ConfDet: Learning Reliable Confidence for MLLM-based Detection

Xuanjie Mao[1,2,*] **Peng Ye**[3,*] **Ziteng Ma**[2] **Lin Zhang**[2]
Jiakang Yuan[2] **Haoyu Zhang**[2] **Fujun Han**[4]
Jiayuan Fan[2,†] **Tao Chen**[1,2,†]
[1]Shanghai Innovation Institute
[2]Fudan University
[3]The Chinese University of Hong Kong
[4]The Chinese University of Hong Kong, Shenzhen
[*]Equal contribution. [†]Corresponding authors.
jyfan@fudan.edu.cn, eetchen@fudan.edu.cn

Abstract

MLLM-based detection has shown promising end-to-end detection ability in recent years, but its generated detection results still lack reliable confidence for evaluation and interpretation. Since detection data itself do not provide human-annotated confidence labels, existing MLLM-based detection methods mostly obtain confidence implicitly from prompted self-assessment, generation probabilities, or external proxy scores. While these scores are available for evaluation, they are not explicitly connected with the actual detection quality. Inspired by conventional detectors that optimize confidence through classification, objectness, or quality estimation branches, we formulate confidence in MLLM-based detection as an explicit modeling target to be learned, optimized and calibrated. Specifically, we present ConfDet, a practical confidence framework that first uses discrete confidence tokens based supervised fine-tuning to enable stable and parsable confidence generation, then applies GRPO with detection and separation joint rewards to optimize confidence score separation and preserve detection quality, and finally performs condition-aware post-hoc calibration to align confidence with empirical accuracy. Experiments across different in-distribution and out-of-distribution benchmarks show that, ConfDet greatly improves AP through better confidence ranking and reduces confidence miscalibration across benchmarks, providing a reliable confidence for MLLM-based detection.

1 Introduction

In recent years, multimodal large language models (MLLMs) have demonstrated remarkable potential in visual understanding [1, 2], complex reasoning [3], and in-context learning [4]. Through visual-language alignment [5, 6], MLLM-based detection can understand target objects from natural language instructions and directly generate detection results. Benefiting from the semantic knowledge inherited from large language models, MLLM can support a wide range of detection tasks, including object detection [7, 8], open-vocabulary detection [9], complex semantic grounding [10, 11], and cross-task visual localization [12, 8]. As MLLM-based detection advances toward practical detection systems, producing accurate localization results alone is insufficient: model outputs also need to be accompanied by reliable confidence estimation. Such confidence provides an essential basis for standard evaluation [13, 14] and result interpretation, and further serves as a fundamental signal for ranking [13, 14], calibration [15, 16], and real-world deployment.

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

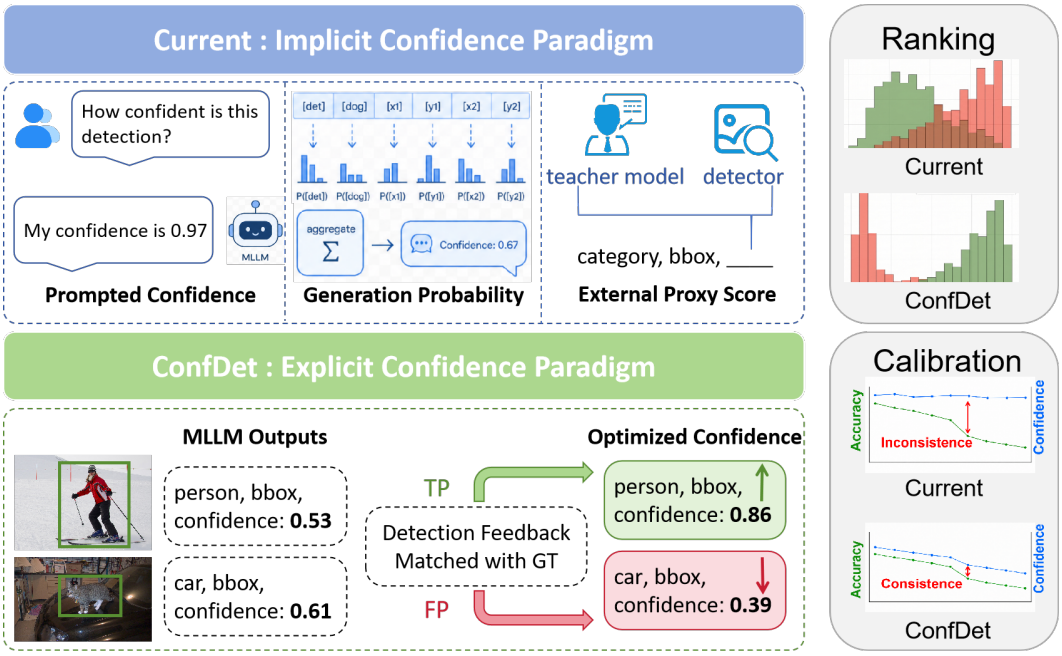

Figure 1: Comparison of confidence paradigms in MLLM-based detection. Left: the current implicit confidence paradigm and explicit confidence paradigm introduced by ConfDet. Right: compared with the current paradigm, ConfDet yields clearer TP–FP score separation for confidence ranking and better consistency between confidence and empirical accuracy for calibration.

Existing MLLM-based detection methods primarily focus on generating accurate categories and bounding boxes, while reliable confidence estimation remains underexplored. When confidence scores are required for AP evaluation, existing methods usually obtain them in indirect ways. Some works prompt the model to report its own certainty [17, 18], but such outputs are essentially language-level self-assessments and can be affected by instruction sensitivity and expression biases. Some works derive confidence from generation probabilities [12], but these probabilities mainly reflect sequence plausibility rather than detection quality. Other methods rely on external proxy scores, such as specialist-detector pseudo scores [7] and verifier-based re-scoring [19], but these scores remain post-hoc or inherit teacher bias. Thus, while existing methods can produce usable confidence scores, they typically treat confidence as an auxiliary score to be obtained after prediction, rather than as an explicit modeling target to be learned, optimized and calibrated.

This limitation is not accidental but stems from the training paradigm of MLLM-based detection. Detection annotations typically contain only category labels and bounding-box coordinates [14, 20], without human-annotated confidence labels for individual detections. Consequently, the training supervision for MLLM-based detection mainly teaches the model *what* to detect and *where* to localize it, but provides no explicit guidance on *how confident* each predicted detection should be. In contrast, conventional detectors usually produce confidence through dedicated classification, objectness, or localization-quality branches [21–23]. These outputs can be directly optimized with classification losses, IoU-aware objectives [24], or quality-aware supervision [25, 26], thereby establishing a clear connection between confidence and detection quality. Inspired by this comparison, we identify a key question: how can reliable confidence be explicitly modeled in MLLM-based detection as a learnable and optimizable target, despite the absence of human-annotated confidence labels?

To address this issue, we propose ConfDet, a practical confidence framework for MLLM-based detection. ConfDet explicitly turns confidence estimation in MLLM-based detection into a learnable, optimizable, and calibratable prediction problem, as illustrated in Figure 1. Concretely, ConfDet proceeds in three steps: First, we discretize confidence into a set of dedicated tokens [27], construct pseudo-labeled data with confidence annotations [9], and fine-tune the model to generate stable and parsable confidence. Second, considering that the previous step initializes confidence learning from pseudo labels and may inherit teacher bias, we further apply GRPO [28] to refine confidence

with ground-truth detection feedback, using a joint reward to promote TP–FP score separation and preserve detection quality. Finally, in order to address the observed conditional shift in empirical accuracy [29], we fit a lightweight condition-aware post-hoc calibration module [30] on a validation set and calibrate the confidence at inference time, making confidence more consistent with empirical accuracy.

