[arXiv'20]	63.3	80.2	62.8	64.1	55.2	62.1	69.8	71.2	49.5	73.5	45.5	60.1	63.5	84.5	46.7	59.8
DN-DETR[CVPR'22]	62.3	85.1	66.5	58.5	60.8	55.5	71.5	72.5	52.5	76.2	44.2	54.5	62.5	82.3	40.5	61.5
RT-DETRv1[CVPR'24]	69.4	88.8	65.0	71.1	59.1	63.4	72.5	80.5	54.2	80.0	45.1	68.9	69.0	92.1	67.5	63.7
LW-DETR[arXiv'24]	66.7	85.5	66.5	67.2	58.2	65.5	73.5	75.5	52.5	77.5	47.2	62.5	67.5	89.0	48.5	63.6
Focus-DETR[ICCV'23]	69.4	89.4	71.3	51.5	66.8	62.8	84.1	87.9	90.7	48.2	72.5	57.8	63.3	79.7	54.9	60.1
D-FINE[ICLR'25]	68.9	88.5	68.2	72.6	60.9	63.9	74.9	76.9	56.5	82.0	48.3	67.7	70.1	93.2	45.6	64.6
DEIM[CVPR'25]	69.6	89.6	66.5	74.4	61.3	65.7	75.4	75.8	56.3	81.6	47.8	67.2	70.8	93.4	53.4	65.6
DEIMv2[arXiv'25]	70.1	89.5	69.7	70.9	61.2	68.8	77.2	78.8	54.7	81.5	50.0	66.5	71.0	93.4	51.2	66.7
Salience DETR[CVPR'24]	64.3	86.1	63.2	61.4	58.6	59.9	67.7	70.5	53.4	74.8	41.7	59.6	56.4	90.2	53.3	67.7
SMWG-DETR[TGRS'26]	70.4	89.0	68.2	74.4	60.4	66.4	77.5	79.9	56.3	81.5	49.3	67.8	70.4	93.2	53.2	68.5
RT-DETRv4[ECCV'26]	71.4	89.5	69.0	74.2	60.7	66.5	77.5	79.8	56.0	81.2	49.2	66.0	70.5	93.0	68.5	69.5
LWGA-RT-DETR[AAAI'26]	69.9	88.9	68.0	73.5	59.9	65.7	76.8	79.3	55.6	81.0	48.8	65.5	69.9	92.8	52.5	70.3
RSRT-DETR[TGRS'26]	69.7	89.0	68.1	73.5	59.8	66.3	77.2	79.6	55.5	81.2	49.2	65.6	70.0	92.9	52.5	67.8
SKYDET-T[TGRS'26]	70.8	88.5	68.5	78.2	59.8	66.8	79.5	80.2	57.8	82.5	50.1	64.2	70.5	92.5	52.1	69.5
SKYDET-C[TGRS'26]	<u>72.6</u>	89.8	71.2	81.5	60.9	69.5	82.1	81.5	59.4	83.8	51.9	66.5	72.5	94.2	53.5	70.6
MoRE-DETR	**73.0**	92.0	75.6	57.7	69.7	67.4	86.0	87.8	92.2	61.0	76.4	62.0	64.3	82.4	57.9	63.0

Table 2: Comparison on the VisDrone dataset.

Method	VisDrone			
	mAP	mAP_{50}	AP_{vt}	AP_t
Transformer Based				
Focus-DETR[ICCV'23]	23.5	41.9	3.7	10.6
Salience DETR[CVPR'24]	28.4	48.7	5.4	13.7
Florence-2-Base[CVPR'24]	20.3	33.5	-	-
Florence-2-Large[CVPR'24]	21.8	35.2	-	-
Grounding DINO[ECCV'24]	15.2	28.3	4.6	8.6
LLMDet[CVPR'25]	20.4	34.1	7.4	12.8
RT-DETRv4[ECCV'26]	26.5	43.5	5.1	12.9
SMWG-DETR[TGRS'26]	25.0	43.7	3.6	10.7
CNN Based:				
RetinaNet[ICCV'17]	7.9	15.9	0.1	0.7
Faster RCNN[TPAMI'17]	22.3	38.0	0.1	6.2
Cascade RCNN[CVPR'18]	22.5	38.5	0.5	6.8
FCOS[ICCV'19]	14.1	25.5	0.1	2.1
DetectoRS[CVPR'21]	25.7	41.7	0.5	7.6
RFLA[ECCV'22]	27.4	45.3	4.5	12.9
YOLO-World[CVPR'24]	14.3	23.6	2.8	7.1
KLDet[TGRS'24]	28.5	50.7	-	-
SR-TOD[ECCV'24]	27.3	46.9	2.3	11.5
RetinaNet/VLM4RSDet[CVPR'26]	23.5	41.0	2.7	6.8
FCOS/VLM4RSDet[CVPR'26]	22.9	40.7	3.6	9.2
Faster R-CNN/VLM4RSDet[CVPR'26]	26.7	47.9	8.3	14.8
DetectoRS/VLM4RSDet[CVPR'26]	31.4	53.0	<u>9.3</u>	**16.5**
MoRE-DETR	**31.6**	**53.7**	**9.4**	<u>16.3</u>

Table 3: Comparison on the AI-TOD dataset.

Method	AITOD			Params	FLOPs
	mAP_{50}	mAP_{75}	mAP		
CNN Based					
YOLOv8[Online'23]	45.7	18.0	22.1	68.16M	258.2G
YOLOv10[NIPS'24]	45.2	17.3	21.2	31.67M	171.1G
YOLOv11[Online'24]	47.3	17.6	22.0	56.88M	193.5G
YOLOv12[arXiv'25]	42.3	14.3	18.8	59.13M	199.9G
YOLOv13[arXiv'25]	45.2	17.6	21.9	63.97M	200.1G
FBRT-YOLO[AAAI'25]	42.1	15.8	19.8	23.26M	187.1G
Hyper-YOLO[TPAMI'24]	46.1	18.3	22.4	68.16M	258.2G
FCOS[ICCV'19]	47.8	18.5	22.8	50.95M	234.3G
Transformer Based					
Deformable DETR[arXiv'20]	38.5	10.5	12.4	41.06M	175.4G
Focus-DETR[ICCV'23]	53.2	15.0	22.1	49.2M	113.8G
RT-DETRv1[CVPR'24]	52.1	14.8	22.0	76.39M	200.4G
LW-DETR[arXiv'24]	37.3	9.8	15.2	118.0M	174.2G
Salience DETR[CVPR'24]	45.7	12.5	19.1	62.52M	205.8G
D-FINE[ICLR'25]	51.3	16.9	22.4	61.55M	202.2G
DEIM[CVPR'25]	49.0	16.4	21.8	61.63M	202.3G
SMWG-DETR[TGRS'26]	55.2	21.8	26.5	65.9M	194.1G
RT-DETRv4[ECCV'26]	55.3	21.0	25.8	62.1M	202.3G
LWGA-RT-DETR[AAAI'26]	53.6	18.8	23.8	63.8M	184.6G
RSRT-DETR[TGRS'26]	54.2	20.1	24.8	66.8M	203.6G
SKYDET-T[TGRS'26]	54.7	**23.5**	<u>27.8</u>	805.5M	458.2G
SKYDET-C[TGRS'26]	<u>56.1</u>	21.2	26.4	20.58M	237.8G
MoRE-DETR	**56.6**	<u>22.4</u>	**28.0**	46.24M	117.8G

