# MLLM Makes Strong Backbone for Multi-Modal Object Detection

**Yifeng Yang**[1] **Yutong Li**[1] **Jubo Feng**[1] **Qinying Gu**[2]
**Cewu Lu**[1,3] **Guang-Zhong Yang**[1] **Xinbing Wang**[1] **Nanyang Ye**[1,2,3*]
[1]Shanghai Jiao Tong University, Shanghai, China
[2]Shanghai Artificial Intelligence Laboratory, Shanghai, China
[3]Shanghai Innovation Institute, Shanghai, China

## Abstract

Multimodal large language models (MLLMs) provide a promising alternative by taking object detection as a sequence generation task. However, applying MLLMs to multimodal object detection poses challenges in visual cross-modal (RGB-Infrared) fusion and coordinate prediction under Cross-Entropy supervision. In this paper, we propose a novel MLLM-based object detection framework that addresses these challenges through improving visual cross-modal token alignment and coordinate generation. Specifically, we formulate RGB-IR fusion as pre-decoding token-space modality routing, implemented by the *Cross-Modal Complementary Adapters* and the *Visual Token-wise Router*. We further reformulate coordinate-token learning as scale-adaptive neighborhood likelihood maximization through our proposed *Geometry-Aware Loss*, bridging discrete token prediction and continuous geometric localization. Extensive experiments show that our method achieves state-of-the-art performance, reaching 67.81% mAP@0.5 under in-distribution scenarios, outperforming existing baselines by over 8%, while also exhibiting strong generalization to out-of-distribution and referring object detection tasks.

## 1 Introduction

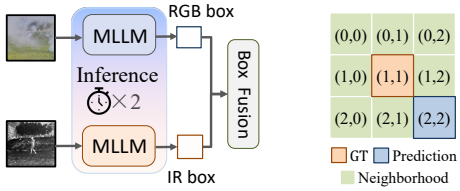
Object detection is a fundamental task in computer vision [25, 37, 18], yet RGB-based detectors often fail under adverse conditions such as low illumination, rain, and fog [11]. Multimodal object detection has therefore emerged as a promising solution by combining rich visual details from RGB modality with illumination-invariant thermal cues from infrared (IR) modality [20]. This complementary fusion has shown great potential in safety-critical scenarios such as autonomous driving [9], remote sensing [34, 27], and UAV perception [31].

Recently, RGB-IR fusion strategies in traditional multi-modal object detection have been well investigated across different stages of the detection pipeline [7, 28, 4, 36]. Despite these advances, such methods remain constrained by the architectural design of traditional backbone-based detectors. In particular, their detection heads operate on a fixed set of predefined categories, which prevents open-set object detection. To overcome this limitation, multimodal large language models (MLLMs) shift detection from traditional coordinate regression to a more general detection-as-generation paradigm, where pioneering works such as Pix2Seq [6] and Rex-Omni [19] demonstrate that serializing bounding boxes into discrete token sequences enables open-vocabulary object detection with enhanced semantic understanding. However, as shown in Figure 1, directly extending MLLMs to multimodal object detection remains largely unexplored and faces two critical challenges: **(1) Architectural gap**: Existing MLLMs are primarily designed for single image input, lacking effective mechanisms for visual cross-modal fusion. **(2) Geometric objective gap**: Discrete next-token generation is inherently misaligned

*Corresponding author: ynylincoln@sjtu.edu.cn. Code: https://github.com/zxk1212/rgb_ir.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

with the continuous nature of bounding box prediction. Therefore, optimizing coordinate tokens with standard Cross-Entropy loss imposes overly rigid supervision for a distance-based geometric task. Motivated by these challenges, we introduce a unified token-space fusion framework of RGB-IR object detection with MLLMs, operating jointly across two axes of the token space. **(1) On the visual side:** we introduce a parameter-efficient module, termed Cross-Modal Complementary Adapters (CCA, in §3.2), to generate modality-enhanced tokens by bidirectionally injecting complementary RGB and IR cues. A Visual Token-wise Router (in §3.3) then performs a binary token-level selection between the RGB-enhanced and IR-enhanced tokens at each spatial position. This converts RGB-IR fusion from box fusion into pre-decoding token-level modality routing. Instead of decoding RGB and IR branches separately with the LLM decoder and merging their predicted boxes afterward, our method produces a routed visual token sequence with the same token count as a single-image input for subsequent decoding, thereby substantially reducing LLM decoding cost. **(2) On the localization side:** By extending the vocabulary with atomic coordinate tokens in §3.4, we avoid multi-digit tokenization collisions. We then replace the probability of the ground-truth coordinate token in the Cross-Entropy loss with the summed probability of nearby coordinate tokens in §3.5, bridging next-token prediction and numerical coordinates and provide theoretical guarantees.




(a) MLLM-based object detection is designed for a single image. (b) Coordinates should allow for deviation.

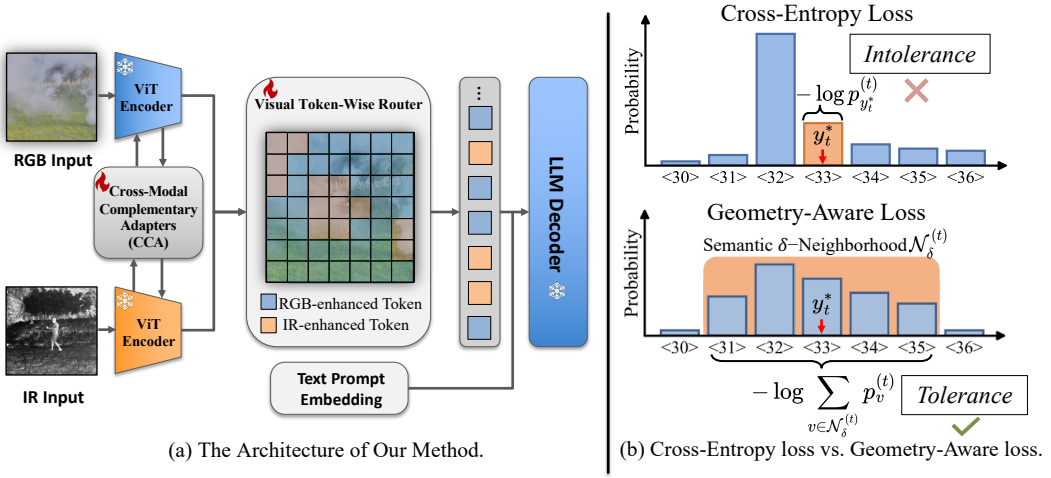
Figure 1: Challenges in extending MLLMs to multimodal object detection. (a) Existing MLLMs can only process RGB and IR images separately and perform fusion at the bounding-box level, which prevents visual cross-modal interaction and incurs redundant inference. (b) Discrete token supervision is rigid. For example, a prediction at (2, 2) is very close to the ground truth (GT) (1, 1) and should be considered acceptable.

Eventually, our method establishes a new state of the art on in-distribution common object detection benchmarks, achieving **67.81%** mAP@0.5 and surpassing the second-best baseline by more than **8%**. Moreover, it maintains strong performance on out-of-distribution object detection and referring object detection tasks without requiring fine-tuning of LLM decoder, highlighting its generalization capability.

## 2 Related Work

**MLLMs for Object Detection.** With the rapid advancement of MLLMs [1, 3, 2], object detection is increasingly framed as a detection-as-generation problem rather than continuous coordinate regression. Pix2Seq [6] pioneers this paradigm by unifying detection into a sequence generation task through bounding box tokenization. Building upon this, LLMdet [10] leverages region- and image-level captions to improve semantic alignment and generalization to unseen categories, while Rex-Omni [19] proposes a next-point prediction framework that unifies spatial tasks via coordinate quantization and reinforcement learning. However, existing MLLMs still face a fundamental mismatch between next-token optimization and numerical coordinate prediction. String-based methods such as NTL [32] and RAFT [23] treat multi-digit values as token sequences optimized with Mean Squared Error (MSE) loss, but are sensitive to digit-level positional weighting and misaligned with the Cross-Entropy loss. To our best knowledge, these numerical modeling strategies have not yet been perfectly applied to object detection, leaving a critical gap between sequence-based generation and numerical coordinates.

**Multimodal Object Detection.** Complementary fusion between RGB and IR modalities is crucial for multimodal object detection. Existing fusion strategies are generally categorized into pixel-level, feature-level, and bounding-box-level fusion [28]. Early fusion methods, such as pixel-wise concatenation, are conceptually straightforward but highly sensitive to spatial misalignment between modalities [36]. At the feature level, methods such as ADCNet [12] and OAFA [5] emphasize spatial alignment across modalities. However, these approaches typically rely on static mid-fusion backbones, which limits their ability to adapt to dynamic scene statistics. To overcome this limitation, MoE-based dynamic fusion strategies [4] have been proposed, introducing local-to-global experts that are dynamically aggregated via a gating network to adaptively preserve texture and thermal-contrast information. Our approach extends this line of work by introducing a token-wise router for finer-grained adaptive fusion. In contrast, bounding-box–level fusion relies on post-processing




(a) The Architecture of Our Method.

(b) Cross-Entropy loss vs. Geometry-Aware loss.

Figure 2: Overview of the proposed framework. (a) The overall architecture, where paired RGB and infrared images are encoded into visual tokens, enhanced through cross-modal interaction, and selectively routed to form a unified visual token sequence for LLM decoder. (b) Illustration of the semantic $\delta$-neighborhood used in geometry-aware supervision (see §3.5 for the definition of $\delta$), which relaxes strict point-wise matching imposed by Cross-Entropy loss and introduces scale-adaptive error tolerance for predictions falling within the neighborhood of the ground truth token.

techniques, such as non-maximum suppression (NMS) or average pooling, to merge detection results from different modalities [7].

## 3 Method

The overall framework adopts a unified detection-as-generation paradigm built upon an MLLM backbone, as illustrated in Figure 2. RGB and infrared images are encoded into visual token sequences and projected into the LLM embedding space (see §3.1), where they undergo token-level cross-modal interactions (see §3.2-§3.3). The training objective further incorporates geometric constraints to better align discrete token prediction with spatial coordinates (see §3.4-§3.5).

### 3.1 Preliminaries: From Patch to Token

Let $\mathcal{X} \subset \mathbb{R}^{H\times W\times C}$ be the image space, and $(\mathbf{x}_{\text{rgb}}, \mathbf{x}_{\text{ir}}) \in \mathcal{X}^2$ be a spatially aligned RGB-IR pair. We use a ViT-style visual encoder (as in Qwen2.5-VL [3]), composed of a patch feature extractor $\Phi$ and a merger $\mathcal{M}$. Given an image $\mathbf{x} \in \mathcal{X}$, the feature extractor produces patch features $\mathbf{Z} = \Phi(\mathbf{x}) \in \mathbb{R}^{d_v\times L}$, where $d_v$ is the visual embedding dimension and $L$ is the number of ViT patches (determined by image resolution and patch size $P$). To reduce redundancy before feeding features into the LLM, the merger $\mathcal{M}$ compresses the sequence by aggregating non-overlapping $k \times k$ spatial windows, producing $N = L/k^2$ tokens in the LLM hidden dimension $d_l$. Let $\text{Reshape}(\cdot)$ rearrange $\mathbb{R}^{d_v\times L}$ into $\mathbb{R}^{k^2 d_v\times N}$ by grouping features inside each $k \times k$ window and $\text{LN}(\cdot)$ denote layer normalization (e.g., RMSNorm). The grouped features are projected to the LLM hidden dimension $d_l$ via a two-layer MLP with weights $\mathbf{W}_1 \in \mathbb{R}^{k^2 d_v\times k^2 d_v}$ and $\mathbf{W}_2 \in \mathbb{R}^{d_l\times k^2 d_v}$, the visual token $\mathbf{V}$ is computed as:

$$\mathbf{V} = \mathbf{W}_2 \cdot \text{GeLU}\left(\mathbf{W}_1 \cdot \text{Reshape}\big(\text{LN}(\mathbf{Z})\big)\right), \tag{1}$$

where GeLU is the GeLU [13] activation.