We evaluate ConfDet from detection, calibration, and ablation perspectives. For detection, we report MS COCO Average Precision (AP) [14] on seven benchmarks, covering three in-distribution datasets and four out-of-distribution datasets. ConfDet improves AP on source domains and transfers consistently to unseen benchmarks. We further validate that the improvement mainly comes from better confidence ranking rather than the impact of training. For calibration, we evaluate probability calibration on four evaluation datasets using Expected Calibration Error (ECE) and Maximum Calibration Error (MCE) [15, 16]. ConfDet consistently yields better calibration, and reliability diagram shows that its calibrated confidence more closely tracks empirical detection accuracy. Finally, ablation studies verify the role of each stage: confidence-token-based SFT enables parsable confidence score generation, GRPO with detection and separation joint rewards improves ranking utility, and condition-aware post-hoc calibration further reduces miscalibration.

Our contributions are summarized as follows:

- **Explicit Problem Formulation.** For the first time, we identify reliable confidence as an important bottleneck in MLLM-based detection, and formulate reliable confidence as an explicit learnable, optimizable and calibratable model target, in the absence of human-annotated confidence labels.
- **Novel Three-stage Method.** We propose ConfDet, a practical framework that explicitly learns, optimizes, and calibrates confidence for MLLM-based detection. The framework leverages confidence-token-based SFT to generate parsable confidence, detection and separation joint rewards based GRPO to enable better ranking, and condition-aware post-hoc calibration to obtain reliable confidence.
- **Comprehensive Experiments.** We demonstrate the superiority of ConfDet on 7 benchmarks under both in-distribution and out-of-distribution settings. For example, on four out-of-distribution benchmarks, ConfDet improves mAP by 5.1 points on average over Rex-Omni and reduces the average ECE to 3.2, showing SOTA detection results while providing reliable confidence for ranking and calibration in MLLM-based detection.

2 Related Work

2.1 MLLM-based Detection

With the rapid progress of MLLMs [1, 3], object localization has become an important step toward fine-grained visual perception. Shikra [10] is an early grounded MLLM that incorporates spatial coordinates into the language interface, enabling the model to understand and generate location-aware responses in natural language. Kosmos-2 [11] further represents object descriptions with location tokens and grounds text to visual regions. ROD-MLLM [31] studies reliable object detection with free-form language by combining low-level localization and high-level comprehension. LMM-Det [7] studies vanilla object detection with MLLMs and improves recall through data distribution adjustment and inference design. Vision-R1 [32] introduces vision-guided reinforcement learning for localization beyond supervised fine-tuning. Rex-Omni [8] formulates diverse perception tasks as next-point prediction with large-scale grounding data and GRPO post-training. DetPO [19] further explores few-shot object detection with MLLMs through test-time prompt optimization and confidence-aware refinement. These works substantially improve the detection ability of MLLMs and make MLLM-based detection a promising generative detection paradigm. However, their main focus remains on generating more accurate categories and boxes, while reliable confidence for evaluation and interpretation is still less explored.

2.2 Detection Confidence

Conventional object detectors have developed rich confidence designs for both ranking and calibration [21–23]. For confidence ranking, IoU-Net [24] estimates localization confidence for predicted boxes, FCOS [23] introduces centerness to suppress low-quality predictions, and quality-aware detectors

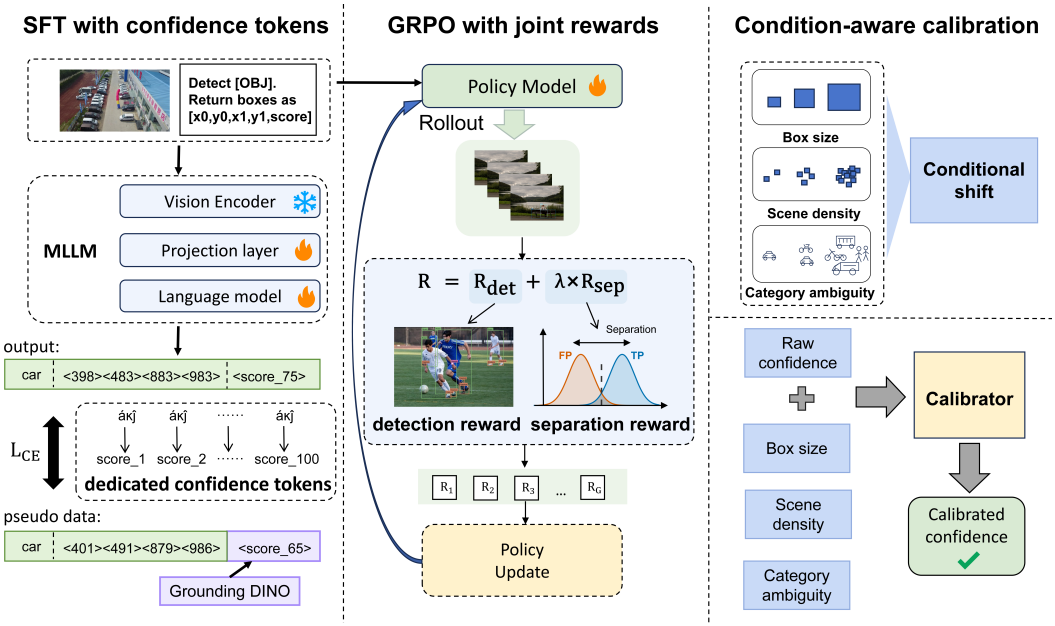

Figure 2: Overview of ConfDet. SFT with dedicated confidence tokens initializes stable and parsable confidence generation, GRPO with detection and separation joint rewards optimizes TP–FP score separation while preserving detection quality, and condition-aware post-hoc calibration aligns confidence with empirical accuracy at inference time.

such as GFL [25] and VarifocalNet [26] integrate localization quality into confidence targets. For confidence calibration, Küppers et al. [16] introduce multivariate confidence calibration for object detection by incorporating detection-specific variables such as box location and scale. Gilg et al. [29] further reveal box-size confidence bias and propose conditional calibration to reduce the mismatch between confidence and detector behavior. MCCL [33] jointly calibrates multiclass confidence and localization, while Cal-DETR [34] studies calibrated detection transformers through uncertainty-guided logit modulation and calibration-aware training. These works show that confidence in conventional detection is usually treated as learned, optimized and calibrated model output. In contrast, MLLM-based detection usually obtains confidence indirectly rather than treating it as a primary modeling target. ConfDet addresses this gap by turning confidence estimation in MLLM-based detection into an explicit learnable, optimizable and calibratable prediction problem.

3 Method

To explicitly learn, optimize and calibrate reliable confidence in MLLM-based detection in the absence of human-annotated confidence labels, we propose ConfDet, a three-stage framework consisting of SFT with dedicated confidence tokens, GRPO with detection and separation joint rewards, and condition-aware post-hoc calibration. We use Rex-Omni [8] as the base MLLM in our experiments. The overall framework is illustrated in Figure 2.