Implementation details. We adopt Focus-DETR as the baseline model and follow its default hyperparameter settings. The AdamW optimizer is used with an initial learning rate of 1×10^{-4}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a weight decay of 1×10^{-4}. For training, we follow the standard $3\times$ training schedule for DOTA-1.0 and AI-TOD, and $1\times$ training schedule for VisDrone, with a batch size of 2 for all datasets. For the VLAS module, we employ the InternVL3.5-1B model(Wang et al., 2025), whose language model is Qwen3-0.6B. For evaluation, we employ multiple metrics to assess detection performance, including mAP_{50}, mAP_{75}, and $mAP_{50:95}$. On the DOTA-1.0 dataset, we additionally report the AP_{50} of each category. We also report mAP_{vt} and mAP_t, which corre-

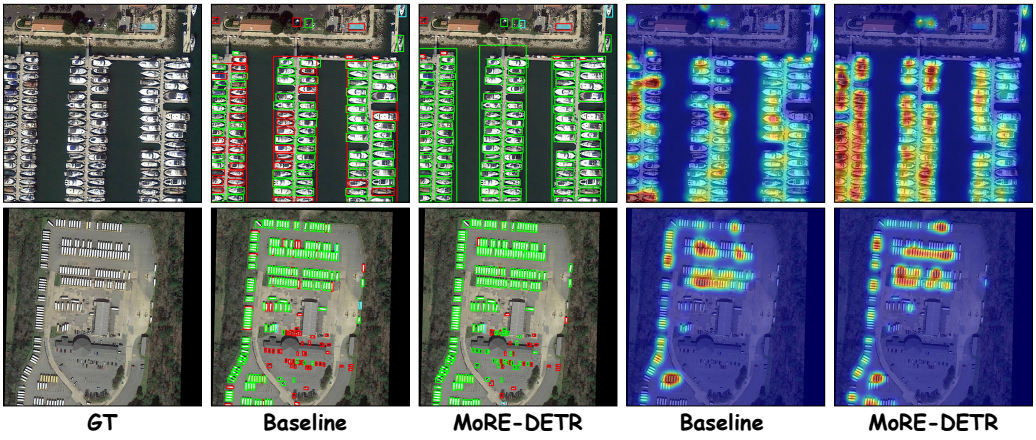

Figure 4: Visualization comparison on DOTA-1.0 dataset. Green, blue, and red rectangles indicate True Positives (TPs), False Positives (FPs), and False Negatives (FNs), respectively, with an IoU threshold of 0.5 for determining correct detections.

spond to objects with areas smaller than 8^2 pixels and between 8^2 and 16^2 pixels, respectively. All experiments are conducted on two NVIDIA RTX 3090 GPUs.

4.2 Comparison with State-of-the-arts

Compared methods. We compare with representative CNN-based detectors (e.g., YOLO series, RetinaNet, Faster R-CNN, RFLA, VLM4RSDet) and Transformer-based detectors (e.g., Deformable DETR, DN-DETR, RT-DETR, Focus-DETR, D-FINE, Florence-2, Grounding DINO). *See Appendix for details.*

Quantitative comparison. Tables 1-3 report the quantitative results of MoRE-DETR and other competitive methods on three benchmark datasets. The main observations are summarized as follows: (1) **Strong overall performance across all datasets.** MoRE-DETR achieves consistently strong results on all three benchmarks. On DOTA-1.0, it outperforms the competitive SKYDET by 0.4% in mAP_{50}, and on VisDrone it surpasses VLM4RSDet by 0.2% in mAP. This gains stem from the complementary designs of our framework: the uncertainty-aware MoR router concentrates recursive computation on semantically ambiguous foreground tokens, while the VLAS branch supplies explicit geometric supervision from VLM hidden states, jointly improving hard-object refinement. (2) **Favorable accuracy-efficiency trade-off.** The early-exit behavior of ARTR prevents redundant computation on easy tokens, and the VLM branch is removed at inference, so no additional VLM overhead is incurred. Consequently, although MoRE-DETR is neither the smallest nor the fastest, it provides a favorable balance between accuracy and cost. Compared with the real-time detector RT-DETRv4, it improves mAP_{50} by 1.4% on AI-TOD at comparable complexity. (3) **Competitive performance on tiny objects.** MoRE-DETR achieves the same mAP_{vt} as VLM4RSDet on AI-TOD, with mAP_t only 0.2% lower. This is mainly attributed to the uncertainty-aware routing, which keeps ambiguous tiny-object tokens in recursion for additional refinement, and to the scale-assigned VLAS supervision that aligns visual tokens with geometric targets across scales.

Visualization comparison. As shown in Fig. 4, we qualitatively compare MoRE-DETR with the baseline (Focus-DETR) on DOTA-1.0, a challenging benchmark with large scale variations and complex spatial distributions. For consistency, horizontal bounding-box annotations and the standard protocol are used. The first column shows ground truth, and the second and third columns show baseline and MoRE-DETR detections at IoU 0.5. MoRE-DETR produces fewer false positives and false negatives, indicating more accurate localization. The last two columns show attention heatmaps, where MoRE-DETR focuses more on object regions with fewer background activations. These results demonstrate its effectiveness in complex remote sensing scenes.

4.3 Ablation Study and Discussions

Effectiveness of each proposed module. Table 4 reports the ablation results. Adding ARTR alone improves mAP from 23.5% to 24.2%, showing that dynamically allocating recursive computation

Table 4: Ablation study of each proposed module on VisDrone dataset. Unc: Uncertainty.

Base	ARTR	VLAS	mAP	mAP_{75}	mAP_{50}
✓			23.5	22.3	41.9
✓		✓	25.3	24.2	44.8
✓	MLP+Unc.		24.2	23.6	42.7
✓	MLP	✓	28.8	28.5	50.1
✓	MLP+Unc.	✓	29.2	28.8	50.5

Table 5: Ablation study of VLM supervision strategies. Unfrz.last 2 FFN: unfreezing the last two FFN layers.

Strategy	$\mathcal{L}_{CE}$	$\mathcal{L}_{box}$	mAP	mAP_{75}	mAP_{50}
Frozen	✓		28.7	28.6	49.8
Frozen	✓	✓	29.2	28.8	50.5
Unfrz. last 2 FFN	✓	✓	29.6	29.4	51.2

Table 6: Ablation study of different routing strategies.