### 3.2 Cross-Modal Complementary Adapters

To effectively integrate the complementary cues between RGB and IR modalities, we introduce ***Cross-Modal Complementary Adapters (CCA)***. CCA is designed to enable a bidirectional flow of information and compensate for the weaknesses of one modality with the strengths of the other. Specifically, let $\mathbf{Z}_{\text{rgb}}$, $\mathbf{Z}_{\text{ir}} \in \mathbb{R}^{d_v\times L}$ denote the extracted visual features obtained from RGB-IR

pair $(\mathbf{x}_{\text{rgb}}, \mathbf{x}_{\text{ir}})$, respectively. We define two symmetric cross-modal adapters $\mathbf{A}_{\text{ir}\to\text{rgb}}$ and $\mathbf{A}_{\text{rgb}\to\text{ir}}$ (structurally identical to $\mathcal{M}$ for architectural alignment) that operate in a residual manner to enhance the visual tokens $\mathbf{V}_{\text{rgb}}$ and $\mathbf{V}_{\text{ir}}$:

$$\hat{\mathbf{V}}_{\text{rgb}} = \mathbf{V}_{\text{rgb}} + \gamma_r \cdot \mathbf{A}_{\text{ir}\to\text{rgb}}(\mathbf{Z}_{\text{ir}}), \tag{2}$$

$$\hat{\mathbf{V}}_{\text{ir}} = \mathbf{V}_{\text{ir}} + \gamma_i \cdot \mathbf{A}_{\text{rgb}\to\text{ir}}(\mathbf{Z}_{\text{rgb}}), \tag{3}$$

where $\gamma_r$ and $\gamma_i$ are learnable gating scalars constrained by the Sigmoid function to adaptively control the intensity of cross-modal injection. Crucially, we adopt a zero-initialization strategy for both the final layers of the adapters, enabling a smooth shift from the pre-trained identity mapping to a learned complementary cross-modal fusion, avoiding catastrophic interference with the base MLLM's capabilities during the initial training phase.

### 3.3 Visual Token-wise Router

Given the RGB-enhanced and IR-enhanced tokens generated by CCA, the key question is how to expose them to the LLM decoder without doubling the decoding burden. To this end, we propose the *Visual Token-wise Router*, designed for adaptive token selection to achieve effective RGB-IR fusion within MLLMs. Unlike existing RGB-IR fusion methods, our approach is the first to leverage the inherent capabilities of MLLMs for modality-specific routing rather than feature concatenation or late box fusion. Specifically, after CCA processing, the RGB-enhanced and IR-enhanced tokens $\hat{\mathbf{V}}_{\text{rgb}}$ and $\hat{\mathbf{V}}_{\text{ir}}$ are treated as two modality-specific experts. We then fuse the tokens by applying $\text{Cat}(\cdot)$ operator to concatenate them along the LLM's hidden dimension:

$$\hat{\mathbf{V}} = \text{Cat}\left(\hat{\mathbf{V}}_{\text{rgb}},\ \hat{\mathbf{V}}_{\text{ir}}\right) \in \mathbb{R}^{2d_l \times N}. \tag{4}$$

The router $\mathcal{R}$ takes the concatenated token embeddings as input and predicts token-wise routing logits for each modality. While inspired by Mixture-of-Experts (MoE [15]) models, our design differs fundamentally in that expert selection is modality-specific and token-wise, rather than example-wise or task-wise. The router $\mathcal{R}$ produces token-wise routing logits $\mathbf{L} \in \mathbb{R}^{2\times N}$:

$$\mathbf{L} = \mathcal{R}(\hat{\mathbf{V}}) = \hat{\mathbf{W}}_2 \cdot \text{GeLU}\left(\hat{\mathbf{W}}_1 \cdot \text{LN}(\hat{\mathbf{V}})\right), \tag{5}$$

where $\hat{\mathbf{W}}_1 \in \mathbb{R}^{d_l \times 2d_l}$, $\hat{\mathbf{W}}_2 \in \mathbb{R}^{2\times d_l}$ are learnable parameters. To learn discrete expert selection while maintaining differentiability, we employ the Gumbel-Softmax with Straight-Through Estimation (STE) [16, 24]. Based on the router logits $\mathbf{L}$, in the forward pass, we apply a hard $\arg\max$ over the modality dimension to generate a binary mask $\mathbf{M}_{\text{hard}} \in \mathbb{R}^{2\times N}$. During backpropagation, gradients flow through a continuous softmax approximation $\mathbf{M}_{\text{soft}} \in \mathbb{R}^{2\times N}$:

$$\mathbf{M} = \mathbf{M}_{\text{hard}} - \text{sg}(\mathbf{M}_{\text{soft}}) + \mathbf{M}_{\text{soft}}, \tag{6}$$

where $\text{sg}(\cdot)$ denotes the stop-gradient operation. By slicing $\mathbf{M} \in \mathbb{R}^{2\times N}$ into modality-specific masks $\mathbf{M}_{\text{rgb}} \in \mathbb{R}^{1\times N}$ and $\mathbf{M}_{\text{ir}} \in \mathbb{R}^{1\times N}$, we compute the final routed tokens:

$$\hat{\mathbf{V}}_{\text{out}} = \mathbf{M}_{\text{rgb}} \odot \hat{\mathbf{V}}_{\text{rgb}} + \mathbf{M}_{\text{ir}} \odot \hat{\mathbf{V}}_{\text{ir}}, \tag{7}$$

where $\odot$ denotes element-wise multiplication. After routing, the two modality visual tokens aligned at the same patch position are reduced to one single selected token. This prevents LLM decoder from performing redundant inference on two modality tokens that correspond to the same patch position.

### 3.4 Atomic Coordinate Tokenization

Let $\mathcal{Y} = \{\langle 0\rangle, \langle 1\rangle, \ldots, \langle 999\rangle\}$ denote the set of coordinate tokens, where each token $\langle i\rangle$ ($i \in [0, 999]$) corresponds to a numerical value. We adopt the coordinate representation convention from Qwen3-VL series, utilizing a resolution-independent normalized range of $[0, 999]$ for the $[x_1, y_1, x_2, y_2]$ format. However, traditional string-based numerical representations often suffer from severe discretization bias during tokenization, where multi-digit numbers are fragmented into independent digit tokens. This fragmentation causes a misalignment between the model's perceived digit-wise distance and the actual geometric distance. For instance, due to the tokenizer's splitting mechanism, the digit-wise distance between "101" and "201" ($|1-2| + |0-0| + |1-1| = 1$) is misleadingly smaller than that

between "30" and "39" ($|3-3|+|0-9|=9$). To resolve this, we employ vocabulary expansion [8, 19] by adding coordinate tokens to achieve an atomic representation of coordinates. Specifically, we introduce 1,000 unique tokens, $\langle 0 \rangle$ to $\langle 999 \rangle$, into the vocabulary to establish a one-to-one mapping between numerical values and coordinate tokens. This approach fundamentally eliminates coordinate token fragmentation, ensuring the model to learn spatial relationships directly within the token space. Additionally, this atomic representation reduces the sequence length for coordinate prediction, thereby lowering inference latency.

### 3.5 Geometry-Aware Loss

Standard Cross-Entropy loss treats coordinate prediction as a categorical classification problem with mutually independent indices, thereby ignoring the inherent spatial continuity between neighboring coordinates. As a consequence, when the ground-truth token is $\langle 33 \rangle$, a prediction of an adjacent coordinate (e.g., $\langle 32 \rangle$) may be penalized as severely as a distant error (e.g., $\langle 900 \rangle$), despite their close spatial proximity. This mismatch leads to unstable gradients during training, particularly when the model exhibits high semantic confidence but slight spatial deviations. To address this issue, we propose the *Geometry-Aware Loss* $\mathcal{L}_{\text{geo}}$ (Equation 10), which relaxes the strict point-wise supervision by introducing a dynamic semantic neighborhood for coordinate tokens. During training, the optimization objective for each generated token at time step $t$ is adjusted based on whether it is a coordinate token.

**Definition 3.1.** Let $\mathcal{Y}$ be the coordinate token space and $y_t^*$ denote the ground-truth token at time step $t$. When $y_t^* \in \mathcal{Y}$ and $\delta \geq 0$, we define the semantic $\delta$-neighborhood as:

$$\mathcal{N}_\delta^{(t)} = \{y \in \mathcal{Y} \mid |\text{val}(y) - \text{val}(y_t^*)| \leq \delta\}, \tag{8}$$

where $\text{val}(\cdot)$ denotes the numeric value represented by a coordinate token. Let $\hat{y}_t$ be the token sampled from the model-generated token probability distribution $\mathbf{p}$ at time step $t$. The generated token distribution $\mathbf{p}$ is defined as $(\epsilon, \delta)$-confident if the probability of the event $\hat{y}_t \in \mathcal{N}_\delta^{(t)}$ satisfies:

$$\mathbb{P}(\hat{y}_t \in \mathcal{N}_\delta^{(t)}) \geq 1 - \epsilon \tag{9}$$

We then denote the family of $(\epsilon, \delta)$-confident distributions at time step $t$ by $\mathcal{F}_t(\epsilon, \delta)$.

To account for object scale, $\delta$ is dynamically adjusted according to the ground-truth bounding box. Let $w$ and $h$ denote the width and height of the bounding box respectively. We set $\delta = \lfloor \lambda w \rfloor + \mathbf{1}_{\{\lambda > 0\}}$ for the $x$-coordinates and $\delta = \lfloor \lambda h \rfloor + \mathbf{1}_{\{\lambda > 0\}}$ for the $y$-coordinates, where $\lambda \geq 0$ is a hyperparameter and $\mathbf{1}_{(\cdot)}$ is the indicator function. This ensures a minimum tolerance of 1 whenever $\lambda > 0$. By expanding point-wise coordinate predictions into semantic regions, this design facilitates a spatial tolerance proportional to the object size. Accordingly, for the generated $t$-th token, *the Geometry-Aware loss* is formulated as:

$$\mathcal{L}_{\text{geo}}^{(t)}(\mathbf{p}, y_t^*) = \begin{cases} -\log p_{y_t^*}^{(t)}, & \text{if } y_t^* \text{ is not coordinate,} \\ -\log \sum_{v \in \mathcal{N}_\delta^{(t)}} p_v^{(t)}, & \text{otherwise,} \end{cases} \tag{10}$$

where $\mathbf{p}$ and $p_v^{(t)}$ denote the generated token probability distribution and the predicted probability of token $v$ at time step $t$, respectively.