3.1 SFT with Dedicated Confidence Tokens

Conventional detectors directly output continuous confidence scores [21–23]. However, such scores are not naturally compatible with the autoregressive generation paradigm of language models [4], because continuous numbers can be fragmented by the tokenizer into unstable sub-sequences. To address this issue, ConfDet selects 100 low-frequency tokens from the model vocabulary and remaps them to dedicated confidence tokens $\{\texttt{score1}, \texttt{score2}, \ldots, \texttt{score100}\}$ [27]. This design constrains confidence prediction to a unified and controllable discrete space, which facilitates subsequent training and analysis. It also prevents numerical confidence values from being decomposed by the tokenizer into unstable multi-token forms.

Pseudo data construction: Since detection annotations contain only category labels and bounding boxes coordinates without human-annotated confidence labels [14, 20], we initialize confidence generation from pseudo confidence labels. We build a multi-source mixed dataset and use Grounding DINO [9] as the teacher model to provide pseudo confidence labels. The sampling distribution is summarized in Table 1.

Table 1: Sampling statistics of the mixed training dataset.

Data source	Quantity	Features
Objects365	5,000	Generic object detection
BDD100K	2,500	Autonomous-driving detection
VisDrone	2,500	Drone-view detection

For each sampled instance, we run inference with Grounding DINO and obtain a teacher prediction set

$$D = \{(\mathrm{cls}_k, \mathrm{bbox}_k, \mathrm{score}_k)\}. \tag{1}$$

We match D with the ground-truth annotations under geometric constraints and category matching. Specifically, we compute the intersection-over-union (IoU) between bbox_D and $\mathrm{bbox}_{\mathrm{GT}}$. A prediction is regarded as successfully matched when the predicted class is consistent with the ground-truth class and $\mathrm{IoU}(\mathrm{bbox}_D, \mathrm{bbox}_{\mathrm{GT}}) > 0.5$. For matched targets, we retain the ground-truth category and box coordinates, and initialize the confidence score with the corresponding teacher score. For unmatched targets, we retain the ground-truth category and box coordinates but assign a low initial confidence, since such targets are typically hard positives that even the teacher detector fails to recover.

Supervised fine-tuning: We perform supervised fine-tuning [6, 35] on the base MLLM using the pseudo training dataset. We keep the visual encoder frozen and train the projection layer and the LLM. Given an image I, an instruction Q, and a target serialized detection sequence $Y^\star = (y_1^\star, \ldots, y_T^\star)$, the model is trained by minimizing the token-level cross-entropy loss

$$L_{\mathrm{CE}} = -\sum_{t=1}^{T} \log \pi_\theta(y_t^\star \mid I, Q, y_{<t}^\star). \tag{2}$$

This objective teaches the model to jointly generate category, box coordinates, and tokenized confidence in a unified structured sequence.

The SFT stage equips the model with an initial ability to generate confidence scores. However, since this stage is essentially behavior cloning, the model is prone to inheriting teacher bias and does not directly optimize confidence. This motivates the GRPO stage below, which refines confidence with detection feedback.

3.2 GRPO with Detection and Separation Joint Rewards

To address the above limitations of SFT, we further apply GRPO [28] with detection and separation joint rewards to explicitly refine confidence learning with ground-truth detection feedback. For each image-instruction pair, we sample multiple rollouts from the current policy and match their generated detections with ground-truth boxes. The resulting rewards jointly evaluate detection quality and TP–FP score separation, allowing the model to optimize confidence without human-annotated confidence labels.

Given the same image I and instruction Q as in SFT, we sample G rollouts from the current policy model π_θ, producing an output set

$$\mathcal{Y} = \{Y_1, Y_2, \ldots, Y_G\}, \tag{3}$$

where each output sequence contains multiple predicted class labels, bounding boxes and confidence scores. For each sampled output Y_g in the group, after matching it with the ground-truth boxes, the

environment computes a scalar reward R_g using the joint reward function defined below. Based on the rewards of the whole group, we compute the relative advantage A_g of each output:

$$A_g = \frac{R_g - \text{mean}(\{R_1, \ldots, R_G\})}{\text{std}(\{R_1, \ldots, R_G\}) + \epsilon}. \tag{4}$$

The policy is then updated with a GRPO objective L_{GRPO} [28], which contains a policy-gradient term under importance-sampling constraints together with a KL-divergence penalty [36]. This regularization stabilizes policy updates and prevents the model from drifting away from the structured detection format established during SFT.

Detection and separation joint rewards. We define a joint reward as

$$R = R_{\text{det}} + \lambda R_{\text{sep}}, \tag{5}$$

where R_{det} preserves detection quality, R_{sep} enhances TP–FP score separation, and λ balances the two terms.

Detection reward R_{det}. We inherit the detection reward from Rex-Omni [8]. It is computed based on an F1-style IoU metric between the predicted boxes and the ground-truth boxes, and is used to preserve the model's spatial localization accuracy while suppressing hallucinated or duplicate predictions during generation. Specifically,

$$\text{Recall} = \frac{\sum_{i=1}^{m} r_i}{n}, \qquad \text{Precision} = \frac{\sum_{i=1}^{m} r_i}{m}, R_{\text{det}} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall} + \epsilon}, \tag{6}$$

where r_i indicates whether the i-th prediction is successfully matched, m is the number of predicted boxes, and n is the number of ground-truth boxes.

Separation reward R_{sep}. To improve the ranking ability of detection confidence, we introduce a separation reward that maximizes the margin between true positives and false positives in score space:

$$R_{\text{sep}} = \frac{\sum_{i \in \text{TP}} p_i + \sum_{j \in \text{FP}} (1 - p_j)}{|\text{TP}| + |\text{FP}|}. \tag{7}$$

This objective rewards high scores on TPs and low scores on FPs. It therefore optimizes discriminative score separation rather than probability calibration [15, 30], and we compare the two forms in Appendix B.1. In practice, it encourages the policy to assign higher scores to correct detections and lower scores to uncertain or erroneous ones, directly improving the ranking utility of the generated confidence. Reward matching and boundary cases are described in Appendix B.2.

3.3 Condition-aware Post-hoc Calibration

We collect the inference outputs of the GRPO-trained model and examine empirical accuracy under three inference-time conditions: box size, scene density, and category ambiguity. Box size is computed from the predicted bounding box and grouped by the COCO area thresholds 32^2 and 96^2 [14] into small, medium, and large; scene density is measured by the number of predicted detections in an image and grouped with thresholds 20 and 50; category ambiguity is estimated offline on the validation set as a category-level misclassification prior, which is measured by the probability that a detected object is assigned an incorrect class, and grouped into low, medium, high with thresholds 0.2 and 0.4. At inference time, this condition is obtained by looking up the ambiguity group of the model-predicted category, without using ground-truth labels.

As shown in Figure 3, accuracy varies substantially across box size [37], scene density, and category ambiguity. This indicates clear systematic conditional shift, which weakens the calibration of the confidence.

To address this conditional shift, we fit a lightweight condition-aware post-hoc calibrator on a held-out validation set. This module does not participate in main-model training and does not update the base model parameters. Instead, it learns Beta calibration [30] conditioned on predicted box size, scene density, and predicted-category ambiguity, all of which are available at inference time.