Routing strategy	R	N_r	mAP	mAP_{75}	mAP_{50}	Params	FLOPs
Token-choice	3	1	28.4	28.0	49.4	44.85M	208.1G
Token-choice	2	2	28.9	28.6	50.1	46.84M	219.8G
Expert-choice	3	1	29.1	28.6	50.8	44.69M	218.8G
Expert-choice	2	2	29.2	28.8	50.5	46.24M	226.3G
Expert-choice	3	2	29.1	28.7	50.2	46.24M	237.9G

to foreground tokens outperforms the static cascade token reduction in the baseline. Adding VLAS alone yields a 1.8% gain, confirming that VLM-based supervision effectively guides the detector. Combining both achieves the best result, improving mAP by 5.7%. Within ARTR, uncertainty score refinement brings an additional 0.4% gain. These results validate the effectiveness of both modules.

Analysis of VLM supervision. Table 5 compares different VLM supervision strategies on VisDrone, where "Strategy" denotes how the VLM is adapted, $\mathcal{L}_{CE}$ is the text prediction loss, and $\mathcal{L}_{box}$ is the box regression loss. Using only $\mathcal{L}_{CE}$ yields 28.7% mAP. Adding $\mathcal{L}_{box}$ improves mAP by 0.5%, confirming the benefit of explicit geometric supervision. Further unfreezing the FFN layers in the last two Transformer blocks, while keeping the rest frozen, allows the localization hidden states to better adapt to remote sensing data and the box objective, bringing an additional 0.9% gain. These results show that decoding continuous boxes from dedicated hidden states effectively exploits pretrained VLM representations for remote sensing detection.

Analysis of routing strategies. Table 6 compares Expert-choice and Token-choice routing under various recursion settings, with the regularization loss for Token-choice detailed in the Appendix. Under identical recursion depth and shared blocks, Expert-choice consistently outperforms Token-choice, matching prior findings (Bae et al., 2026; He et al., 2026). We attribute this to Token-choice's sensitivity to the routing threshold and temperature, which induces unstable routing decisions and sparse gradients, and to its regularization objective being less aligned with the detection objective. Among Expert-choice settings, $R = 2$ and $N_r = 2$ achieves the best mAP at moderate cost, and we adopt it as the default. Further increasing the recursion depth or the number of shared blocks, e.g., $R = 3$ and $N_r = 2$, brings no additional gain but raises FLOPs from 226.3G to 237.9G, confirming that the chosen setting offers the best accuracy-efficiency trade-off.

5 Conclusion

We proposed MoRE-DETR, a novel adaptive recursive reasoning framework for remote sensing object detection, which consists of two main modules. The Adaptive Recursive Token Routing module employs an uncertainty-aware router to dynamically allocate recursive computation according to token difficulty, enabling challenging foreground tokens to receive more targeted and effective feature refinement. In addition, the Visual-Language Aligned Supervision module leverages a pretrained VLM to provide complementary detection supervision. By extracting localization representations from VLM hidden states and directly regressing continuous bounding-box coordinates, it provides more explicit geometric supervision for object localization. Extensive experiments on multiple remote sensing object detection benchmarks demonstrate that MoRE-DETR consistently improves detection performance and achieves state-of-the-art results on several benchmarks. A promising direction for future work is to improve cross-domain generalization, so that MoRE-DETR can transfer more reliably across diverse remote sensing scenes while preserving its inference efficiency.

AI USE STATEMENT

In the preparation of this manuscript, we used generative AI tools, specifically ChatGPT-5.6 Luna, to assist with language polishing, including the correction of grammatical errors and improvement of clarity and readability. Generative AI was not used to generate or modify figures or images, or to alter the technical content or methodology of this work. All research ideas, methodological decisions, interpretations, and experiments were developed and conducted independently by the authors. All AI-assisted text was carefully reviewed by the authors, and we take full responsibility for the final content of this work, including all text, figures, and experimental results.

REFERENCES

Sangmin Bae, Yujin Kim, Reza Bayat, Sungnyun Kim, Jiyoun Ha, Tal Schuster, Adam Fisch, Hrayr Harutyunyan, Ziwei Ji, Aaron Courville, et al. Mixture-of-recursions: Learning dynamic recursive depths for adaptive token-level computation. *Advances in Neural Information Processing Systems*, 38:96572–96617, 2026.

Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 6154–6162. Ieee, 2018.

Bing Cao, Haiyu Yao, Pengfei Zhu, and Qinghua Hu. Visible and clear: Finding tiny objects in difference map. In *European Conference on Computer Vision*, pp. 1–18. Springer, 2024.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

Mingshu Chen, Wei Zhao, Nannan Li, Dongjin Li, Rufei Zhang, and Jingyu Xu. Smwg-detr: Detr enhanced by fourier spectral modulation and wavelet-guided fusion for tiny object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 2026.

Qiang Chen, Xiangbo Su, Xinyu Zhang, Jian Wang, Jiahui Chen, Yunpeng Shen, Chuchu Han, Ziliang Chen, Weixiang Xu, Fanrong Li, et al. Lw-detr: A transformer replacement to yolo for real-time detection. *arXiv preprint arXiv:2406.03459*, 2024.

Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16901–16911. IEEE, 2024.

Chenlong Ding, Chengyi Shi, Daofang Liu, Zhihao Shi, Xin Lyu, Zhenyu Fang, Xue Liu, Lingqiang Meng, Yiwei Fang, Chengming Zhang, et al. Efficient adaptation of vision foundation model for high-resolution remote sensing image segmentation via spatial-frequency modeling and sparse refinement. *Remote Sensing*, 18(9):1295, 2026.

Jian Ding, Nan Xue, Gui-Song Xia, Xiang Bai, Wen Yang, Michael Yang, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Object detection in aerial images: A large-scale benchmark and challenges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7778–7796, 2022. doi: 10.1109/TPAMI.2021.3117983.

Yifan Feng, Jiangang Huang, Shaoyi Du, Shihui Ying, Jun-Hai Yong, Yipeng Li, Guiguang Ding, Rongrong Ji, and Yue Gao. Hyper-yolo: When visual object detection meets hypergraph computation. *IEEE transactions on pattern analysis and machine intelligence*, 47(4):2388–2401, 2024.

Shenghao Fu, Qize Yang, Qijie Mo, Junkai Yan, Xihan Wei, Jingke Meng, Xiaohua Xie, and Wei-Shi Zheng. Llmdet: Learning strong open-vocabulary object detectors under the supervision of large language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14987–14997. IEEE, 2025.

Yunhong He, Zhengqing Yuan, Weixiang Sun, Yiyang Li, Yixin Liu, Yanfang Ye, and Lichao Sun. Vision-mor: Scaling vision transformer via patch-level mixture-of-recursions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 4699–4707, 2026.