**Theorem 1** (Convergence Analysis)**.** *For any step $t$ where the ground-truth token is a coordinate, if $\mathcal{F}_t(\epsilon, \delta)$ is the family of $(\epsilon, \delta)$-confident distributions, then for the Geometry-Aware loss $\mathcal{L}_{geo}^{(t)}$ and Cross-Entropy loss $\mathcal{L}_{ce}^{(t)}$, the following holds:*

$$\begin{aligned} \sup_{\mathbf{p} \in \mathcal{F}_t(\epsilon,\delta)} \lim_{\epsilon \to 0} \mathcal{L}_{geo}^{(t)}(\mathbf{p}, y_t^*) = 0, \\ \sup_{\mathbf{p} \in \mathcal{F}_t(\epsilon,\delta)} \lim_{\epsilon \to 0} \mathcal{L}_{ce}^{(t)}(\mathbf{p}, y_t^*) = \infty. \end{aligned} \tag{11}$$

*This implies that $\mathcal{L}_{geo}^{(t)}$ vanishes as the probability mass concentrates within the semantic $\delta$-neighborhood $\mathcal{N}_{\delta}^{(t)}$, while $\mathcal{L}_{ce}^{(t)}$ remains unbounded regardless of the total probability mass assigned to the semantic $\delta$-neighborhood $\mathcal{N}_{\delta}^{(t)}$.*

The proof of Theorem 1 is provided in Appendix D. Theorem 1 demonstrates that $\mathcal{L}_{\text{geo}}$ provides a more stable optimization landscape by tolerating minor spatial shifts, whereas Cross-Entropy loss remains overly sensitive to point-wise deviations even when the model is spatially accurate. This ensures that the training process is not disrupted by negligible coordinate offsets when $\mathcal{L}_{\text{geo}}$ is applied. Since a bounding box is specified by four coordinate tokens, the probability mass accumulated within the corresponding semantic $\delta$-neighborhood $\mathcal{N}_{\delta}^{(t)}$ for each coordinate naturally characterizes the confidence of that coordinate prediction. Accordingly, we use the average semantic $\delta$-neighborhood probability over the four coordinates as the overall confidence estimate of the bounding box.

# 4 Experiments

## 4.1 Experimental Setting

We conduct experiments on two tasks: (i) common object detection and (ii) referring object detection. **(i) Common object detection:** We evaluate our method under both In-Distribution (*In-Dist.*) and Out-of-Distribution (*Out-of-Dist.*) protocols to comprehensively assess closed-set in-domain performance as well as generalization under distribution shift. The In-Dist. benchmark consists of three widely used aligned RGB-IR datasets: M³FD [21], FLIR-Aligned [33], and LLVIP [17]. Following prior work [4], we adopt the same training and validation splits for all methods, and **train exclusively on the In-Dist. training set** throughout all experiments. To evaluate the generalization under distribution shift, we further test our approach on MSRS [30] and VTUAV [35] as Out-of-Dist. datasets, which differ markedly from the In-Dist. benchmark in terms of camera viewpoints and scene distributions. **(ii) Referring Object Detection:** Since no public RGB-IR referring object detection benchmark is currently available, we construct a new evaluation dataset based on the In-Dist. validation sets using a fully automated multi-agent pipeline to generate 3,429 natural language referring expressions. To ensure fairness and eliminate human annotation bias, the entire dataset construction process is fully automatic. Further details are provided in Appendix F.

**Baseline Methods.** We benchmark our method against three categories of mainstream baselines: (i) Unimodal detectors, specifically Faster R-CNN [26], to establish performance limits for individual modalities; (ii) Conventional state-of-the-art RGB-IR fusion via MoE-Fusion [4]; and (iii) Grounding-based semantic models, including the open-set detector Grounding DINO (gDino-T) [22] as well as large-scale MLLMs such as Qwen3-VL [2], and detection-specialized MLLMs like LLMdet [10] and Rex-Omni [19]. Furthermore, for multimodal bounding box fusion, inspired by [7], we employ *Non-Maximum Suppression* (NMS) and *Average Box Pooling* (Pooling), more details can be found in Appendix E.

**Implementation Details.** We initialize our model from the Rex-Omni pretrained model built upon Qwen2.5-VL [3] and trained on a cluster of 8 NVIDIA H100 GPUs. The model is trained for a total of 4 epochs with a global batch size of 128. Optimization is performed using AdamW with a weight decay of 0.01 and a cosine learning rate decay schedule with 10 warm-up steps. We set $\lambda = 0.05$ by default for the semantic $\delta$-neighborhood. To ensure stable optimization of the Visual Token Router, a progressive two-stage training scheme is applied. During Stage I, only the two submodules of the CCA module are optimized independently using a learning rate of $1 \times 10^{-3}$ for 2 epochs, while all other parameters are frozen. In Stage II, the Visual Token Router is activated and trained for 2 epochs with a learning rate of $1 \times 10^{-3}$. Meanwhile, the learning rate of the CCA module is decayed to $1 \times 10^{-6}$ to preserve previously learned representations while adapting to the new module. All other modules (especially LLM decoder) are kept frozen throughout both stages. More details can be found in Appendix I.

**Evaluation Metrics.** For comparison among different methods, we follow the evaluation protocols adopted in [4]. We first report results under the standard COCO evaluation protocol, where mean Average Precision at an IoU threshold of 0.5 (mAP@0.5) is measured. These mAP-based metrics rely on confidence scores to dynamically balance precision and recall. Nevertheless, confidence estimation in MLLMs lacks a unified and fair calibration standard. Consequently, we also adopt the

Table 1: Evaluation results on the In-Distribution (In-Dist.) common object detection benchmark. Results are reported in terms of mAP@0.5, precision (P), recall (R), and F1-score at IoU thresholds of 0.5, 0.95 and mIoU. For mIoU-based metrics, the mIoU is computed by averaging IoU thresholds from 0.5 to 0.95 with a step size of 0.05. Best results are highlighted in bold.

| Type | Method | Decision | In-Distribution (In-Dist.) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | mAP@0.5 | P@IoU 0.5 | R@IoU 0.5 | F1@IoU 0.5 | P@IoU 0.95 | R@IoU 0.95 | F1@IoU 0.95 | P@ mIoU | R@ mIoU | F1@ mIoU |
| Closed-set | Visible | RGB-only | 49.24 | 71.22 | 53.70 | 61.23 | 11.35 | 2.62 | 4.26 | 52.61 | 31.92 | 39.73 |
| | Infrared | IR-only | 51.30 | 55.63 | 56.35 | 55.99 | 10.97 | 4.06 | 5.93 | 50.48 | 32.57 | 39.59 |
| | MoE-Fusion | Early | 51.58 | 44.19 | 61.33 | 51.37 | 17.55 | 3.15 | 5.35 | 31.71 | 33.84 | 32.74 |
| Open-set | gDino-T | RGB | 57.68 | 73.68 | 63.94 | 68.47 | 20.27 | 7.05 | 10.46 | 50.15 | 41.31 | 45.30 |
| | | Infrared | 40.05 | 70.87 | 44.97 | 55.02 | 12.07 | 6.58 | 8.52 | 50.93 | 29.17 | 37.09 |
| | | NMS | 59.32 | 59.74 | 69.30 | 64.17 | 14.75 | 8.28 | 10.60 | 42.90 | 45.78 | 44.30 |
| | | Pooling | 58.05 | 59.71 | 69.25 | 64.13 | 15.59 | 8.61 | 11.09 | 42.05 | 46.20 | 44.03 |
| MLLM | Qwen3-VL-4B | NMS | 32.91 | 60.30 | 49.28 | 54.24 | 3.05 | 2.45 | 2.72 | 38.58 | 29.27 | 33.29 |
| | | Pooling | 32.91 | 60.28 | 49.31 | 54.24 | 3.85 | 2.75 | 3.21 | 39.17 | 29.79 | 33.84 |
| | Qwen3-VL-8B | NMS | 36.39 | 58.95 | 56.56 | 57.73 | 5.27 | 3.89 | 4.47 | 37.47 | 34.08 | 35.70 |
| | | Pooling | 36.26 | 59.05 | 56.54 | 57.77 | 3.10 | 2.64 | 2.85 | 37.98 | 34.15 | 35.96 |
| | LLMdet | NMS | 59.70 | 45.37 | 74.61 | 56.43 | 12.87 | 5.19 | 7.40 | 48.31 | 33.15 | 39.32 |
| | | Pooling | 58.60 | 45.67 | **74.65** | 56.67 | 12.99 | 4.56 | 6.75 | 49.11 | 39.18 | 39.18 |
| | Rex-Omni | RGB | 56.22 | 75.32 | 63.70 | 69.03 | 26.25 | 5.75 | 9.44 | 50.82 | 38.52 | 43.83 |
| | | Infrared | 41.29 | 70.21 | 46.58 | 56.00 | 27.80 | 4.79 | 8.18 | 50.42 | 29.65 | 37.34 |
| | | NMS | 57.90 | 59.08 | 71.58 | 64.73 | 14.10 | 6.51 | 8.91 | 41.40 | 45.29 | 43.26 |
| | | Pooling | 56.69 | 59.21 | 71.68 | 64.85 | 15.05 | 7.10 | 9.64 | 41.51 | 45.66 | 43.48 |
| | Ours | Router | **67.81** | **79.53** | 74.48 | **76.92** | **30.73** | **7.81** | **12.46** | **56.41** | **45.78** | **50.54** |

Table 2: Evaluation results on the Out-of-Distribution (Out-of-Dist.) common object detection benchmark. Best results are highlighted in bold.

| Type | Method | Decision | MRSR | | | | VTUAV | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | mAP@0.5 | P@ mIoU | R@ mIoU | F1@ mIoU | mAP@0.5 | P@ mIoU | R@ mIoU | F1@ mIoU |
| Open-set | gDino-T | NMS | 89.43 | 42.52 | 76.77 | 54.73 | 11.50 | 10.58 | 9.48 | 10.00 |
| | | Pooling | 88.04 | 41.26 | 78.73 | 54.15 | 11.18 | 10.70 | 9.58 | 10.11 |
| MLLM | Qwen3-VL-4B | NMS | 74.68 | 37.67 | 67.17 | 48.27 | 8.32 | 12.01 | 7.54 | 9.26 |
| | | Pooling | 73.60 | 38.84 | 68.91 | 49.68 | 8.26 | 12.02 | 7.55 | 9.27 |
| | Qwen3-VL-8B | NMS | 65.02 | 37.19 | 68.01 | 48.09 | 8.01 | 10.51 | 7.96 | 9.06 |
| | | Pooling | 64.14 | 34.38 | 67.59 | 45.58 | 8.12 | 10.52 | 7.94 | 9.05 |
| | LLMdet | NMS | 91.43 | 33.64 | **82.69** | 47.83 | 14.51 | 8.22 | 15.25 | 10.69 |
| | | Pooling | 89.40 | 30.91 | 81.66 | 44.85 | 14.42 | 8.27 | **15.26** | 10.73 |
| | Rex-Omni | NMS | 83.75 | 49.06 | 73.78 | 58.71 | 13.31 | 9.49 | 13.37 | 11.10 |
| | | Pooling | 82.09 | 44.62 | 73.40 | 55.50 | 13.01 | 9.83 | 13.46 | 11.36 |
| | Ours | Router | **92.82** | **56.01** | 75.48 | **64.30** | **16.18** | **12.21** | 14.27 | **13.16** |

evaluation strategy proposed by [19], reporting P@IoU, R@IoU, and F1@IoU, which provide more straightforward evaluations across different IoU thresholds.