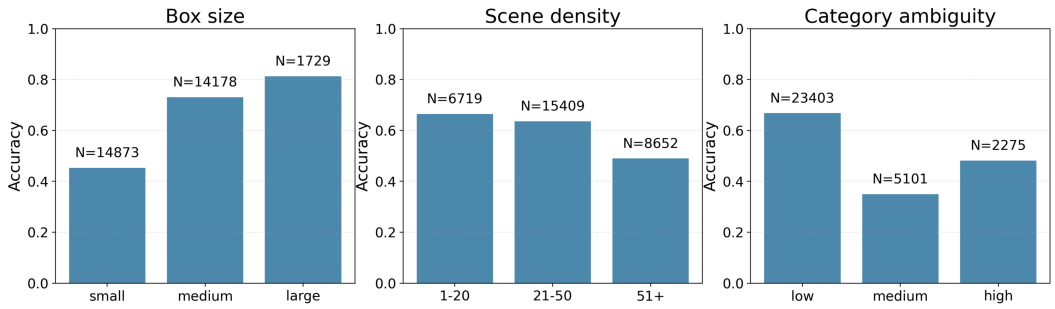

Figure 3: Conditional shift across box size, scene density, and category ambiguity.

Formally, if p denotes the raw confidence and $c = (c_{\text{size}}, c_{\text{dens}}, c_{\text{cat}})$ denotes the condition variables, the condition-aware Beta calibrator is defined as

$$\hat{p} = \sigma(a(c)\log p + b(c)\log(1-p) + d(c)), \tag{8}$$

where $\sigma(\cdot)$ is the sigmoid function, and $a(c)$, $b(c)$, and $d(c)$ are lightweight condition-dependent calibration parameters learned on the validation set. During inference, the calibrator maps raw confidence into a calibrated estimate that is better aligned with empirical detection accuracy.

4 Experiments

4.1 Evaluation Settings

We evaluate ConfDet from three perspectives: confidence ranking, calibration, and the contribution of each stage. For in-distribution (ID) evaluation, we use the three training-source benchmarks, Objects365 [20], BDD100K [38], and VisDrone [39], to examine source-domain behavior. For out-of-distribution (OD) evaluation, we evaluate zero-shot transfer on COCO [14], LVIS [40], KITTI [41], and UAVDT [42] to test confidence ranking under distribution shift. We adopt the AP metrics under the MS COCO evaluation protocol [14]. We also report ConfDet (no_score), where ConfDet uses constant scores instead of its predicted confidence, to validate that the AP gains mainly come from better confidence ranking.

We also evaluate confidence calibration on four datasets using Expected Calibration Error (ECE) and Maximum Calibration Error (MCE) [15, 16], and compare against several confidence sources.

4.2 Object Detection

In-distribution results. As shown in Table 2, ConfDet improves mAP over Rex-Omni [8] on Objects365, BDD100K, and VisDrone. These results indicate that explicit confidence modeling preserves source-domain detection behavior while providing a stronger ranking signal for AP.

Table 2: In-distribution detection on training-source benchmarks.

Dataset	Method	mAP	AP50	AP75	AP_s	AP_m	AP_l	AR
Objects365	Rex-Omni [8]	14.1	28.2	12.2	6.2	23.1	28.3	**27.9**
	ConfDet	**17.9**	**32.0**	**16.6**	**6.3**	**24.7**	**30.0**	27.1
BDD100K	Rex-Omni	20.7	43.8	16.9	**9.4**	29.1	55.7	**32.5**
	ConfDet	**24.7**	**45.9**	**22.4**	8.8	**31.4**	**56.8**	30.8
VisDrone	Rex-Omni	8.3	17.0	7.3	2.4	14.5	28.7	15.8
	ConfDet	**12.5**	**23.1**	**11.7**	**3.6**	**20.6**	**44.8**	**18.2**

Out-of-distribution results. As shown in Table 3, ConfDet achieves better AP than the compared MLLM baselines on COCO, LVIS, KITTI, and UAVDT. ConfDet (no_score) keeps generated detections from ConfDet but removes its learned confidence. The performance of ConfDet (no_score) is

close to Rex-Omni, whereas ConfDet consistently achieves higher AP than Rex-Omni. This indicates that the main AP gain comes from confidence ranking rather than the impact of training.

Table 3: Out-of-distribution detection on COCO, LVIS, KITTI, and UAVDT. ConfDet (no_score) isolates gains from confidence ranking rather than the impact of training.

Dataset	Method	Params	mAP	AP50	AP75	AP_s	AP_m	AP_l	AR
COCO	Seed1.6-Visual [43]	23B	36.9	56.4	38.3	10.5	38.6	57.0	46.5
	DeepSeek-VL2-Small [44]	3B	32.2	47.8	33.8	5.7	32.3	51.7	43.2
	MiniCPM-V 4.5 [45]	8B	19.2	32.3	19.1	1.2	11.8	34.7	24.2
	InternVL3-8B [46]	8B	24.6	41.2	23.9	7.3	23.7	40.8	35.4
	Qwen3-VL-8B [47]	8B	35.6	53.1	37.0	7.1	33.4	58.4	45.3
	Qwen3-VL-4B [47]	4B	35.5	53.2	36.8	7.8	35.0	57.2	45.3
	Rex-Omni [8]	3B	39.0	60.2	39.5	**13.0**	41.3	61.0	**52.0**
	ConfDet (no_score)	3B	38.1	59.2	38.5	11.4	40.7	60.8	51.1
	ConfDet	3B	**43.5**	**63.9**	**45.2**	12.2	**43.5**	**62.6**	51.1
LVIS	Seed1.6-Visual	23B	34.8	50.2	36.4	6.8	35.7	62.6	38.4
	DeepSeek-VL2-Small	3B	32.4	44.2	33.2	4.7	32.2	57.5	36.8
	MiniCPM-V 4.5	8B	14.3	24.7	13.5	0.6	7.4	32.7	15.6
	InternVL3-8B	8B	21.7	35.2	21.1	4.3	22.1	38.6	26.2
	Qwen3-VL-8B	8B	31.0	45.6	31.4	7.8	31.2	55.5	34.4
	Qwen3-VL-4B	4B	32.5	47.2	33.5	8.0	33.7	57.1	36.0
	Rex-Omni	3B	36.1	51.7	37.6	**10.7**	38.1	62.0	**41.2**
	ConfDet (no_score)	3B	36.2	52.4	37.1	10.4	38.4	62.3	41.1
	ConfDet	3B	**37.8**	**53.4**	**39.0**	10.5	**39.8**	**63.0**	41.1
KITTI	Seed1.6-Visual	23B	17.6	39.7	13.9	9.7	19.4	27.0	30.0
	DeepSeek-VL2-Small	3B	13.5	30.6	11.0	7.2	16.5	23.4	22.4
	MiniCPM-V 4.5	8B	11.3	27.9	7.9	5.5	9.7	18.7	16.8
	InternVL3-8B	8B	4.4	10.3	3.6	2.2	4.2	7.7	10.0
	Qwen3-VL-8B	8B	9.0	23.8	5.9	3.3	9.8	21.4	17.8
	Qwen3-VL-4B	4B	14.0	32.9	10.7	8.4	16.0	22.8	23.2
	Rex-Omni	3B	18.5	39.1	16.4	13.1	21.3	28.3	34.0
	ConfDet (no_score)	3B	19.4	41.3	16.6	11.9	21.4	29.7	34.7
	ConfDet	3B	**25.4**	**49.8**	**23.4**	**20.6**	**25.4**	**31.4**	**34.7**
UAVDT	Seed1.6-Visual	23B	24.8	43.0	24.2	4.9	34.4	48.9	39.0
	DeepSeek-VL2-Small	3B	3.6	7.1	3.4	0.2	6.5	22.4	13.2
	MiniCPM-V 4.5	8B	3.4	8.5	2.0	0.0	3.8	19.9	6.4
	InternVL3-8B	8B	1.0	2.8	0.7	0.1	0.4	10.5	1.2
	Qwen3-VL-8B	8B	10.5	17.3	11.8	1.8	16.3	39.8	15.6
	Qwen3-VL-4B	4B	14.5	23.8	16.2	2.7	36.3	48.9	41.0
	Rex-Omni	3B	24.4	40.4	26.9	9.4	39.0	65.8	**47.1**
	ConfDet (no_score)	3B	23.7	40.5	24.3	10.3	34.9	**66.2**	45.1
	ConfDet	3B	**31.5**	**49.9**	**34.8**	**14.2**	**40.4**	65.7	45.1

4.3 Probability Calibration

We evaluate confidence calibration by measuring how well predicted confidence matches empirical detection accuracy. We compare ConfDet with representative confidence sources in the current implicit paradigm: Prompt asks Qwen3-VL-8B [47] to report confidence, Griffon [12] uses generation probabilities, LMM-Det [7] learns detector-provided pseudo confidence, and Rex-Omni [8] (logit) averages output-token logits.