Xiuquan Hou, Meiqin Liu, Senlin Zhang, Ping Wei, and Badong Chen. Salience detr: Enhancing detection transformer with hierarchical salience filtering refinement. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17574–17583. IEEE, 2024.

Zhangchi Hu, Peixi Wu, Jie Chen, Huyue Zhu, Yijun Wang, Yansong Peng, Hebei Li, and Xiaoyan Sun. Dome-detr: Detr with density-oriented feature-query manipulation for efficient tiny object detection. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 101–110, 2025.

Shihua Huang, Yongjie Hou, Longfei Liu, Xuanlong Yu, and Xi Shen. Real-time object detection meets dinov3. *arXiv preprint arXiv:2509.20787*, 2025a.

Shihua Huang, Zhichao Lu, Xiaodong Cun, Yongjun Yu, Xiao Zhou, and Xi Shen. Deim: Detr with improved matching for fast convergence. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15162–15171. IEEE, 2025b.

Yi-Xin Huang, Hou-I Liu, Hong-Han Shuai, and Wen-Huang Cheng. Dq-detr: Detr with dynamic query for tiny object detection. In *European Conference on Computer Vision*, pp. 290–305. Springer, 2025c.

Ziyue Huang, Yongchao Feng, Ziqi Liu, Shuai Yang, Qingjie Liu, and Yunhong Wang. Openrsd: Towards open-prompts for object detection in remote sensing images. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8384–8394. IEEE, 2025d.

Dingding Jia, Jiankang Wang, Longlong Zhang, Zhiheng Liu, and Xuan Wang. Sfs-detr: Spatial-frequency selection for uav object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6582–6591, 2026.

Han Jiang, Wenfei Yang, Tianzhu Zhang, and Yongdong Zhang. Scoa: Revisiting domain generalized object detection with style-conditioned adaptation. In *Forty-third International Conference on Machine Learning*.

Glenn Jocher and Jing Qiu. Ultralytics yolo11, 2024. URL `https://github.com/ultralytics/ultralytics`.

Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics yolov8, 2023. URL `https://github.com/ultralytics/ultralytics`.

Mengqi Lei, Siqi Li, Yihong Wu, Han Hu, You Zhou, Xinhu Zheng, Guiguang Ding, Shaoyi Du, Zongze Wu, and Yue Gao. Yolov13: Real-time object detection with hypergraph-enhanced adaptive visual perception. *arXiv preprint arXiv:2506.17733*, 2025.

Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13619–13627, 2022.

Jingfang Li, Haoran Zhu, Wen Yang, Jinrui Zhang, Fang Xu, Haijian Zhang, and Gui-Song Xia. Uhr-detr: Efficient end-to-end small object detection for ultra-high-resolution remote sensing imagery. *arXiv preprint arXiv:2604.21435*, 2026a.

Nianxin Li, Mao Ye, Lihua Zhou, Song Tang, Yan Gan, Zizhuo Liang, and Xiatian Zhu. Self-prompting analogical reasoning for uav object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 18412–18420, 2025a.

Nianxin Li, Mao Ye, Lihua Zhou, Shuaifeng Li, Song Tang, Luping Ji, and Ce Zhu. Multimodal causal reasoning for uav object detection. *Advances in Neural Information Processing Systems*, 38:168069–168094, 2026b.

Qingyun Li, Xin He, Xinya Shu, Yi Yu, Dong Chen, Yushi Chen, and Xue Yang. A simple aerial detection baseline of multimodal language models. In *IGARSS 2025-2025 IEEE International Geoscience and Remote Sensing Symposium*, pp. 6833–6837. IEEE, 2025b.

Yan Li, Weiwei Guo, Xue Yang, Ning Liao, Dunyun He, Jiaqi Zhou, and Wenxian Yu. Toward open vocabulary aerial object detection with clip-activated student-teacher learning. In *European Conference on Computer Vision*, pp. 431–448. Springer, 2024.

Yuxuan Li, Yuming Chen, Yunheng Li, Ming-Ming Cheng, Xiang Li, and Jian Yang. Unifying heterogeneous multi-modal remote sensing detection via language-pivoted pretraining. *arXiv preprint arXiv:2603.01758*, 2026c.

Yuxuan Li, Xiang Li, Yunheng Li, Yicheng Zhang, Yimian Dai, Qibin Hou, Ming-Ming Cheng, and Jian Yang. Sm3det: A unified model for multi-modal remote sensing object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 6717–6725, 2026d.

Zijun Liao, Yian Zhao, Xin Shan, Yu Yan, Chang Liu, Lei Lu, Xiangyang Ji, and Jie Chen. Rt-detrv4: Painlessly furthering real-time object detection with vision foundation models. *arXiv preprint arXiv:2510.25257*, 2025.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

Xu Lin, Jinlong Peng, Zhenye Gan, Jiawen Zhu, and Jun Liu. Yolo-master: Moe-accelerated with specialized transformers for enhanced real-time detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18440–18449, 2026.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024.

Wei Lu, Xue Yang, and Si-Bao Chen. Lwganet: Addressing spatial and channel redundancy in remote sensing visual tasks with light-weight grouped attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 7574–7582, 2026.

Yehao Lu, Minghe Weng, Zekang Xiao, Rui Jiang, Wei Su, Guangcong Zheng, Ping Lu, and Xi Li. Dynamic-dino: Fine-grained mixture of experts tuning for real-time open-vocabulary object detection. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20847–20856. IEEE, 2025.

Jiancheng Pan, Yanxing Liu, Yuqian Fu, Muyuan Ma, Jiahao Li, Danda Pani Paudel, Luc Van Gool, and Xiaomeng Huang. Locate anything on earth: Advancing open-vocabulary object detection for remote sensing community. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 6281–6289, 2025.

Yansong Peng, Hebei Li, Peixi Wu, Yueyi Zhang, Xiaoyan Sun, and Feng Wu. D-fine: Redefine regression task of detrs as fine-grained distribution refinement. In *International Conference on Learning Representations*, volume 2025, pp. 44015–44031, 2025.

Siyuan Qiao, Liang-Chieh Chen, and Alan Yuille. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10208–10219. IEEE, 2021.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

Byungseok Roh, JaeWoong Shin, Wuhyun Shin, and Saehoon Kim. Sparse detr: Efficient end-to-end object detection with learnable sparsity. *arXiv preprint arXiv:2111.14330*, 2021.

Shuohao Shi, Qiang Fang, and Xin Xu. Vlm4rsdet: Collaborative optimization with vision-language model for enhancing remote sensing object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18450–18460, 2026.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Yunjie Tian, Qixiang Ye, and David Doermann. Yolov12: Attention-centric real-time object detectors. In *Advances in Neural Information Processing Systems*, volume 38, pp. 78433–78457, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/7103444259031cc58051f8c9a4868533-Paper-Conference.pdf.

Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Yolov10: Real-time end-to-end object detection. *Advances in neural information processing systems*, 37: 107984–108011, 2024.

Jinwang Wang, Wen Yang, Haowen Guo, Ruixiang Zhang, and Gui-Song Xia. Tiny object detection in aerial images. In *2020 25th international conference on pattern recognition (ICPR)*, pp. 3791–3798. IEEE, 2021.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Guoting Wei, Xia Yuan, Yu Liu, Zhenhao Shang, Kelu Yao, Chao Li, Qingsen Yan, Chunxia Zhao, Haokui Zhang, and Rong Xiao. Ova-detr: Open vocabulary aerial object detection using image-text alignment and fusion. *arXiv preprint arXiv:2408.12246*, 2(3), 2024.

Ke Wu, Yanan Zhang, Yingjie Gao, Wenhao Li, Chenyu Zhou, XinZhu Ma, Jiaxin Chen, and Di Huang. Dronefine: Domain-aware parameter-efficient fine-tuning of vision-language detectors for drone images. *arXiv preprint arXiv:2607.00338*, 2026.

Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4818–4829. IEEE, 2024.

Yao Xiao, Tingfa Xu, Yu Xin, and Jianan Li. Fbrt-yolo: Faster and better for real-time aerial image detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 8673–8681, 2025.

Chang Xu, Jinwang Wang, Wen Yang, Huai Yu, Lei Yu, and Gui-Song Xia. Rfla: Gaussian receptive field based label assignment for tiny object detection. In *European conference on computer vision*, pp. 526–543. Springer, 2022.

Huaxiang Zhang, Hao Zhang, Kai Liu, Zhongxue Gan, and Guo-Niu Zhu. Uav-detr: efficient end-to-end object detection for unmanned aerial vehicle imagery. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 15143–15149. IEEE, 2025.

Jinlong Zhang, Sanping Zhou, Ye Deng, and Le Wang. Adaptive receptive field network for small objects detection in remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 2026a.

Yao Zhang, Wei Guo, Boxiang Xie, Lingfeng Lin, Jie Zhang, Hongwei Yang, Yuke Meng, Yi Liu, and Wei Zhang. Skydet: An end-to-end multi-scale attentive detection network from foundation models for small objects in remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 2026b.

Yin Zhang, Mu Ye, Guiyi Zhu, Yong Liu, Pengyu Guo, and Junhua Yan. Ffca-yolo for small object detection in remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 62: 1–15, 2024.

Hongyang Zhao, Kai Chen, Yao Zhang, Xingdong Li, Honggang Li, and Jing Jin. Rsrt-detr: Hierarchical polar attention and multiscale hypergraph networks for dense remote sensing small-object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 64:1–21, 2026.

Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. Detrs beat yolos on real-time object detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16965–16974. IEEE, 2024.

Dehua Zheng, Wenhui Dong, Hailin Hu, Xinghao Chen, and Yunhe Wang. Less is more: Focus attention for efficient detr. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6651–6660. IEEE, 2023.

Yuanjie Zhi, Jianglin Zhao, Chao Song, Mingyang Ma, and Shaohui Mei. Saff-detr: An end-to-end object detection network for remote sensing images with targets of varying sizes based on scale adaptation and frequency fusion. *IEEE Transactions on Geoscience and Remote Sensing*, 2025.

Zhuangzhuang Zhou and Yingying Zhu. Kldet: Detecting tiny objects in remote sensing images via kullback–leibler divergence. *IEEE transactions on geoscience and remote sensing*, 62:1–16, 2024.

Pengfei Zhu, Longyin Wen, Dawei Du, Xiao Bian, Heng Fan, Qinghua Hu, and Haibin Ling. Detection and tracking meet drones challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7380–7399, 2022. doi: 10.1109/TPAMI.2021.3119563.

Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.

A Appendix

A.1 More Related Works

Dynamic Routing in Object Detection. Dynamic routing aims to adaptively allocate computation according to the input, and existing methods can be broadly divided into two categories. The first category (Lu et al. (2025); Zhang et al. (2026a); Lin et al. (2026); Li et al. (2026d); Jiang et al.;) is mainly based on Mixture of Experts (MoE), which expands the network horizontally by providing diverse computation paths. It typically contains multiple parallel expert branches, with a router dynamically activating a Top-K subset of experts for each input. The second category is based on Mixture of Recursions (MoR) Bae et al. (2026), which extends computation along the depth dimension. It employs a router to dynamically determine the recursion depth of each token. Vision-MoR He et al. (2026) further extends this idea to vision tasks through adaptive recursive computation at the image-patch level. We first introduce the MoR mechanism into remote sensing object detection and develop an uncertainty-aware router to better process challenging foreground tokens.

A.2 Implementation Details

A.2.1 Preliminary knowledge on MoR

MoR Bae et al. (2026) introduces two distinct routing strategies for dynamically allocating recursive computation. Each MoR block composed of several transformer blocks is repeatedly applied across multiple recursive steps.

Expert-choice routing. It defines each recursion step as an expert. For example, when $R = 3$, there are three experts, and each expert dynamically selects the active tokens at its corresponding recursion step. Specifically, at each recursion step, the router takes the current hidden states H^r as input and produces a routing score $g^r = \text{Router}(H^r)$. Then, tokens with scores larger than the threshold β^r are selected to pass through the shared recursion block:

$$H_i^{r+1} = \begin{cases} g_i^r \times f(H_i^r) + H_i^r, & \text{if } g_i^r \geq \beta^r \\ H_i^r, & \text{otherwise} \end{cases} \tag{8}$$

where β^r denotes the routing threshold at the r-th recursion step, and $f(\cdot)$ denotes the computation of the shared recursion block.

Token-choice routing. Different from expert-choice routing, token-choice routing determines the recursion depth of each token at the beginning. Specifically, the router takes the initial hidden

states H^0 as input and predicts a set of recursion scores $g = \text{Router}(H^0)$. The recursion depth of each token is then determined by $d_i = argmax(g_i^r)$, where $d_i^r \in \{0, 1, 2, 3, \ldots, R\}$ denotes the predetermined recursion depth of token i. The hidden state is then recursively updated according to the assigned depth:

$$H_i^{r+1} = \begin{cases} g_i^r \times f(H_i^r) + H_i^0, & \text{if } d_i \geq r, \\ H_i^r, & \text{otherwise.} \end{cases} \tag{9}$$

A.2.2 Detection Head and Loss Function

Following the DETR framework, our model is optimized in an end-to-end manner. The overall training objective is formulated as:

$$\mathcal{L}_{total} = \mathcal{L}_{Focus} + \mathcal{L}_{vlm} \tag{10}$$

where $\mathcal{L}_{Focus}$ denotes the detection loss of the baseline model, and $\mathcal{L}_{vlm}$ represents the supervision loss of the VLAS module. $\mathcal{L}_{Focus}$ is defined as:

$$\mathcal{L}_{Focus} = \lambda_m \mathcal{L}_{match} + \lambda_d \mathcal{L}_{dn} + \lambda_f \mathcal{L}_f + \lambda_e \mathcal{L}_{enc} + \mathcal{L}_{vlm} \tag{11}$$

where $\mathcal{L}_{match}$ denotes the Hungarian matching loss, $\mathcal{L}_{dn}$ is the denoising loss, $\mathcal{L}_f$ represents the foreground token selection loss, and $\mathcal{L}_{enc}$ is the auxiliary encoder loss. In our method, $\mathcal{L}_{enc}$ is used to supervise the uncertainty classification head. The coefficients λ_m, λ_d, λ_f, and λ_e are used to balance the corresponding loss terms.