### 4.2 Common Object Detection

**Quantitative Comparisons.** From Table 1, we observe that the proposed router-based decision method consistently achieves the best performance across all evaluation metrics under the ***In-Dist.*** setting, demonstrating a superior balance between Precision and Recall. Specifically, our method attains an mAP@0.5 of **67.81%**, outperforming the second-best baseline (LLMdet) by more than **8%**, which highlights the effectiveness of the proposed visual token-wise cross-modal interaction for object detection. In contrast, simple unimodal detectors (RGB-only or IR-only) as well as late-fusion strategies based on NMS or pooling exhibit a pronounced Precision-Recall trade-off, suffering either from redundant predictions or missed detections, which leads to inferior F1 scores. Notably, even under the more challenging IoU = 0.95 evaluation criterion, our method remains superior, which further validates the effectiveness of the proposed Geometry-Aware loss in scenarios requiring precise boundary localization.

In the ***Out-of-Dist.*** setting (Table 2), the discrepancy between test scenes and training environments poses generalization challenges. Compared with the street-view scenes in MSRS, VTUAV contains UAV views with smaller, more crowded and frequently occluded pedestrians, making detection more difficult. Existing methods on the VTUAV dataset exhibit substantial performance degradation under complex backgrounds and distribution shifts, with both Precision and Recall dropping markedly. Despite these challenges, our method achieves the best performance on 6 out of 8 evaluation metrics in Table 2. Notably, it attains the highest F1@mIoU score, reflecting the most favorable trade-off

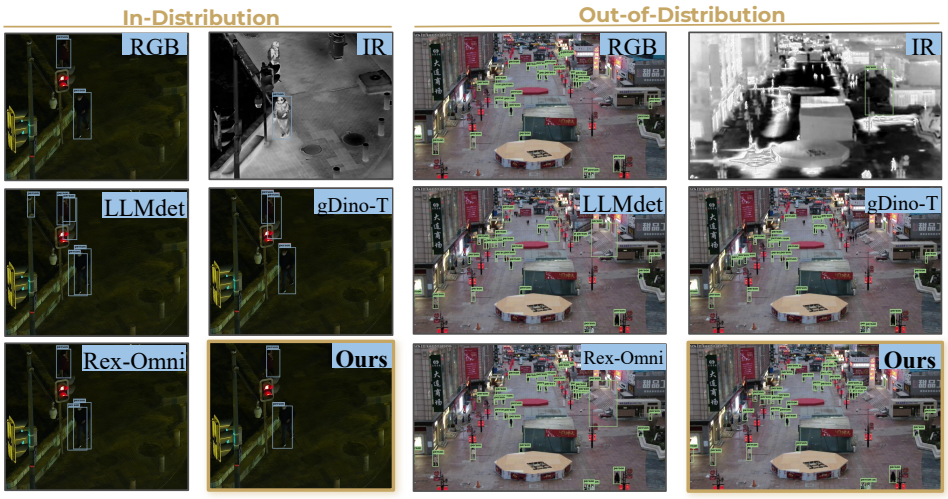



Figure 3: Images labeled RGB and IR show unimodal detection results of Rex-Omni. In contrast, images labeled LLMdet, gDino-T, and Rex-Omni use RGB-IR inputs with NMS-based box fusion. Due to space limitations, for these results we visualize the fused bounding boxes only on the RGB images. In *In-Dist.*, NMS-based fusion retains redundant overlapping boxes, while in *Out-of-Dist.* dense scene, other methods suffer from missed detections and IR-induced false positives. Our method consistently achieves accurate, non-redundant localization across both scenarios.

between Precision and Recall. These results demonstrate the strong generalization capability of the proposed method under distribution shifts.

**Qualitative Comparisons.** As shown in Figure 3, we conduct qualitative comparisons under both *In-Dist.* and *Out-of-Dist.* scenarios. In the *In-Dist.* scenario, although both RGB and IR detectors can correctly localize pedestrians, their independently predicted bounding boxes fail to meet the NMS suppression threshold, resulting in redundant detections. In the more challenging *Out-of-Dist.* setting, crowded scenes lead to frequent missed detections and reduced localization accuracy. For example, IR-induced false positives are not effectively suppressed by RGB cues, causing cross-modal error accumulation during post-hoc fusion. In contrast, our method abandons post-hoc fusion and instead establishes complementary cross-modal representations through visual token-wise interaction. This design enables precise and non-redundant localization in *In-Dist. scenarios*, while improving small-object detection in crowded *Out-of-Dist.* scenes.

### 4.3 Referring Object Detection

Referring object detection aims to recognize and localize objects based on natural language descriptions. Compared with category-level common object detection, this task requires models to simultaneously possess fine-grained language comprehension and reliable localization capability. In RGB-IR object detection, these requirements are further increased due to the need for effective visual cross-modal collaboration and semantic understanding. As shown in Table 3, compared to existing methods, our approach achieves the best results across all F1 thresholds. Notably, our method achieves these gains without introducing any modification to LLM decoder, relying solely on visual cross-modal alignment. This design choice preserves the original semantic understanding capabilities of LLM decoder, while multi-modal object detection performance is improved. Further visualizations can be found in Appendix H.

Table 3: Evaluation results on the proposed referring object detection benchmark. All methods except ours adopt NMS fusion.

| Type | Method | F1@0.5 | F1@0.95 | F1@mIoU |
|---|---|---|---|---|
| Open-set | gDino-T | 50.96 | 4.83 | 37.17 |
| MLLM | Qwen3-VL-4B | 76.52 | 4.06 | 49.99 |
| | Qwen3-VL-8B | 82.26 | 6.14 | 51.15 |
| | LLMdet | 47.86 | 5.50 | 28.52 |
| | Rex-Omni | 63.71 | 6.18 | 47.88 |
| | Ours | **86.18** | **7.61** | **58.15** |

### 4.4 Analysis and Ablation

**Sensitivity Analysis.** To analyze the combined effects of the scale coefficient $\lambda$ in the Geometry-Aware Loss on training stability and detection performance, we conducted comparative experiments under different $\lambda$ settings.

From the optimization perspective, as illustrated in Figure 4, the training loss curves exhibit notable differences across different values of $\lambda$. When $\lambda = 0$, our loss degrades into the standard Cross-Entropy loss. In this case, the training loss shows more oscillations and converges to a relatively high value. As $\lambda$ gradually increases, the loss curves become substantially smoother, accompanied by improved convergence speed and enhanced training stability. Furthermore, Table 4 reports the variance and maximum values of the training loss under different $\lambda$. As $\lambda$ increases from 0 to 0.1, the loss variance decreases steadily from 0.048 to 0.017, while the maximum loss value is also gradually reduced. These results confirm that our Geometry-Aware Loss alleviates gradient instability during optimization.

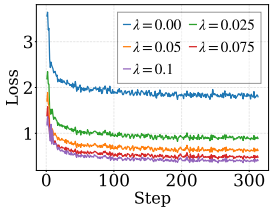



Figure 4: Training loss curves under different $\lambda$.

| $\lambda$ | Variance | Max |
|---|---|---|
| 0.00 | 0.048 | 3.644 |
| 0.025 | 0.031 | 2.347 |
| 0.05 | 0.022 | 1.892 |
| 0.075 | 0.018 | 1.580 |
| 0.1 | 0.017 | 1.362 |

Table 4: Variance and max statistics of loss for different $\lambda$.

Beyond training stability, the choice of $\lambda$ also influences detection performance. As reported in Table 5, increasing $\lambda$ from 0 to 0.05 leads to consistent improvements across all object scales. When $\lambda$ is further increased to 0.075 or 0.1, the performance gains tend to saturate, suggesting that overly broad semantic neighborhoods may also reduce precise localization. Therefore, considering both training stability and detection performance, we adopt $\lambda = 0.05$ as the default scale coefficient for Geometry-Aware Loss in all experiments, as it offers a favorable balance between optimization stability and localization precision.

Table 5: Common object detection performance under different $\lambda$ values. $\mathrm{AP}_s, \mathrm{AP}_m, \mathrm{AP}_l$ are evaluation metrics for objects with small (area $< 32^2$ pixels), medium ($32^2$ pixels $\leq$ area $\leq 96^2$ pixels), and large (area $> 96^2$ pixels) scales, respectively.

| $\lambda$ | 0.00 | 0.025 | 0.05 | 0.075 | 0.1 |
|---|---|---|---|---|---|
| $\mathrm{AP}_l$ | 59.50 | 60.80 | 62.60 | 62.10 | 56.90 |
| $\mathrm{AP}_m$ | 41.50 | 41.60 | 42.30 | 41.20 | 35.50 |
| $\mathrm{AP}_s$ | 13.70 | 14.00 | 15.80 | 15.50 | 11.10 |

Table 6: Evaluation of RGB-IR modality configurations and our proposed modules. The hyperparameters of LoRA can be seen in Appendix I.

| In-Distribution | | | | | MSRS (Out-of-Distribution) | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Modality | mAP@0.5 | P@mIoU | R@mIoU | F1@mIoU | Modality | mAP@0.5 | P@mIoU | R@mIoU | F1@mIoU |
| RGB | 56.22 | 50.82 | 38.52 | 43.83 | RGB | 73.81 | 48.31 | 56.33 | 52.02 |
| +LoRA | 59.44 | 50.59 | 41.90 | 45.84 | +LoRA | 81.24 | 48.80 | 58.37 | 53.16 |
| +CCA | 66.52 | 55.45 | 45.13 | 49.76 | +CCA | 92.80 | 51.35 | 75.31 | 61.06 |
| Infrared | 41.29 | 50.42 | 29.65 | 37.34 | Infrared | 77.14 | **56.43** | 64.29 | 60.11 |
| +LoRA | 46.10 | 47.41 | 33.30 | 39.12 | +LoRA | 90.90 | 48.73 | 74.01 | 58.77 |
| +CCA | 57.28 | 51.71 | 39.67 | 44.90 | +CCA | 92.28 | 52.54 | **77.37** | 62.58 |
| Router | **67.81** | **56.41** | **45.78** | **50.54** | Router | **92.82** | 56.01 | 75.48 | **64.30** |

**Effectiveness of the Proposed Modules.** We evaluate the effectiveness of the proposed modules by comparing them with LoRA [14], one of the most representative parameter-efficient fine-tuning (PEFT) approaches applied to LLM decoder. Table 6 reports an ablation study of the proposed modules. When using a single modality (RGB or infrared), detection performance is clearly limited. Introducing CCA consistently improves performance for both modalities, demonstrating its effectiveness in injecting complementary cross-modal information into unimodal representations. Building upon CCA, the proposed Visual Token-wise Router further enhances performance by enabling adaptive, token-level modality selection. Notably, our method outperforms LoRA-based fine-tuning across all evaluation metrics and achieves consistent gains under both *In-Dist.* and *Out-of-Dist.* settings. Controlled fusion comparisons and additional Rex-Omni fine-tuning results are provided in Appendices C.5 and C.6; extended loss ablations are reported in Appendix C.1. Further backbone and router analyses are provided in Appendix C.