We fit the same condition-aware Beta calibrator separately to each method using identical image splits and fitting settings. As shown in Table 4, ConfDet consistently achieves the lowest ECE and MCE across all datasets after matched calibration. Figure 4 further shows that its calibrated confidence closely tracks empirical detection accuracy across confidence bins, indicating that the calibration stage reduces miscalibration under conditional shift. More analyses on the importance of confidence and additional validation are provided in Appendix B.3.

4.4 Ablation Study

We conduct an ablation study on COCO over the main components of ConfDet: SFT with dedicated confidence tokens for parsable confidence generation, GRPO with R_{det} and R_{sep} joint rewards for ranking optimization, and condition-aware post-hoc calibration for empirical accuracy alignment. Except for the removed component, all other settings are kept identical.

Table 5 shows the role of each stage. SFT is necessary to produce parsable confidence scores. In GRPO, R_{sep} alone hurts AP because score separation is optimized without preserving detection

Table 4: Confidence calibration on four evaluation datasets before and after matched calibration. Entries are ECE / MCE (%).

Method	COCO	LVIS	KITTI	UAVDT
Before calibration (raw)				
Prompt	27.4 / 60.0	31.3 / 72.9	25.1 / 64.3	15.3 / 59.9
Griffon	60.5 / 96.0	59.8 / 97.0	40.8 / 94.7	36.0 / 95.4
LMM-Det	13.7 / 21.1	9.3 / 13.9	23.9 / 47.5	13.2 / 20.0
Rex-Omni (logit)	22.4 / 28.1	22.3 / 26.2	12.2 / 18.8	11.6 / 36.1
ConfDet	6.2 / 20.0	13.4 / 22.0	9.1 / 21.0	12.4 / 35.1
After matched calibration				
Prompt	15.8 / 38.6	18.3 / 33.0	19.1 / 64.3	14.9 / 60.0
Griffon	48.4 / 96.7	46.2 / 94.4	19.7 / 98.1	20.6 / 99.0
LMM-Det	8.9 / 14.8	5.1 / 13.4	23.5 / 58.3	6.5 / 19.2
Rex-Omni (logit)	11.2 / 17.3	11.4 / 14.0	9.8 / 14.9	6.4 / 41.9
ConfDet	**2.0 / 8.2**	**3.4 / 6.6**	**2.3 / 7.1**	**5.0 / 11.3**

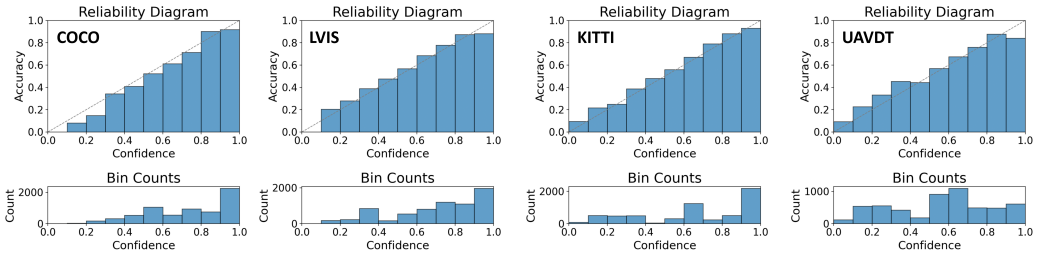

Figure 4: Reliability diagrams across four evaluation datasets. In each dataset, the upper panel shows empirical accuracy within each confidence bin, and the lower panel shows the number of predictions in each bin.

Table 5: Ablation on COCO. SFT with dedicated confidence tokens initializes confidence generation, GRPO with detection and separation joint rewards drives ranking utility, and condition-aware post-hoc calibration reduces miscalibration.

SFT	R_{det}	R_{sep}	Calib.	mAP	ECE ↓
×	×	×	×	39.03	–
✓	×	×	×	40.29	20.55
✓	✓	×	×	39.50	18.01
✓	×	✓	×	34.49	23.19
✓	✓	✓	×	43.52	6.23
✓	✓	✓	✓	**43.52**	**2.00**

quality, while the joint use of R_{det} and R_{sep} yields the best ranking performance. Post-hoc calibration leaves AP unchanged but further reduces ECE, showing that it mainly improves probability alignment rather than detection ranking.

5 Conclusion

Summary. In this paper, we formulate reliable confidence in MLLM-based detection as an explicit modeling target, despite the absence of human-annotated confidence labels. We propose ConfDet, a three-stage framework that learns, optimizes, and calibrates confidence for generated detections. ConfDet first performs SFT with dedicated confidence tokens to initialize parsable confidence generation, then applies GRPO with detection and separation joint rewards to optimize TP–FP score separation while preserving detection quality, and finally performs condition-aware post-hoc calibration to align confidence with empirical detection accuracy. Experiments show that ConfDet improves AP mainly through better confidence ranking and consistently reduces confidence miscalibration across benchmarks.

Limitations. This work has two main limitations. First, the SFT stage still relies on teacher-provided pseudo confidence labels, and residual teacher bias may remain after subsequent optimization and calibration. Second, ConfDet follows the current autoregressive MLLM-based detection paradigm, whose inference efficiency remains lower than conventional detectors, especially for edge-side deployment.

Outlook. Overall, ConfDet provides an important step toward making reliable confidence a standard output of MLLM-based detection. We hope this work can encourage future research on trustworthy MLLM-based detection, including better confidence supervision, more efficient confidence-aware inference, and stronger calibration under open-world conditions.

Acknowledgments and Disclosure of Funding

This work is supported by National Natural Science Foundation of China (No. 62571138), National Key R&D Program of China (No. 2026YFE0101200), National Natural Science Foundation of China (No. 62671175), and Shanghai General AI Foundation Models Program (No. 2025SHZDZX026D38). The computations in this research were performed using the CFFF platform of Fudan University.

References

[1] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.

[2] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[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] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020.

[5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.

[6] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, 2023.

[7] Jincheng Li, Chunyu Xie, Ji Ao, Dawei Leng, and Yuhui Yin. LMM-Det: Make large multimodal models excel in object detection. *arXiv preprint arXiv:2507.18300*, 2025.

[8] 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.

[9] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

[10] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal LLM's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023.

[11] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. KOSMOS-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023.

[12] Yufei Zhan, Yousong Zhu, Zhiyang Chen, Fan Yang, Ming Tang, and Jinqiao Wang. Griffon: Spelling out all object locations at any granularity with large language models. In *European Conference on Computer Vision*, 2024.

[13] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*, pages 740–755, 2014.

[15] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR, 2017.