Token-choice regularization. To encourage confident depth assignments, we introduce a depth entropy regularization:

$$\mathcal{L}_{ent} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{d=0}^{R} p_i^r \log p_i^r, \tag{12}$$

where $p_i^r = \text{Softmax}(g_i^r)$ denotes the probability that token i is assigned recursion depth r. Minimizing this entropy encourages each token to produce a sharp depth distribution, resulting in more decisive and stable routing decisions.

A.2.3 Top-down SCAM in Visual Adapter.

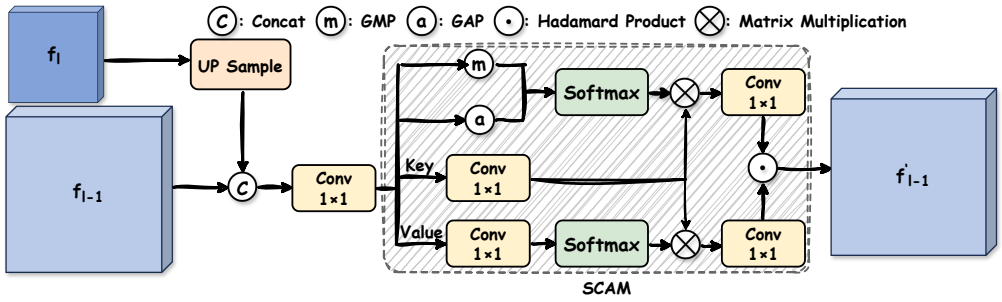

Figure 5: Detailed architecture of the top-down SCAM.

As mentioned above, high-level features contain richer semantic information, whereas shallow features preserve finer spatial details. As shown in Fig. 5, motivated by the top-down score modulation in the Foreground Token Selector Zheng et al. (2023) and the spatial context-aware mechanism Zhang et al. (2024), we introduce a top-down SCAM module to enhance the interaction between different feature levels. Since the encoder mainly performs self-attention among selected active tokens and deformable attention with a limited number of sampled positions, the resulting features may lack sufficient cross-level and global contextual information. This limitation may affect the visual understanding of the VLM, especially in scenes with densely distributed small objects. Specifically, through lightweight convolutional operations, SCAM computes attention weights to emphasize informative features while suppressing irrelevant background information. Based on these reweighted features, the top-down SCAM propagates high-level semantic information to shallow features, thereby producing more informative and context-enhanced visual representations.

A.2.4 ALGORITHMIC DETAILS OF ARTR AND VLAS.

Algorithm 1 Adaptive Recursive Token Reasoning (Middle-Cycle)

Input: The original flattened tokens T_o.
Output: The updated original tokens $T_o^{'}$.

Foreground tokens and their indices $A^0, idx^0 \leftarrow \mathrm{FTS}(T_o)$.
$A^{0'} \leftarrow \mathrm{FirstBlack}(A^0)$.
for $r = 1, \ldots, R$ **do**
  $idx^r \leftarrow s_i^{r-1} \geq \beta^r$
  $A^r \leftarrow A^{r-1}[idx^r]$
  $s^r \leftarrow \mathbf{MLP}(A^r)$
  $u^r \leftarrow \mathrm{Uncertainty}(\mathbf{Cls}(A^r))$
  $\hat{s}^r \leftarrow s^r + \lambda u^r$.
  $A^{r'} \leftarrow \mathrm{SharedBlocks}(A^r)$
  $A^r \leftarrow \hat{s}^r \times A^{r'} + (1 - \hat{s}^r) \times A^r$
end for
$A^{0'} \leftarrow \mathrm{EndBlock}(A^{0'})$
update $A^{0'}$ in T_0 according to idx^0
return $T_o^{'}$

Algorithm 2 Vision-Language ALigned Supervision

Input: The encoder output tokens T_o, Prompt token T_P, and serialized ground-truth tokens T_Y.
Output: The predicited text T_e, The predicited bounding boxes b.

Multi-scale features $\{f_1, f_2, f_3, f_4\} \leftarrow \mathrm{Reshape}(T_0)$.
for $l = 4, 3, 2$ **do**
  $f_l^{'} \leftarrow \mathbf{Up}(f_l)$
  $f_{l-1}^{'} \leftarrow \mathrm{SCAM}(\mathbf{Concat})(f_{l-1}, f_l)$
end for
for $l = 1, \ldots, 4$ **do**
  $T_v^l \leftarrow \mathrm{Lora}(\mathrm{Interpolate}(f_l))$
  $T_{in}^l \leftarrow \mathbf{Concat}(T_P^l, T_v^l, T_Y^l)$
  Hidden states $H^l \leftarrow \mathrm{Qwen3}(T_{in}^l)$
  Predicted Text $T_e^l \leftarrow \mathrm{LMHead}(H^l)$
  Predicted bounding box $b_i^l \leftarrow \mathbf{MLP}(H^l[loc_i])$
end for
return $\{T_e^1, T_e^2, T_e^3, T_e^4\}$, $\{b^1, b^2, b^3, b^4\}$,

The overall procedures of the proposed modules are summarized in the two algorithms above. Algorithm 1 first filters background tokens through the Foreground Token Selector and then iteratively routes the selected foreground tokens according to their feature-based importance and classification uncertainty. The active tokens are recursively refined by the shared Transformer blocks with gated updates, while inactive tokens exit early, after which the refined tokens are written back to their original positions. The Transformer blocks used above follow the same architecture and operations as the ViT blocks in the baseline. Algorithm 2 first reconstructs the encoder output into multi-scale features and enhances them with the top-down SCAM. The visual features are then projected into the Qwen3 semantic space and combined with the prompt and serialized ground-truth tokens. The resulting hidden states are used for text prediction and continuous bounding-box regression, providing semantic and geometric supervision during training.