**Efficiency Analysis.** We further analyze the parameter size and inference latency of each module, as summarized in Table 7. Although the proposed Router and CCA introduce additional parameters,

Table 7: Model Parameter Size and Inference Latency

| Module | Params (M) | Latency (s/img) | FPS |
|---|---|---|---|
| Router | 8.39 | 0.0002 | 5000 |
| CCA | 73.41 | 0.003 | 333 |
| LLM Decoder | 3085.93 | 0.085 | 12 |

their inference overhead is negligible compared to LLM decoder, indicating that the performance gains are achieved with minimal impact on inference efficiency. Full parameter counts, runtime measurements, and hardware settings are provided in Appendix I.1.

## 5 Conclusion

In this work, we demonstrate that MLLM can serve as a powerful backbone for multimodal object detection by combining adaptive visual cross-modal interaction with geometry-aware localization optimization. Extensive experiments demonstrate state-of-the-art performance and strong generalization across in-distribution, out-of-distribution, and referring object detection tasks. We hope this work can highlight the potential of MLLMs as a universal backbone for multimodal object detection and open new directions for detection-as-generation under more complex visual settings.

## Acknowledgments and Disclosure of Funding

This work was supported by the New Generation Artificial Intelligence–National Science and Technology Major Project (No. 2025ZD0122901) and the National Natural Science Foundation of China (No. 62622215 and 62572313). The authors declare no competing interests.

## References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[4] Bing Cao, Yiming Sun, Pengfei Zhu, and Qinghua Hu. Multi-modal gated mixture of local-to-global experts for dynamic image fusion. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 23555–23564, 2023.

[5] Chen Chen, Jiahao Qi, Xingyue Liu, Kangcheng Bin, Ruigang Fu, Xikun Hu, and Ping Zhong. Weakly misalignment-free adaptive feature alignment for uavs-based multimodal object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26836–26845, 2024.

[6] Ting Chen, Saurabh Saxena, Lala Li, David J Fleet, and Geoffrey Hinton. Pix2seq: A language modeling framework for object detection. In *International Conference on Learning Representations*, 2022.

[7] Yi-Ting Chen, Jinghao Shi, Zelin Ye, Christoph Mertz, Deva Ramanan, and Shu Kong. Multimodal object detection via probabilistic ensembling. In *European Conference on Computer Vision*, pages 139–158. Springer, 2022.

[8] Yiming Cui, Ziqing Yang, and Xin Yao. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*, 2023.

[9] Di Feng, Christian Haase-Schütz, Lars Rosenbaum, Heinz Hertlein, Claudius Glaeser, Fabian Timm, Werner Wiesbeck, and Klaus Dietmayer. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1341–1360, 2020.

[10] 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14987–14997, 2025.

[11] Himanshu Gupta, Oleksandr Kotlyar, Henrik Andreasson, and Achim J Lilienthal. Robust object detection in challenging weather conditions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7523–7532, 2024.

[12] Mingzhou He, Qingbo Wu, King Ngi Ngan, Feng Jiang, Fanman Meng, and Linfeng Xu. Misaligned rgb-infrared object detection via adaptive dual-discrepancy calibration. *Remote Sensing*, 15(19):4887, 2023.

[13] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[14] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1 (2):3, 2022.

[15] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[16] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017.

[17] Xinyu Jia, Chuang Zhu, Minzhen Li, Wenqi Tang, and Wenli Zhou. Llvip: A visible-infrared paired dataset for low-light vision. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3496–3504, 2021.

[18] Peiyuan Jiang, Daji Ergu, Fangyao Liu, Ying Cai, and Bo Ma. A review of yolo algorithm developments. *Procedia computer science*, 199:1066–1073, 2022.

[19] Qing Jiang, Junan Huo, Xingyu Chen, Yuda Xiong, Zhaoyang Zeng, Yihao Chen, Tianhe Ren, Junzhi Yu, and Lei Zhang. Detect anything via next point prediction. *arXiv preprint arXiv:2510.12798*, 2025.

[20] Gang Li, Yonggui Wang, Bin He, Tao Pang, and Mingke Gao. Low-light multimodal object detection: A survey. *Computer Science Review*, 58:100804, 2025.

[21] Jinyuan Liu, Xin Fan, Zhanbo Huang, Guanyao Wu, Risheng Liu, Wei Zhong, and Zhongxuan Luo. Target-aware dual adversarial learning and a multi-scenario multi-modality benchmark to fuse infrared and visible for object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5802–5811, 2022.

[22] 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*, pages 38–55. Springer, 2024.

[23] Michal Lukasik, Zhao Meng, Harikrishna Narasimhan, Yin-Wen Chang, Aditya Krishna Menon, Felix Yu, and Sanjiv Kumar. Better autoregressive regression with llms via regression-aware fine-tuning. In *The Thirteenth International Conference on Learning Representations*, 2025.

[24] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *International Conference on Learning Representations*, 2017.

[25] 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.

[26] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[27] Nandini Saini, Ashudeep Dubey, Debasis Das, and Chiranjoy Chattopadhyay. Advancing open-set object detection in remote sensing using multimodal large language model. In *Proceedings of the Winter Conference on Applications of Computer Vision*, pages 451–458, 2025.

[28] Yuxuan Sun, Yuanqin Meng, Qingbo Wang, Minghua Tang, Tao Shen, and Qingwang Wang. Visible and infrared image fusion for object detection: a survey. In *International Conference on Image, Vision and Intelligent Systems*, pages 236–248. Springer, 2023.

[29] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[30] Linfeng Tang, Jiteng Yuan, Hao Zhang, Xingyu Jiang, and Jiayi Ma. Piafusion: A progressive infrared and visible image fusion network based on illumination aware. *Information Fusion*, 83: 79–92, 2022.

[31] Xin Wu, Li Wang, Jian Guan, Haoyang Ji, Lianming Xu, Yingyan Hou, and Aiguo Fei. Dhanet: Dual-stream hierarchical interaction networks for multimodal drone object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 2025.

[32] Jonas Zausinger, Lars Pennig, Anamarija Kozina, Sean Sdahl, Julian Sikora, Adrian Dendorfer, Timofey Kuznetsov, Mohamad Hagog, Nina Wiedemann, Kacper Chlodny, et al. Regress, don't guess–a regression-like loss on number tokens for language models. *arXiv preprint arXiv:2411.02083*, 2024.

[33] Heng Zhang, Elisa Fromont, Sébastien Lefevre, and Bruno Avignon. Multispectral fusion for object detection with cyclic fuse-and-refine blocks. In *2020 IEEE International conference on image processing (ICIP)*, pages 276–280. IEEE, 2020.

[34] Jiaqing Zhang, Jie Lei, Weiying Xie, Zhenman Fang, Yunsong Li, and Qian Du. Superyolo: Super resolution assisted object detection in multimodal remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–15, 2023.

[35] Pengyu Zhang, Jie Zhao, Dong Wang, Huchuan Lu, and Xiang Ruan. Visible-thermal uav tracking: A large-scale benchmark and new baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8886–8895, 2022.

[36] Xue Zhang, Si-Yuan Cao, Fang Wang, Runmin Zhang, Zhe Wu, Xiaohan Zhang, Xiaokai Bai, and Hui-Liang Shen. Rethinking early-fusion strategies for improved multispectral object detection. *IEEE Transactions on Intelligent Vehicles*, 2024.

[37] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11): 3212–3232, 2019.

## Appendix Contents


**A Limitations** **14**

**B Broader Impacts** **14**

**C Further Discussion** **14**

C.1 Comparison with Label Smoothing . . . . . 14

C.2 Generalizability to other MLLM Backbone . . . . . 15

C.3 Regarding the effect of unfreezing the IR encoder . . . . . 15

C.4 Deeper Analysis of the Router . . . . . 16

C.5 Controlled Fusion Comparison . . . . . 17

C.6 Additional Rex-Omni Fine-Tuning . . . . . 17

**D Proof of Theorem 1** **18**

**E Multimodal Bounding Box Fusion** **19**

**F More Details about Datasets** **19**

F.1 Common Object Detection Datasets . . . . . 19

F.2 Annotations for Referring Object Detection . . . . . 20

**G Prompt for Object Detection** **21**

**H More Visualizations** **21**

**I Implementation Details** **21**

I.1 Model Size and Efficiency . . . . . 22

# A Limitations

One limitation of our method is the inference cost inherited from the LLM decoder. Although the proposed CCA and router modules add negligible overhead, the full pipeline still requires 8 NVIDIA H100 GPUs with FlashAttention-2 to achieve approximately 10 FPS. The router reduces redundant decoding by converting the two RGB-IR branches into a single routed visual token sequence before entering the LLM decoder, preserving the token count of a single-image input. Nevertheless, the LLM decoder remains the dominant computational bottleneck. Therefore, our method is better suited to server-side or offline high-compute scenarios where sub-second latency is acceptable and accuracy is prioritized. Future work will explore model compression, quantization, and more efficient multimodal decoding to enable deployment under stricter resource constraints. We also acknowledge that the Geometry-Aware Loss is closely tied to atomic coordinate tokenization and requires compatible coordinate modeling mechanisms for extension. We expect this paradigm to gain broader adoption in future MLLMs. Freezing the RGB-pretrained visual encoder may also limit infrared-specific adaptation; Appendix C.3 examines this trade-off.

# B Broader Impacts

This work advances multi-modal object detection by leveraging RGB and infrared fusion within the MLLM framework to improve robustness and generalization under challenging conditions. Such capabilities may have positive impacts in autonomous driving, robotics, UAV perception, remote sensing, and safety-critical monitoring, where reliable object localization is important for downstream decision making. However, the same capability may also raise societal concerns. RGB-IR perception can operate in low-light or nighttime environments, which may increase the risk of privacy-invasive surveillance or other sensitive uses if deployed without proper safeguards. In addition, the computational cost of MLLM-based detection may also increase energy consumption and limit accessibility. We therefore recommend responsible deployment with human oversight, privacy protection, and careful application-specific validation.

# C Further Discussion

## C.1 Comparison with Label Smoothing

We further compare the proposed Geometry-Aware Loss with standard label smoothing [29] to clarify their differences. Although both methods relax overly strict one-hot supervision, they differ in both design motivation and supervision structure.

Standard label smoothing is primarily designed for classification and uniformly distributes a fixed amount of probability mass to non-target classes. In contrast, our Geometry-Aware Loss is designed for discretized coordinate prediction in detection-as-generation models, where coordinate tokens have an inherent spatial order. Instead of smoothing over all non-target tokens, our loss aggregates probability only within a semantic $\delta$-neighborhood of the ground-truth coordinate token. Moreover, the neighborhood size is dynamically determined by the object scale, providing larger tolerance for larger objects and stricter supervision for smaller objects.