[16] Fabian Küppers, Jan Kronenberger, Amirhossein Shantia, and Anselm Haselhoff. Multivariate confidence calibration for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020.

[17] Jincheng Huang, Jie Xu, Xiaoshuang Shi, Ping Hu, Lei Feng, and Xiaofeng Zhu. Revisiting confidence calibration for misclassification detection in VLMs. In *International Conference on Learning Representations*, 2026.

[18] David Bani-Harouni, Chantal Pellegrini, Paul Stangel, Ege Özsoy, Kamilia Zaripova, Nassir Navab, and Matthias Keicher. Rewarding doubt: A reinforcement learning approach to calibrated confidence expression of large language models. In *International Conference on Learning Representations*, 2026.

[19] Gautam Rajendrakumar Gare, Neehar Peri, Matvei Popov, Shruti Jain, John Galeotti, and Deva Ramanan. DetPO: In-context learning with multi-modal LLMs for few-shot object detection. *arXiv preprint arXiv:2603.23455*, 2026.

[20] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[21] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 2015.

[22] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[23] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[24] Borui Jiang, Ruixuan Luo, Jiayuan Mao, Tao Xiao, and Yuning Jiang. Acquisition of localization confidence for accurate object detection. In *European Conference on Computer Vision*, 2018.

[25] Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. In *Advances in Neural Information Processing Systems*, 2020.

[26] Haoyang Zhang, Ying Wang, Feras Dayoub, and Niko Sünderhauf. Varifocalnet: An IoU-aware dense object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[27] Yu-Neng Chuang, Prathusha Kameswara Sarma, Parikshit Gopalan, John Boccio, Sara Bolouki, Xia Hu, and Helen Zhou. Learning to route LLMs with confidence tokens. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 10859–10878. PMLR, 2025.

[28] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[29] Johannes Gilg, Torben Teepe, Fabian Herzog, and Gerhard Rigoll. The box size confidence bias harms your object detector. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023.

[30] Meelis Kull, Telmo Silva Filho, and Peter Flach. Beta calibration: A well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 623–631. PMLR, 2017.

[31] Heng Yin, Yuqiang Ren, Ke Yan, Shouhong Ding, and Yongtao Hao. ROD-MLLM: Towards more reliable object detection in multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14358–14368, 2025.

[32] Yufei Zhan, Yousong Zhu, Shurong Zheng, Hongyin Zhao, Fan Yang, Ming Tang, and Jinqiao Wang. Vision-R1: Evolving human-free alignment in large vision-language models via vision-guided reinforcement learning. *arXiv preprint arXiv:2503.18013*, 2025.

[33] Bimsara Pathiraja, Malitha Gunawardhana, and Muhammad Haris Khan. Multiclass confidence and localization calibration for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[34] Muhammad Akhtar Munir, Salman Khan, Muhammad Haris Khan, Mohsen Ali, and Fahad Shahbaz Khan. Cal-DETR: Calibrated detection transformer. In *Advances in Neural Information Processing Systems*, 2023.

[35] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, 2022.

[36] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[37] Fujun Han, Junan Chen, Xintong Zhu, Jingqi Ye, Xuanjie Mao, Tao Chen, and Peng Ye. Can multimodal large language models truly understand small objects? *arXiv preprint arXiv:2604.22884*, 2026.

[38] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. BDD100K: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[39] Pengfei Zhu, Longyin Wen, Xiao Bian, Haibin Ling, and Qinghua Hu. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*, 2018.

[40] Agrim Gupta, Piotr Dollár, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[41] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[42] Dawei Du, Yuankai Qi, Hongyang Yu, Yi Yang, Kaiwen Duan, Guorong Li, Weigang Zhang, Qingming Huang, and Qi Tian. The unmanned aerial vehicle benchmark: Object detection and tracking. In *European Conference on Computer Vision*, 2018.

[43] ByteDance Seed Team. Introduction to techniques used in Seed1.6. `https://seed.bytedance.com/en/blog/introduction-to-techniques-used-in-seed1-6`, 2025.

[44] Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, Zhenda Xie, Yu Wu, Kai Hu, Jiawei Wang, Yaofeng Sun, Yukun Li, Yishi Piao, Kang Guan, Aixin Liu, Xin Xie, Yuxiang You, Kai Dong, Xingkai Yu, Haowei Zhang, Liang Zhao, Yisong Wang, and Chong Ruan. DeepSeek-VL2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*, 2024.

[45] Tianyu Yu, Zefan Wang, Chongyi Wang, Fuwei Huang, Wenshuo Ma, Zhihui He, Tianchi Cai, Weize Chen, Yuxiang Huang, Yuanqian Zhao, et al. MiniCPM-V 4.5: Cooking efficient MLLMs via architecture, data, and training recipe. *arXiv preprint arXiv:2509.18154*, 2025.

[46] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Haojia Gao, Lijie Cui, Xizhou Zhu, Lewei Lu, Bin Li, Yu Qiao, and Jifeng Dai. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

[47] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[48] Peng Ye, Baopu Li, Yikang Li, Tao Chen, Jiayuan Fan, and Wanli Ouyang. β-DARTS: Beta-decay regularization for differentiable architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10874–10883, 2022.

[49] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78 (1):1–3, 1950.

[50] Peng Ye, Tong He, Baopu Li, Tao Chen, Lei Bai, and Wanli Ouyang. β-DARTS++: Bi-level regularization for proxy-robust differentiable architecture search. *arXiv preprint arXiv:2301.06393*, 2023. URL `https://arxiv.org/abs/2301.06393`.

[51] Peng Ye, Tong He, Shengji Tang, Baopu Li, Tao Chen, Lei Bai, and Wanli Ouyang. Stimulative training++: Go beyond the performance limits of residual networks. *arXiv preprint arXiv:2305.02507*, 2023. URL `https://arxiv.org/abs/2305.02507`.

[52] Peng Ye, Chenyu Huang, Mingzhu Shen, Tao Chen, Yongqi Huang, and Wanli Ouyang. Dynamic model merging with mixture of weights. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(8): 7925–7939, 2025. doi: 10.1109/TCSVT.2025.3544680.

A Broader Impact

ConfDet studies reliable confidence for MLLM-based detection. Better confidence can help rank detections, select confidence thresholds, calibrate predicted scores, and route uncertain predictions to review. This may be useful for risk-aware perception systems.

The method may also be misused in privacy-sensitive detection or monitoring scenarios. In addition, calibrated confidence is not a guarantee of correctness, especially under unseen domains, rare categories, low-quality images, or new deployment conditions. For high-stakes use, we recommend target-domain validation, application-specific threshold selection, and human or system-level oversight.

B Additional Discussion

B.1 Form of the confidence learning reward in GRPO

Our R_{sep} adopts an L1-style objective. Objective design also influences optimization stability in architecture search [48]. We also experimented with a Brier-style L2 objective [49]. The comparison below gives an intuition for why the L2 objective is less suitable for confidence ranking in our setting.

Let the model output confidence p, and let the probability that the prediction is a true positive be q. Under the binary target $y \in \{0, 1\}$, the expected squared error is

$$\mathbb{E}\big[(p-y)^2\big] = q(p-1)^2 + (1-q)p^2. \tag{9}$$

Expanding the terms gives

$$\mathbb{E}\big[(p-y)^2\big] = q(p^2 - 2p + 1) + (1-q)p^2 = p^2 - 2qp + q. \tag{10}$$

Taking the derivative with respect to p, we obtain

$$\frac{\partial}{\partial p}\,\mathbb{E}\big[(p-y)^2\big] = 2p - 2q = 2(p-q). \tag{11}$$

Setting the derivative to zero yields

$$p = q. \tag{12}$$

Thus, under a squared-error objective, the optimal prediction tends to match the average TP probability. In our setting, this weakens TP–FP contrast and may push predictions toward a narrow score region, as shown in Figure 5.