A.3 MORE EXPERIMENTAL RESULTS

Compared methods. We compare our method with two main categories of object detection approaches. The first category consists of CNN-based methods, including YOLOv8Jocher et al. (2023), YOLOv10Wang et al. (2024), YOLOv11Jocher & Qiu (2024), YOLOv12Tian et al. (2025), YOLOv13Lei et al. (2025), FBRT-YOLOXiao et al. (2025), Hyper-YOLOFeng et al. (2024), Reti-

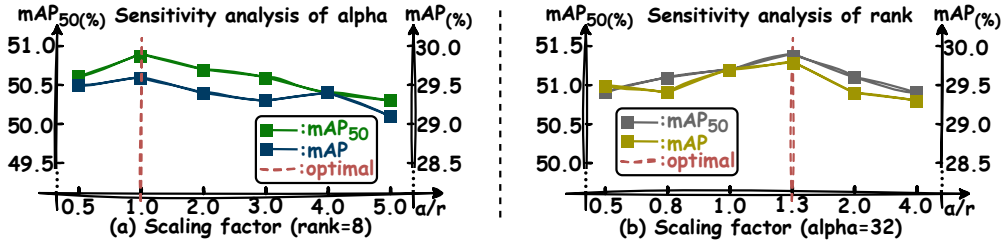

Figure 6: (a) Hyperparameter sensitivity analysis of α with $r = 8$. (b) Hyperparameter sensitivity analysis of r with $\alpha = 32$. All experiments are conducted on the VisDrone dataset.

Table 7: Ablation study of time cost during inference on the VisDrone dataset.

Method	Latency↓	FPS↑
Baseline:		
Focus-DETR	66.21ms	15.10
Ours:		
MoRE-DETR (R=3, N_r=1)	56.92ms	17.57
MoRE-DETR (R=2, N_r=2)	58.60ms	17.07
MoRE-DETR (R=3, N_r=2)	59.54ms	16.80

naNetLin et al. (2017), SPARLi et al. (2025a), Faster RCNNRen et al. (2015), Cascade RCNNCai & Vasconcelos (2018), DetectoRSQiao et al. (2021), YOLO-WorldCheng et al. (2024), RFLAXu et al. (2022), KLDetZhou & Zhu (2024), SR-TODCao et al. (2024), VLM4RSDetShi et al. (2026). The second category consists of Transformer-based methods, including Deformable DETRZhu et al. (2020), DN-DETRLi et al. (2022), RT-DETRv1Zhao et al. (2024), RT-DETRv4Liao et al. (2025), LW-DETRChen et al. (2024), Focus-DETRZheng et al. (2023), D-FINEPeng et al. (2025), DEIMHuang et al. (2025b), DEIMv2Huang et al. (2025a), Salience DETRHou et al. (2024), SKYDETZhang et al. (2026b), Florence-2Xiao et al. (2024), Grounding DINOLiu et al. (2024), LLMDetFu et al. (2025), SMWG-DETRChen et al. (2026), LWGA-RT-DETRLu et al. (2026), RSRT-DETRZhao et al. (2026).

Hyperparameter sensitivity analysis in visual adapter. We investigate the sensitivity of the two hyperparameters, r and α, that determine the LoRA scaling factor α/r on the VisDrone dataset. Here, r denotes the rank of the low-rank update, while α is the scaling parameter. As shown in Fig. 6 (a), with $r = 8$ fixed, both mAP and mAP_{50} gradually decrease as the scaling factor increases beyond 1.0, indicating that an excessively large LoRA update may adversely affect the pretrained representations. In Fig. 6 (b), with $\alpha = 32$ fixed, we further evaluate six different rank settings. The best performance is achieved at $r = 24$, corresponding to a scaling factor of $\alpha/r = 1.33$. Overall, these results suggest that a moderate LoRA scaling factor is beneficial for adapting the pretrained projection. In our main experiments, we set $r = 8$ and $\alpha = 8$.

Analysis of time cost during inference. As shown in Table 7, we further compare the inference cost of our method with the baseline on the VisDrone dataset. We use the Expert-Choice routing strategy, and all experiments are conducted on a single NVIDIA RTX 3090 GPU with a batch size of 1. During inference, each image is resized while preserving its aspect ratio, with the longest side set to 1024 pixels. Our method consistently achieves lower latency and higher FPS than the baseline under different recursive configurations.

Additional visualization comparison. As shown in Fig. 7, we further provide qualitative comparisons among MoRE-DETR, the baseline Focus-DETR, and the frequency-domain state-of-the-art method SMWG-DETR on the VisDrone dataset. VisDrone contains diverse scenes and a wide range of object appearances, making it a challenging benchmark for remote sensing object detection. We select four challenging examples for comparison. As shown in the visualization results, MoRE-DETR successfully detects several objects that are missed by both Focus-DETR and SMWG-DETR, while producing fewer false positives and false negatives. These results further demonstrate the effectiveness of MoRE-DETR in complex remote sensing scenes.

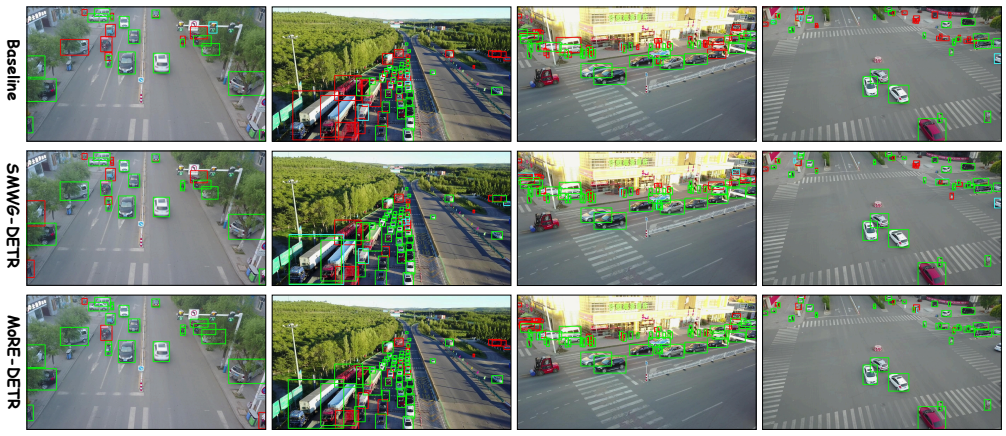

Figure 7: Visualization comparison among the baseline, SMWG-DETR and our method on the VisDrone dataset. Green, blue, and red rectangles indicate True Positives (TPs), False Positives (FPs), and False Negatives (FNs), respectively, with an IoU threshold of 0.5 for determining correct detections. *Zoom in for best view.*