Table 8: Comparison between label smoothing and the proposed Geometry-Aware Loss. $\epsilon$ is the label smoothing coefficient.

| Method | $\mathrm{AP}_l$ | $\mathrm{AP}_m$ | $\mathrm{AP}_s$ |
|---|---|---|---|
| CE baseline | 59.5 | 41.5 | 13.7 |
| Label Smoothing ($\epsilon = 0.05$) | 59.4 | 41.5 | 14.1 |
| Label Smoothing ($\epsilon = 0.10$) | 61.1 | 40.6 | 12.2 |
| Label Smoothing ($\epsilon = 0.20$) | 61.9 | 40.8 | 12.0 |
| Geometry-Aware Loss (Ours) | **62.6** | **42.3** | **15.8** |

We also conduct an empirical comparison with label smoothing. For a fair comparison, label smoothing is applied only to coordinate tokens, while non-coordinate tokens remain supervised by

standard Cross-Entropy. All other training and evaluation settings are kept identical. As shown in Table 8, label smoothing slightly improves large-object AP under some settings, but it degrades medium- and small-object performance as the smoothing coefficient increases. By contrast, the proposed Geometry-Aware Loss achieves the best performance across all object scales, demonstrating that scale-adaptive neighborhood supervision is better suited for discretized coordinate prediction.

**Extended loss ablation.** We further analyze atomic coordinate tokenization, CE, label smoothing, and Geometry-Aware Loss (GAL) variants. GAL Fix (2/4/8) uses fixed neighborhoods with 2/4/8 neighboring atoms, whereas Adaptive uses scale-dependent neighborhoods. Table 9 reports the full detection metrics. Adaptive GAL achieves the highest mAP@0.5 and F1 at IoU 0.5, IoU 0.95, and averaged IoU thresholds.

Table 9: Extended loss ablation on the In-Dist. benchmark. Best results are bold.

| | | IoU 0.5 | | | IoU 0.95 | | | mIoU | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Method | mAP@0.5 | P | R | F1 | P | R | F1 | P | R | F1 |
| CE | 65.07 | 75.82 | 70.49 | 73.05 | 26.34 | 7.12 | 11.22 | 56.07 | 44.87 | 49.85 |
| LS (0.05) | 65.32 | 76.61 | 71.76 | 74.11 | **33.51** | 7.32 | 12.01 | 54.01 | 44.10 | 48.55 |
| LS (0.10) | 64.62 | 75.48 | 71.26 | 73.31 | 27.71 | 5.55 | 9.24 | 55.49 | 43.30 | 48.64 |
| LS (0.20) | 61.95 | 74.69 | 70.24 | 72.39 | 18.37 | 6.44 | 9.53 | 52.70 | 42.55 | 47.08 |
| GAL Fix (2) | 66.00 | 76.28 | 73.17 | 74.70 | 22.61 | 6.48 | 10.07 | 56.26 | 44.48 | 49.68 |
| GAL Fix (4) | 66.62 | 73.59 | 74.05 | 73.82 | 27.82 | 5.94 | 9.79 | 53.14 | 44.61 | 48.50 |
| GAL Fix (8) | 64.12 | 72.09 | 70.90 | 71.49 | 19.04 | 5.99 | 9.11 | 50.61 | 41.31 | 45.49 |
| GAL Adaptive (ours) | **67.81** | **79.53** | **74.48** | **76.92** | 30.73 | **7.81** | **12.46** | **56.41** | **45.78** | **50.54** |

### C.2 Generalizability to other MLLM Backbone

To further validate the generalizability of our framework, we conduct additional experiments on a different MLLM backbone (Qwen3-VL-4B [2]) without using the Geometry-Aware Loss. The results are summarized in Table 10:

Table 10: Generalizability evaluation on Qwen3-VL-4B without Geometry-Aware Loss. The improvements brought by CCA and the router indicate that the proposed framework can transfer to other MLLM backbone.

| **Modality** | **mAP@0.5** | **P@mIoU** | **R@mIoU** | **F1@mIoU** |
|---|---|---|---|---|
| RGB | 30.15 | 44.89 | 22.32 | 29.82 |
| +CCA | 35.62 | 49.91 | 26.00 | 34.19 |
| IR | 29.64 | 49.43 | 21.50 | 29.96 |
| +CCA | 55.82 | 48.74 | 43.52 | 45.98 |
| Router | 56.66 | 49.60 | 43.71 | 46.47 |

These results demonstrate that both the CCA and router consistently improve performance, indicating that these two modules are not tied to a specific backbone. We also acknowledge that the Geometry-Aware Loss is closely tied to atomic coordinate tokenization and requires compatible coordinate modeling mechanisms for extension. We expect this paradigm to gain broader adoption in future MLLMs.

### C.3 Regarding the effect of unfreezing the IR encoder

Since the visual encoder in Qwen2.5-VL is primarily pretrained on RGB images, freezing the backbone can indeed introduce a bottleneck for IR feature adaptation before the CCA module operates. To investigate this, we conduct an additional experiment with the IR encoder unfrozen. The results in Table 11 show that unfreezing the IR encoder and combining it with CCA further improves performance, with the gain being particularly pronounced in the IR + CCA setting (IR+CCA denotes that IR tokens are enhanced according to Eq. 3, while RGB+CCA denotes that RGB tokens are

enhanced according to Eq. 2). This suggests that unfreezing the IR encoder indeed enhances the model's adaptation to the thermal infrared domain.

Table 11: Performance comparison across modalities on ID dataset.

| **Modality** | **P@mIoU** | **R@mIoU** | **F1@mIoU** |
|---|---|---|---|
| RGB | 50.82 | 38.52 | 43.83 |
| w/ CCA (freeze IR) | 55.45 | 45.13 | 49.76 |
| w/ CCA (unfreeze IR) | 56.29 | 45.65 | 50.41 |
| IR | 50.42 | 29.65 | 37.34 |
| w/ CCA (freeze IR) | 51.71 | 39.67 | 44.90 |
| w/ CCA (unfreeze IR) | 59.90 | 42.48 | 49.71 |
| Router (freeze IR) | 56.41 | 45.78 | 50.54 |
| Router (unfreeze IR) | 60.13 | 42.49 | 49.79 |

At the same time, this modification substantially increases the training cost. With the IR encoder frozen, the total training time is about 71 minutes, whereas adapting the IR encoder using LoRA increases the total training time to about 223 minutes when using 8 H100 GPUs. In addition, we observe that after unfreezing the IR encoder, the router tends to preserve more IR-branch information, which shifts the precision-recall trade-off toward the IR + CCA side. Overall, these results indicate that freezing the backbone does limit thermal feature adaptation to some extent, while still providing a more favorable trade-off between performance and training efficiency.

**Severe-weather evaluation.** To assess infrared use when RGB appearance cues are degraded, we query each RGB image in the inference set with: "Does this image show very severe weather conditions (e.g., extremely heavy fog causing very poor visibility)? A. Yes, it shows very severe weather with poor visibility. B. No, it does not show very severe weather." This identifies 26 samples. As shown in Table 12, our performance decreases from 67.81 to 62.22 mAP@0.5 under severe

Table 12: Detection performance under severe weather.

| Method | mAP@0.5 |
|---|---|
| MoE-Fusion | 51.58 |
| Ours | 67.81 |
| Ours (severe weather) | 62.22 |

weather but remains above MoE-Fusion (51.58), suggesting that the framework preserves useful infrared information when RGB cues deteriorate. Possible explanations include semantic priors in the LLM decoder, low-level structural features retained by the visual encoder, and exposure to infrared-like patterns during pretraining. These explanations remain hypotheses; efficient infrared-specific adaptation is a direction for future work.

### C.4 Deeper Analysis of the Router

**Hard versus soft routing.** We first compare our default hard-routing strategy with a soft-routing alternative. In soft routing, the two enhanced modality tokens are fused by weighted averaging rather than discrete selection. As shown in Table 13, hard routing achieves better overall performance than soft routing. Although the margin is not large, hard routing encourages the router to make explicit modality choices, making its behavior easier to interpret at the token level.

Table 13: Hard routing versus soft routing on the ID benchmark.

| Method | F1@IoU 0.95 | P@IoU 0.95 | R@IoU 0.95 | mAP@0.5 |
|---|---|---|---|---|
| Soft Routing | 11.17 | 25.72 | 7.14 | 67.73 |
| Hard Routing | 12.46 | 30.73 | 7.81 | 67.81 |

**Visualization of routing decisions.** To further analyze what our router learns after training, we visualize token-level selections of our router. The visualization in Figure 5 shows that routing decisions vary across spatial locations. Specifically, in regions where thermal contrast is more informative, the router tends to select IR-enhanced tokens; in regions where RGB provides clearer texture, RGB-enhanced tokens are more frequently selected.

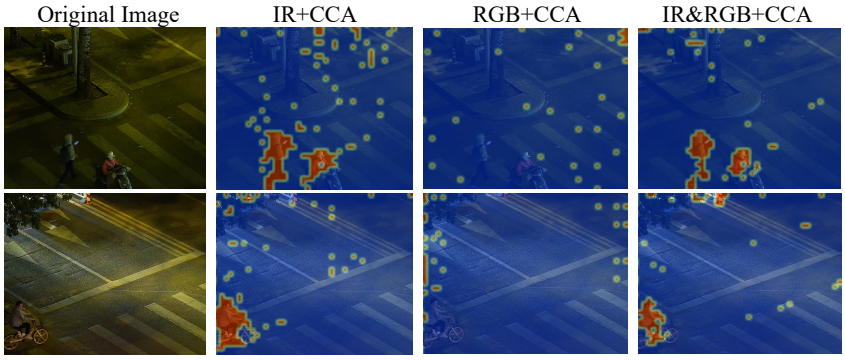



Figure 5: Visualization of token-level selections of our router.

### C.5 Controlled Fusion Comparison

We compare token concatenation, CCA-based late fusion, and token-wise routing under the same CCA training configuration. Concatenation forms $\mathbf{V}_{\text{cat}} = [\mathbf{V}_{\text{rgb}}, \mathbf{V}_{\text{ir}}] \in \mathbb{R}^{d \times 2N}$. Since the backbone does not support multi-image detection, a learnable projection $\widetilde{\mathbf{V}} = \mathbf{V}_{\text{cat}}\mathbf{W}_O$, with $\mathbf{W}_O \in \mathbb{R}^{2N \times N}$, restores the original token length. RGB-init and IR-init initially preserve the corresponding modality and gradually learn cross-modal fusion. Table 14 shows that Router outperforms both concatenation and CCA-based late fusion.