Table 6 gives the quantitative comparison between the L1-style and L2-style confidence learning rewards.

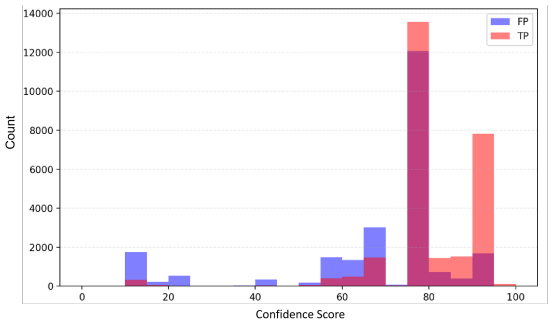

Figure 5: Score distribution under the L2-based separation reward. The learned score distribution collapses to a narrow constant-score region, which weakens TP–FP separation.

Table 6: Comparison between L1-style and L2-style confidence learning rewards.

Dataset	mAP (L1)	mAP (L2)	ECE (L1)	ECE (L2)
COCO	**43.52**	42.20	**4.0**	9.0
LVIS	**37.80**	36.22	**3.0**	8.0
KITTI	**25.45**	24.18	**2.0**	11.0
UAVDT	**31.54**	23.16	**5.0**	12.0

B.2 Reward boundary cases.

In reward computation, invalid or unparsable outputs receive zero reward. When both the ground truth and the prediction are empty, we set the detection reward to 1 and the separation reward to 0. When the ground truth is empty but the model still predicts boxes, the detection reward is set to 0 and the separation reward penalizes high confidence on these predictions. For normal detection cases, we first compute the detection reward using the IoU-based precision–recall F1 score. We then determine TP and FP predictions by same-class greedy matching with an IoU threshold of 0.5. Each ground-truth box can be matched at most once, so duplicate predictions beyond the first match are treated as false positives. Predictions with class mismatch are also treated as false positives. If a predicted box does not contain a valid confidence token, we use the default score 0.5, and all confidence values are clipped into $[0, 1]$ before computing the separation reward.

B.3 Downstream Utility of Confidence

The learned confidence is also useful for downstream utility. We evaluate three simple uses in Figure 6.

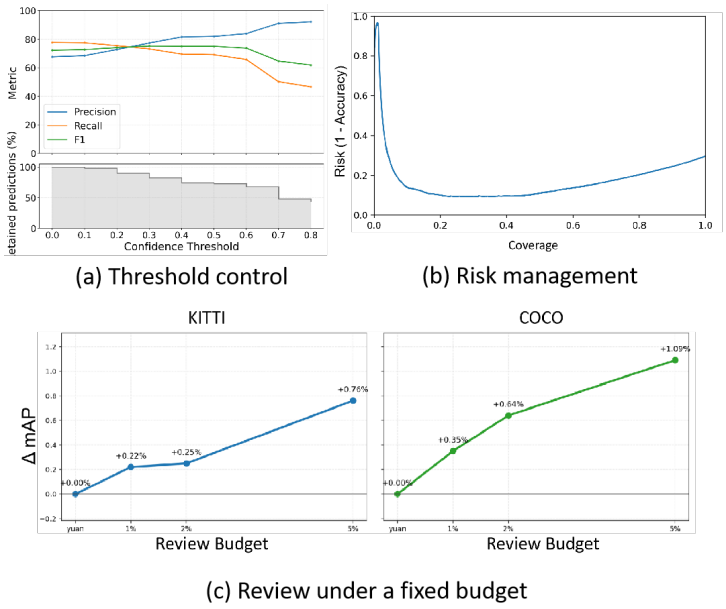

Figure 6: Downstream utility of the learned confidence. (a) Threshold control shows the trade-off between precision, recall, and F1. (b) Risk management shows the risk of retained predictions at different coverage levels. (c) Review under a fixed budget measures the mAP gain from sending low-confidence samples to review.

Threshold control. As the confidence threshold varies, precision, recall, and F1 change smoothly. This shows that the confidence score can be used to select different operating points.

Risk management. The risk–coverage curve evaluates prediction risk after retaining different fractions of high-confidence predictions. Lower coverage corresponds to keeping more confident predictions.

Review under a fixed budget. We send a fixed proportion of low-confidence samples to a review module. The gains increase as the review budget grows.

B.4 Additional Experimental Results

Confidence and localization. On COCO, same-class greedy matching at IoU ≥ 0.5 yields 4,689 true positives. Confidence–IoU Spearman correlation is 0.548. Using learned confidence on the same detections raises AP75 from 38.5 to 45.2.

Table 7: Localization quality by confidence on matched COCO true positives.

Confidence	TPs	Mean IoU	Median IoU	IoU ≥ 0.75
$[0, 0.5)$	1,764	0.748	0.755	51.4%
$[0.5, 0.7)$	155	0.774	0.798	60.0%
$[0.7, 0.9)$	1,993	0.882	0.914	88.7%
$[0.9, 1]$	777	0.899	0.937	90.2%

Teacher analysis. We audit 1,000 COCO images unused in SFT/GRPO, containing 7,204 ground-truth objects, using score-independent same-class one-to-one matching at IoU ≥ 0.5 and 5,000 image-level cluster-bootstrap resamples. Grounding DINO-B misses 1,359 objects (18.86%); small/medium/large miss rates are 32.89%/16.10%/9.65%. Among recovered true positives, con-

Table 8: Teacher versus student: mAP / raw ECE (%) before calibration.

Dataset	Grounding DINO-B	ConfDet
COCO	57.35 / 14.16	43.52 / 6.20
LVIS	32.13 / 22.68	37.80 / 13.40
KITTI	29.74 / 17.60	25.45 / 9.10
UAVDT	12.93 / 17.32	31.54 / 12.40

Table 9: Recall on teacher-hit and teacher-missed ground truths. Gap changes are relative to the base, in percentage points (pp), with 95% confidence intervals.

Stage	Hit recall	Missed recall	Change in hit-minus-missed gap
Rex-Omni base	78.37%	13.83%	Reference
SFT	77.28%	13.76%	-1.02 $[-2.53, +0.37]$
GRPO	77.30%	12.51%	$+0.25$ $[-1.12, +1.61]$

trolling for matched IoU, log ground-truth area, and category fixed effects gives teacher-missed-minus-hit confidence offsets of -2.92 pp (SFT; 95% CI $[-5.75, +0.03]$) and -3.03 pp (GRPO; $[-5.27, -0.74]$). On 4,500 common true positives, the adjusted GRPO-minus-SFT change is -2.42 pp ($[-5.87, +0.96]$). GRPO reduces false positives on this audit subset from 3,036 to 1,949.

Detector comparison and inference efficiency. COCO-trained conventional detectors use native classes on COCO and zero-shot, class-agnostic evaluation on transfer datasets. Grounding DINO-B and ConfDet use language-conditioned categories. End-to-end latency is measured on an RTX 4090 with batch size 1, greedy decoding, 20 warm-ups, and 200 held-out images repeated three times. ConfDet takes 1,312.6 ms/image on average (P95: 2,576.5 ms). Each box adds one confidence token, and the calibrator takes 0.095 ms/image. Generation-latency correlations with box count and token count are 0.988 and 0.996, respectively.