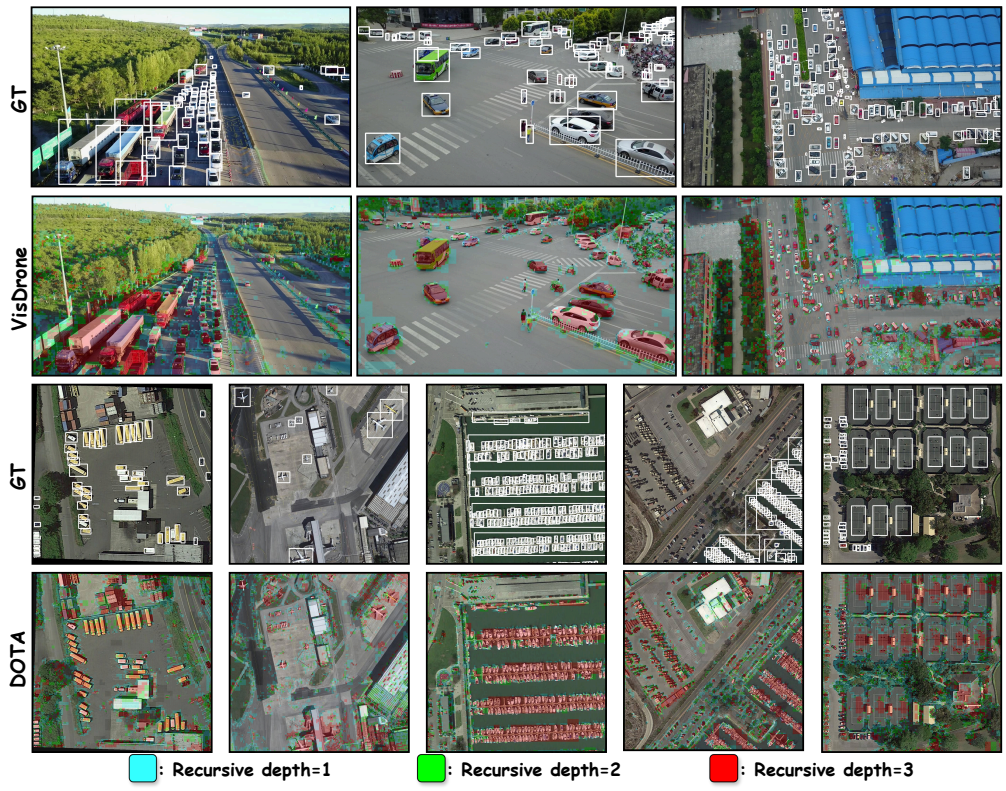

Figure 8: Token-wise recursive depth visualization on the DOTA-1.0 and VisDrone datasets. White rectangles denote ground-truth bounding boxes. Blue, green, and red regions indicate tokens with recursive depths of one, two, and three, respectively. *Zoom in for best view.*

Recursion depth visualization. As shown in Fig. 8, we visualize the token-wise recursive depth on the DOTA-1.0 and VisDrone datasets. The recursive depth denotes the number of times a foreground token is processed by the shared recursive blocks. Specifically, blue, green, and red regions correspond to tokens with recursive depths of one, two, and three, respectively, while the remaining

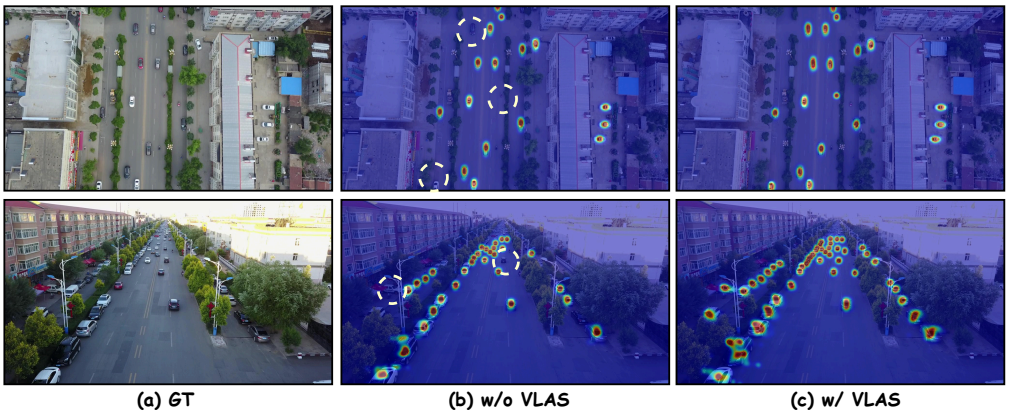

Figure 9: Qualitative ablation study of the VLAS module. (a) Ground-truth images; (b) results without VLAS; (c) results with VLAS.

regions represent tokens that do not enter the recursive blocks. As shown in this picture, tokens with higher recursive depths are mainly distributed around ground-truth objects, but also appear in densely populated regions and structurally complex areas, such as object boundaries and occluded regions. Moreover, even for large objects such as "airplane" and "tennis court", only part of the object region receives deeper recursive computation. These observations indicate that the uncertainty-aware MoR router adaptively allocates more computation to challenging tokens while allowing relatively easy tokens to exit early, rather than treating all foreground tokens equally. This supports the effectiveness of introducing the MoR mechanism into remote sensing object detection.

Qualitative analysis of the VLAS module. To further investigate the effectiveness of the VLAS module, we visualize the attention maps of true-positive (TP) queries to compare their attention patterns with and without VLAS on the VisDrone dataset, as shown in Fig. 9. This visualization serves as a qualitative complement to the quantitative ablation results reported in Table 4. Yellow dashed circles highlight representative object regions that receive relatively weaker attention without VLAS. By comparing the attention heatmaps under this two different settings, we observe that VLAS leads to more pronounced attention responses around these target regions, providing qualitative evidence of its effectiveness.

A.4 Limitations and Future Work

Our method combines adaptive recursive token reasoning with vision-language supervision, dynamically allocating computation to challenging foreground tokens while incorporating pretrained VLM priors to enhance their representations. Despite its promising performance, several limitations remain. First, we introduce the MoR mechanism only into the encoder of the DETR architecture, while its potential in the decoder remains unexplored. In addition, the current token difficulty estimation mainly relies on an MLP and classification uncertainty, which may not fully capture the factors that determine whether a token is critical to the final detection performance. Future work could explore more sophisticated token selection mechanisms, such as causal reasoning approaches based on counterfactual reasoning or backdoor adjustment, to identify tokens that have a more direct impact on detection outcomes and enable more effective computation allocation.

Second, the VLM adopted in our framework is relatively lightweight and has limited capability compared with more advanced vision-language models. Moreover, the general-purpose VLM is not specifically optimized for remote sensing imagery, leaving room for further improvement in domain adaptation and visual-language alignment. In future work, we will investigate stronger VLMs and lightweight adaptation strategies, such as knowledge distillation and model quantization, followed by task-specific fine-tuning on remote sensing data.

Despite existing limitations, we hope that this work provides valuable insights for remote sensing object detection and inspires further research.