Table 14: Controlled fusion comparison on the In-Dist. benchmark.

| Method | mAP@0.5 | F1@IoU 0.5 | F1@IoU 0.95 | F1@mIoU |
|---|---|---|---|---|
| Concatenation (RGB-init) | 57.94 | 69.49 | 6.12 | 42.26 |
| CCA + RGB | 66.52 | 74.49 | 11.22 | 49.76 |
| Concatenation (IR-init) | 44.67 | 58.29 | 9.91 | 39.28 |
| CCA + IR | 57.28 | 62.72 | 8.36 | 44.90 |
| CCA + NMS | 64.88 | 70.26 | 11.49 | 48.40 |
| CCA + Pooling | 64.01 | 70.24 | 11.41 | 46.96 |
| Router | **67.81** | **76.92** | **12.46** | **50.54** |

### C.6 Additional Rex-Omni Fine-Tuning

To control for task-specific training, we fine-tune Rex-Omni on the same training data for four epochs using CE, applying LoRA separately to the LLM decoder or the ViT encoder. These additional experiments use $r = 64$ and $\alpha = 128$, distinct from the $r = 8$, $\alpha = 16$ configuration in Table 6. Table 15 shows that our method outperforms both variants across all reported metrics. Training details are provided in Appendix I.

Table 15: Rex-Omni fine-tuning on the same In-Dist. training data.

| Method | Decision | mAP@0.5 | F1@IoU 0.5 | F1@IoU 0.95 | F1@mIoU |
|---|---|---|---|---|---|
| LLM decoder + LoRA | RGB | 61.76 | 70.52 | 9.62 | 46.04 |
| LLM decoder + LoRA | IR | 47.60 | 57.37 | 10.44 | 40.65 |
| LLM decoder + LoRA | NMS | 65.40 | 62.69 | 11.00 | 44.22 |
| LLM decoder + LoRA | Pooling | 63.58 | 62.77 | 11.35 | 44.33 |
| ViT + LoRA | RGB | 60.21 | 71.78 | 9.13 | 45.24 |
| ViT + LoRA | IR | 44.44 | 58.32 | 8.65 | 38.86 |
| ViT + LoRA | NMS | 61.46 | 67.08 | 9.16 | 44.73 |
| ViT + LoRA | Pooling | 60.28 | 67.13 | 10.02 | 45.08 |
| Ours | Router | **67.81** | **76.92** | **12.46** | **50.54** |

# D Proof of Theorem 1

**Theorem 1** (Convergence Analysis)**.** *For any step $t$ where the ground-truth token is a coordinate, if $\mathcal{F}_t(\epsilon, \delta)$ is the family of $(\epsilon, \delta)$-confident distributions, then for the Geometry-Aware loss $\mathcal{L}_{geo}^{(t)}$ and Cross-Entropy loss $\mathcal{L}_{ce}^{(t)}$, the following holds:*

$$\begin{aligned} \sup_{\mathbf{p}\in\mathcal{F}_t(\epsilon,\delta)} \lim_{\epsilon\to 0} \mathcal{L}_{geo}^{(t)}(\mathbf{p}, y_t^*) = 0, \\ \sup_{\mathbf{p}\in\mathcal{F}_t(\epsilon,\delta)} \lim_{\epsilon\to 0} \mathcal{L}_{ce}^{(t)}(\mathbf{p}, y_t^*) = \infty. \end{aligned} \tag{12}$$

*Proof.* **1. Proof of Uniform Boundedness for $\mathcal{L}_{\text{geo}}^{(t)}$.**

By the definition of $\mathcal{F}_t(\epsilon, \delta)$, any distribution $\mathbf{p} \in \mathcal{F}_t(\epsilon, \delta)$ satisfies

$$\mathbb{P}(\hat{y}_t \in \mathcal{N}_\delta^{(t)}) = \sum_{v\in\mathcal{N}_\delta^{(t)}} p_v^{(t)} \geq 1 - \epsilon. \tag{13}$$

Since $f(x) = -\log x$ is a monotonically decreasing function for $x \in (0, 1]$, we have:

$$\mathcal{L}_{\text{geo}}^{(t)}(\mathbf{p}, y_t^*) = -\log \sum_{v\in\mathcal{N}_\delta^{(t)}} p_v^{(t)} \leq -\log(1-\epsilon). \tag{14}$$

As $\epsilon \to 0$, we apply the Taylor expansion:

$$-\log(1-\epsilon) = \epsilon + \frac{\epsilon^2}{2} + \mathcal{O}(\epsilon^3). \tag{15}$$

Thus,

$$\sup_{\mathbf{p}\in\mathcal{F}_t(\epsilon,\delta)} \lim_{\epsilon\to 0} \mathcal{L}_{\text{geo}}^{(t)}(\mathbf{p}, y_t^*) = 0. \tag{16}$$

**2. Proof of Divergence for $\mathcal{L}_{\text{ce}}^{(t)}$.**

To show $\sup_{\mathbf{p}\in\mathcal{F}_t(\epsilon,\delta)} \lim_{\epsilon\to 0} \mathcal{L}_{\text{ce}}^{(t)}(\mathbf{p}) = \infty$, consider a neighborhood $\mathcal{N}_\delta^{(t)}$ that includes at least one neighbor $y' \neq y_t^*$. Let $\mathcal{V}$ be the total token space. For any $\gamma \in (0, 1-\epsilon)$, there exists a distribution $\mathbf{p} \in \mathcal{F}_t(\epsilon, \delta)$ satisfying:

$$p_v^{(t)} = \begin{cases} \gamma & \text{if } v = y_t^* \\ 1-\epsilon-\gamma & \text{if } v = y' \\ \frac{\epsilon}{|\mathcal{V}\setminus\mathcal{N}_\delta^{(t)}|} & \text{if } v \notin \mathcal{N}_\delta^{(t)} \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

By summing all components, we verify the total probability mass: $\sum_{v\in\mathcal{V}} p_v^{(t)} = \gamma + (1-\epsilon-\gamma) + \epsilon = 1$. The probability mass assigned to the semantic $\delta$-neighborhood is $\mathbb{P}(\hat{y}_t \in \mathcal{N}_\delta^{(t)}) \geq p_{y_t^*}^{(t)} + p_{y'}^{(t)} = 1-\epsilon$, which satisfies the condition for $\mathbf{p} \in \mathcal{F}_t(\epsilon, \delta)$. However, the Cross-Entropy loss is:

$$\mathcal{L}_{\text{ce}}^{(t)}(\mathbf{p}, y_t^*) = -\log p_{y_t^*}^{(t)} = -\log\gamma. \tag{18}$$

Taking the limit as $\gamma \to 0^+$, we obtain:

$$\lim_{\gamma \to 0^+} \mathcal{L}_{\text{ce}}^{(t)}(\mathbf{p}, y_t^*) = +\infty. \tag{19}$$

Therefore, the standard Cross-Entropy loss is asymptotically unbounded on $\mathbf{p} \in \mathcal{F}_t(\epsilon, \delta)$ despite high semantic confidence. □

# E Multimodal Bounding Box Fusion

We consider two multimodal bounding box fusion strategies. The first, shown in Algorithm 1, follows the multimodal *Non-Maximum Suppression* (**NMS**) paradigm [7]. All detections are ranked by confidence score, and the highest-scoring detection is iteratively selected while suppressing overlapping detections above an IoU threshold. This approach retains the most confident detection within each spatial neighborhood. The second strategy called *Average Box Pooling* (**Pooling**), detailed in Algorithm 2, replaces suppression with aggregation. For a given random seed bounding box, all detections with sufficient overlap are grouped together, and their bounding box coordinates are averaged to form a fused prediction. Pooling preserves contributions from all modalities, making it particularly effective in scenarios where confidence scores are unavailable or unreliable.

**Algorithm 1** Multimodal Fusion by Non-Maximum Suppression (NMS)

**Input:** detections $\mathcal{D}$ from multiple modalities; IoU threshold $\tau$ (e.g., 0.75)
**Each detection:** $d = (\mathbf{b}, s) \in \mathcal{D}$ with bounding box $\mathbf{b}$ and confidence score $s$
**Output:** fused detections $\mathcal{F}$
$\mathcal{F} \leftarrow \emptyset$ ▷ init fused set
**while** $\mathcal{D} \neq \emptyset$ **do**
  $d_{\text{best}} \leftarrow \arg\max_{d \in \mathcal{D}} s$ ▷ max confidence
  $\mathcal{T} \leftarrow \{d \in \mathcal{D} \mid \text{IoU}(d, d_{\text{best}}) > \tau\}$ ▷ overlaps
  $\mathcal{F} \leftarrow \mathcal{F} \cup \{d_{\text{best}}\}$ ▷ keep best
  $\mathcal{D} \leftarrow \mathcal{D} \setminus \mathcal{T}$ ▷ drop suppressed
**end while**
**return** $\mathcal{F}$

**Algorithm 2** Multimodal Fusion by Average Box Pooling

**Input:** detections $\mathcal{D}$ from multiple modalities; IoU threshold $\tau$ (e.g., 0.75)
**Each detection:** $d = \mathbf{b} \in \mathcal{D}$ with $\mathbf{b} = [x_1, y_1, x_2, y_2]$
**Output:** fused detections $\mathcal{F}$
$\mathcal{F} \leftarrow \emptyset$ ▷ init fused set
**while** $\mathcal{D} \neq \emptyset$ **do**
  Sample $\mathbf{b}_{\text{seed}} \sim \mathcal{D}$ as the seed box ▷ random seed
  $\mathcal{C} \leftarrow \{\mathbf{b} \in \mathcal{D} \mid \text{IoU}(\mathbf{b}, \mathbf{b}_{\text{seed}}) > \tau\} \cup \{\mathbf{b}_{\text{seed}}\}$
  $N \leftarrow |\mathcal{C}|$ ▷ box count for pooling
  $\mathbf{b}_{\text{avg}} \leftarrow \frac{1}{N} \sum_{\mathbf{b}_i \in \mathcal{C}} \mathbf{b}_i$ ▷ mean box coordinates
  $\mathcal{F} \leftarrow \mathcal{F} \cup \{\mathbf{b}_{\text{avg}}\}$ ▷ add fused box
  $\mathcal{D} \leftarrow \mathcal{D} \setminus \mathcal{C}$ ▷ remove selected boxes
**end while**
**return** $\mathcal{F}$

# F More Details about Datasets

## F.1 Common Object Detection Datasets

As shown in Table 16, we evaluate in-distribution (In-Dist.) performance on three widely used paired infrared-visible benchmarks, following the data splits of Cao et al. [4]. $M^3FD$ [21] contains 4,200 aligned infrared-visible image pairs captured by onboard cameras, annotated with six categories (Person, Car, Bus, Motorcycle, Lamp, Truck); we use 3,900 pairs for training and 300 pairs for

evaluation. FLIR-Aligned [33] provides 5,142 aligned infrared-visible pairs collected by vehicle-mounted sensors with four categories (Bicycle, Car, Person, Dog); 4,129 pairs are used for training and 1,013 pairs for evaluation. LLVIP [17] includes 15,488 aligned infrared-visible pairs captured by surveillance cameras in diverse street scenes, focusing on a single category (Person); 12,025 pairs are used for training and 3,463 pairs for evaluation. In total, the In-Dist. benchmark comprises 20,054 training pairs and 4,776 evaluation pairs. Notably, throughout all experiments, we train only on the In-Dist. training data. We conduct Out-of-Distribution (Out-of-Dist.) evaluation on two public datasets (MSRS [30], VTUAV [35]), comprising 5,458 samples in total and covering three human-related categories (Person, Rider, Crowd).