Calibration-data sensitivity. We fix a held-out COCO evaluation subset and independently sample ten calibration sets at each size from a disjoint pool. Ratios are relative to the 5,000-image validation set; results are mean $\pm$ standard deviation. Raw ECE on this fixed subset is 5.90. Sensitivity to proxy data is also examined in architecture search [50].

Table 10: Detector comparison. Parameters are in millions; latency is ms/image. Conventional-detector and ConfDet timing uses RTX 4090, batch 1. †Grounding DINO-B timing is an RTX A6000 PyTorch value.

Method	Params	Latency	COCO	LVIS	KITTI	UAVDT
Faster R-CNN-R50	41.8	19.3	40.8	8.7	35.7	14.8
RetinaNet-R50	38.0	20.2	39.8	8.4	38.9	15.8
FCOS-R50	32.3	18.6	40.0	8.9	38.8	18.2
ATSS-R50	32.3	19.6	43.9	9.5	40.5	19.1
GFL-R50	32.4	19.8	44.2	9.6	41.2	19.7
VarifocalNet-R50	32.9	23.4	45.2	8.9	42.2	19.6
TOOD-R50	32.2	24.8	46.5	10.3	41.4	20.0
RTMDet-S	8.9	36.2	45.9	10.0	38.5	20.5
Deformable DETR-R50	40.1	25.6	45.8	9.4	35.3	18.4
DINO-R50	47.7	33.7	46.6	10.2	36.1	22.3
Grounding DINO-B†	233.0	148.6	57.4	32.1	29.7	12.9
ConfDet	3,000	1,312.6	43.5	37.8	25.4	31.5

Table 11: Generation latency grouped by number of predicted boxes.

Predicted boxes	Requests	Mean generation latency (ms)
0–20	455	569.8
21–50	131	1,880.7
51+	14	19,842.7

Table 12: COCO calibration-data sensitivity. ECE values are percentages.

Ratio	Calibration images	Raw ECE	Calibrated ECE
0.5%	25	5.90	8.98 ± 3.39
1%	50	5.90	7.32 ± 1.11
2%	100	5.90	5.20 ± 1.23
5%	250	5.90	4.34 ± 1.12
10%	500	5.90	2.70 ± 0.70
15%	750	5.90	2.39 ± 0.63
20%	1,000	5.90	2.01 ± 0.78

C Experimental Details

C.1 Datasets and Evaluation Protocol

Datasets. We evaluate ConfDet in both in-distribution and out-of-distribution settings. Objects365 [20], BDD100K [38], and VisDrone [39] are used for confidence learning and in-distribution evaluation. COCO [14] and LVIS [40], as well as KITTI [41] and UAVDT [42], are used for out-of-distribution evaluation.

Evaluation protocol. For detection evaluation, we follow the MS COCO-style AP protocol [14], and report mAP, AP50, AP75, AP for small/medium/large objects and AR. For calibration evaluation, we report ECE and MCE [15, 16].

Table 13: Dataset usage in ConfDet.

Dataset	Usage	Domain
Objects365	Training / Eval.	Generic objects
BDD100K	Training / Eval.	Driving scenes
VisDrone	Training / Eval.	UAV-view scenes
COCO	Eval.	Common objects
LVIS	Eval.	Long-tail objects
KITTI	Eval.	Driving scenes
UAVDT	Eval.	UAV-view scenes

Table 14: Evaluation settings.

Item	Setting
Detection metric	MS COCO-style AP
Calibration metrics	ECE and MCE
TP criterion for calibration	Same-class matching with IoU ≥ 0.5
Confidence bins	10 equal-width bins

Assets. All datasets, pretrained models, and codebases used in this work are public research assets and are properly cited. We follow their licenses and terms of use.

C.2 Model and Training Details

Base model. ConfDet is built on Rex-Omni [8], a 3B-parameter MLLM detector. We add a discrete confidence vocabulary and train the model to output one confidence token after each generated box.

Training settings. The main SFT and GRPO settings are shown in Tables 15 and 16. Auxiliary subnetwork supervision has also been explored for residual-network training [51].

Table 15: SFT configuration.

Item	Setting
Base checkpoint	Rex-Omni
Trainable modules	LLM and multimodal projector
Frozen modules	Vision encoder
Precision	bf16
Epochs	1
Per-device batch size	8
Gradient accumulation	8
Optimizer	AdamW
Learning rate	2×10^{-5}
Projector learning rate	2×10^{-5}
Vision tower learning rate	–
Weight decay	0.01
Max gradient norm	1.0
Warmup ratio	0.03
LR schedule	Cosine
Max sequence length	4096

Table 16: GRPO configuration.

Item	Setting
Initialization	SFT checkpoint
Advantage estimator	GRPO
Training steps	500
Rollouts per prompt	16
Global actor batch size	32
Rollout batch size	32
Update micro-batch size	2 per device
Experience micro-batch size	4 per device
Optimizer	AdamW
Learning rate	1×10^{-6}
Weight decay	0.01
KL coefficient	1.0×10^{-2}
Reward weight	$\lambda = 0.2$

C.3 Inference and Calibration Details

Inference. At inference time, the model is prompted with category names and asked to return boxes with confidence scores. We use deterministic decoding for evaluation. Input-adaptive weight fusion is explored in dynamic model merging [52].

Post-hoc calibration. The Beta calibrator [30] is fit on held-out validation predictions with box size, scene density, and category ambiguity conditions. It does not update the base model. Unless otherwise specified, 20% of validation images are used to fit the calibrator and the remaining 80% are used to report ECE and MCE. The split is performed at the image level with a fixed random seed.

C.4 Compute Resources

All reported experiments were conducted on a single node with four RTX 4090 GPUs. Runtime varies across datasets and experimental settings.

Table 17: Inference configuration.

Item	Setting
Max generation length	4096
Temperature	0
Top-p	0.8
Top-k	1
Repetition penalty	1.05
Output format	Category, Bounding Box, Confidence score

Table 18: Compute resources.

Stage	Hardware	GPUs	Main cost
SFT	RTX 4090	4	One epoch over training data
GRPO	RTX 4090	4	500 steps, 16 rollouts per prompt
Calibration	CPU / GPU	1	Fit post-hoc calibrator
Evaluation	RTX 4090	1	Inference and metric computation

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 main claims are stated in the abstract and Section 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 limitations are discussed in Section 5.

   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: [N/A]

Justification: This paper does not present formal theoretical results.

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 information needed to reproduce the main experiments is provided in Sections 3–4 and Appendix C.

   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: [Yes]

Justification: An anonymized code package with instructions is provided in the supplementary material.

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 evaluation setting is described in Section 4.1, and additional training, inference, calibration, and compute details are provided in Appendix C.

   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: [N/A]

   Justification: We do not report statistical significance tests for the high computational cost, following common practice for large-scale MLLM post-training experiments.

   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 compute resources are reported in Appendix C.4.

   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: We have reviewed the NeurIPS Code of Ethics and do not identify any deviations.

   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: Broader impacts are discussed in Appendix A.

    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: This paper does not release high-risk trained models.

    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: Existing datasets, pretrained models, and codebases are properly cited, and their licenses and terms of use are respected.

    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: We provide an anonymized code package with documentation in the supplementary material.

    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: This work does not involve crowdsourcing or research with human subjects.

    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: This work does not involve crowdsourcing or research with human subjects.

    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: [N/A]

Justification: LLMs were used only for writing polishing and editing, and were not used as an important component of the core methodology.

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.