Table 16: In-distribution (In-Dist.) benchmark and data split.

| Dataset | #Pairs (Total) | Categories | #Train | #Eval |
|---|---|---|---|---|
| $M^3FD$ | 4,200 | Person, Car, Bus, Motorcycle, Lamp, Truck | 3,900 | 300 |
| FLIR-Aligned | 5,142 | Bicycle, Car, Person, Dog | 4,129 | 1,013 |
| LLVIP | 15,488 | Person | 12,025 | 3,463 |
| Total | 24,830 | | 20,054 | 4,776 |

Note: Throughout all experiments, we train only on the In-Dist. training data.

### F.2 Annotations for Referring Object Detection

As shown in Figure 6, we design a fully automated multi-agent framework for generating referring expressions on an RGB-IR multimodal dataset, aiming to ensure high annotation accuracy while eliminating human bias. The entire pipeline operates without any manual filtering or refinement and achieves quality assurance through four stages: generation, detection, refinement, and validation. Given paired RGB and infrared images along with COCO-format annotations, the generator agent (GPT-5) first produces initial referring expressions based on multimodal visual cues and object locations. The detector agent (GPT-5) then evaluates the expressions by zooming into the target region and assessing their correctness, clarity, and ambiguity. Expressions that pass this quality check are directly accepted as final outputs. Those that fail are accompanied by explicit error types and refinement suggestions and forwarded to the refiner agent. The refiner agent (GPT-5), working without visual input, refines the expressions based solely on feedback from the detector agent. Finally, the validator agent (Gemini3-Flash) performs a final verification step, and only approved expressions are retained. Based on the validation sets of $M^3FD$ [21], FLIR-Aligned [33], and LLVIP [17], we construct a high-quality referring expression dataset containing 3,429 high-quality expressions, which is used to benchmark models for referring object detection.

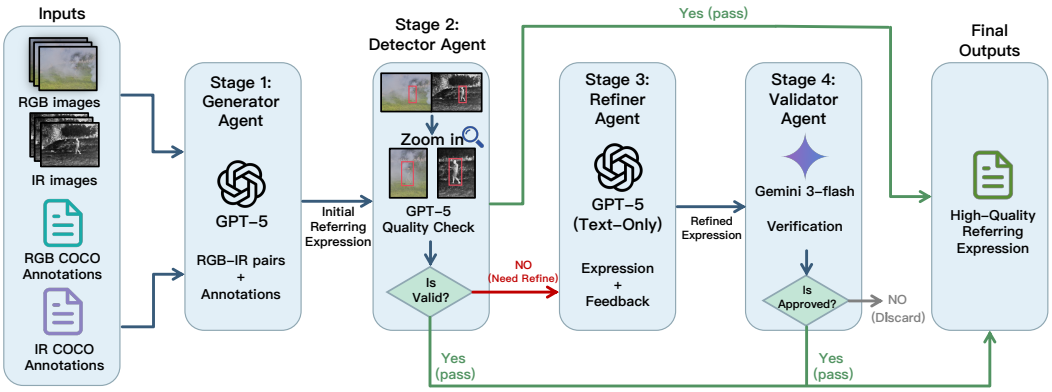



Figure 6: Overview of the proposed fully automated multi-agent pipeline for RGB–IR referring expression generation. The generator agent produces initial expressions from multimodal inputs, which are evaluated by the detector Agent via local zoom-in quality checks. Invalid expressions are refined by the text-only refiner agent based on detector feedback and finally filtered by the validator agent to ensure accuracy and unambiguity.

## G Prompt for Object Detection

The object detection task is described using a unified prompt template for gDino-T and MLLM-based object detection except Qwen3-VL. As shown below, <Objects> denotes the target objects to be detected in the image. In common object detection setting, <Objects> may consist of multiple object categories, such as Bus, Lamp, and Motorcycle. In referring object detection setting, <Objects> is typically a natural language referring expression, for example, "the blue car in the parking lot."

**An template of prompt for object detection**

Detect <Objects>. Output the bounding box coordinates in [x1, y1, x2, y2] format.

For our method and Rex-Omni, the output follows a unified generative format. Specifically, for each detected instance, the semantic reference is enclosed by <|object_ref_start|> and <|object_ref_end|>, which represents either the object category name or a referring expression. The predicted bounding box is then specified between <|box_start|>and <|box_end|> in the [x1, y1, x2, y2] format, where (x1, y1) and (x2, y2) correspond to the top-left and bottom-right corners of the object, respectively. When multiple instances of the same category are present, the model outputs them in a left-to-right and top-to-bottom order, ensuring a consistent alignment between semantic references and spatial locations.

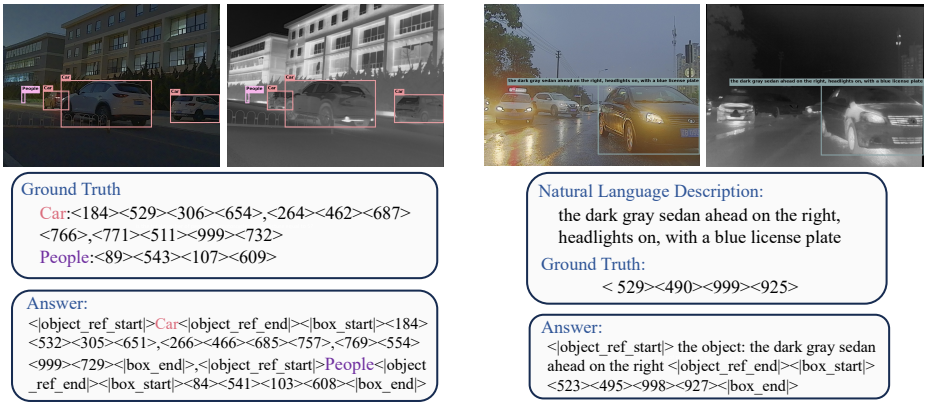
In contrast, gDINO-T and LLMdet adopt non-generative outputs that are similar to those of traditional object detectors. Their predictions inherently follow predefined structured formats, and therefore no additional output constraints are imposed.

**An template of output format for object detection**

<|object_ref_start|>{Object}<|object_ref_end|>
<|box_start|><x1><y1><x2><y2>,... <|box_end|>, ... (other objects if any)

For Qwen3-VL, the output format is predefined during training by the official implementation. We directly adopt the following format: "Detect [Object 1], [Object 2], ... in the image and return their locations. Output: [{'bbox_2d': [x1, y1, x2, y2], 'label': 'class_name'},...]"

## H More Visualizations




(a) Visualization of common object detection

(b) Visualization of referring object detection

Figure 7: Examples of visualizations produced by our method on common object detection and referring object detection tasks. The bounding boxes shown in the images represent the prediction results of our method.

## I Implementation Details

**Baseline protocols.** All conventional closed-set detectors are trained on the same dataset as our method, following the official MoE-Fusion training and evaluation settings. MoE-Fusion uses a box

confidence threshold of 0.05. Other pretrained baselines are evaluated using released checkpoints and their original evaluation protocols, prompts, and default inference settings unless stated otherwise.

**LoRA configurations.** For Table 6, LoRA fine-tuning is performed with lora_r = 8, lora_alpha = 16, and lora_dropout = 0.0, applied to the q_proj, k_proj, v_proj, and o_proj modules. The additional Rex-Omni experiments in Table 15 instead use CE for four epochs with a learning rate of $10^{-4}$, $r = 64$, and $\alpha = 128$, adapting either the ViT encoder or the LLM decoder.

**Inference settings.** For Grounding-DINO, a box_threshold of 0.3 is used to remove low-confidence bounding box predictions, while LLMdet adopts a confidence threshold of 0.25. For generative MLLMs (Qwen3-VL, Rex-Omni, and ours), the bounding box confidence is computed as the average of the semantic $\delta$-neighborhood probabilities of the four corresponding coordinates. All generative MLLMs perform greedy decoding for inference and apply a repetition penalty of 1.05 to avoid repetitive outputs. For our method, generated boxes are retained without confidence-based filtering. For NMS and Pooling fusion strategies, we set the IoU threshold $\tau$ to 0.75.

### I.1 Model Size and Efficiency

Our model uses the Qwen2.5-VL-3B architecture initialized from Rex-Omni. Experiments run on a server with eight NVIDIA H100 GPUs and two Intel Xeon Platinum 8558 CPUs (96 physical cores and 192 threads in total). With a global inference batch size of 288, input resolution of $1280 \times 1024$, and maximum output length of 1024 tokens, peak allocated GPU memory is 34.758 GiB on a single GPU. Table 17 reports component sizes and inference costs. The visual encoder comprises a 631.99M-parameter ViT and a 36.71M-parameter merger. The LLM decoder dominates runtime, while CCA and Router add little overhead.

Table 17: Full model size and inference efficiency.

| Module | Params (M) | Latency (s/img) | FPS |
|---|---|---|---|
| Visual Encoding | 668.70 | 0.0198 | 50.51 |
| Router | 8.39 | 0.0002 | 5000 |
| CCA | 73.41 | 0.003 | 333 |
| LLM Decoder | 3085.93 | 0.085 | 11.76 |
| Total | 3836.43 | 0.108 | 9.26 |

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Introduction state the proposed CCA, Visual Token-wise Router, and Geometry-Aware Loss, and the reported claims are supported by the experimental results in §3.2–§3.5 and §1.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations in Appendix A, including the inference cost inherited from the LLM decoder and the dependence of Geometry-Aware Loss on compatible atomic coordinate tokenization.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The theoretical statement and assumptions for the convergence analysis are given in §3.5, and the complete proof is provided in Appendix D.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper describes datasets, splits, baselines, prompts, training hyperparameters, decoding settings, and evaluation metrics in the Experiments section.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: This paper uses public RGB-IR benchmarks and provides reproducibility details. The code repository is `https://github.com/zxk1212/rgb_ir`; release preparation is ongoing. We will release the relevant code and model weights upon publication.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The Experiments section and Appendix I specify dataset splits, optimizer, batch size, learning rates, training stages, thresholds, decoding settings, and evaluation metrics.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Due to the high computational cost of the experiments, the current submission does not explicitly report statistical significance tests.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports GPU and CPU specifications, inference resolution and batch size, peak allocated GPU memory, and component-level latency in Appendix I.1, with training time comparisons in Appendix C.3.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work uses public datasets and automated evaluation/data-generation procedures, and we are not aware of any deviation from the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: The paper discusses the potential positive societal impacts of the work, such as the ability to detect objects in images and videos, and the potential negative societal impacts of the work, such as the potential for misuse of the technology.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release high-risk generative models, scraped datasets, or other assets that require special misuse safeguards.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper credits the datasets, baselines, and model backbones through citations, but it does not explicitly list the licenses or terms of use for all existing assets.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces a referring object detection benchmark and documents its fully automated construction pipeline, validation procedure, and dataset size in Appendix F.2.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or research with human subjects; the referring expressions are generated and filtered by an automated multi-agent pipeline.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or human-subject research, so IRB approval or equivalent review is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: The paper explicitly builds on MLLM backbones for detection and describes the model usage in the Method and Experiments sections; Appendix F.2 also describes the LLM-based automated referring-expression generation pipeline.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.