LOCATE IN ANY SCENE: AGENTIC OBJECT LOCALIZATION WITH MULTIMODAL LARGE LANGUAGE MODELS UNDER DIVERSE VISUAL DEGRADATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

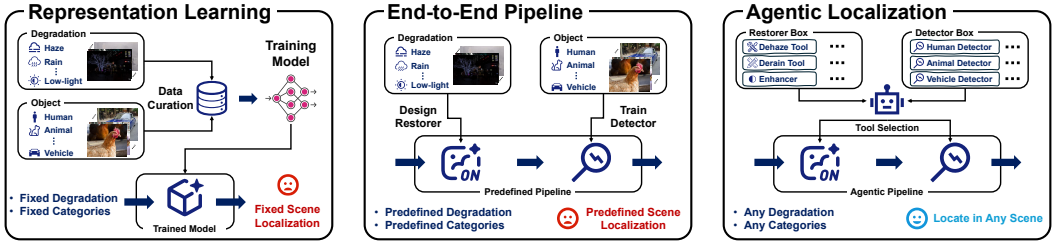
Object localization in real-world scenarios remains challenging due to diverse image degradations and heterogeneous object distributions, which substantially limit the generalization of existing grounding models. Conventional approaches, including scene-specific representation learning and end-to-end pipeline design, are inherently constrained by predefined conditions and therefore lack the flexibility required for evolving environments. In this paper, we propose LocAS, an agentic framework that formulates object localization as an adaptive decision-making process. Rather than relying on static pipelines, LocAS employs a Multimodal Large Language Model (MLLM) as a central agent to dynamically compose localization workflows by selecting from a toolbox of restoration modules and specialized detectors. Specifically, LocAS consists of two key components: Self-Adaptive Image Restoration, which determines whether and how an input image should be enhanced for downstream localization, and Multi-Expertise Localization, which coordinates multiple domain-specialized detectors and consolidates their predictions through instance-level reasoning. To further improve decision quality under fine-grained conditions, we introduce Experience-Aware Calibration and extend LocAS to LocAS-X. LocAS-X accumulates node-level decision experience from a small set of annotated samples and incorporates it into agent memory to support more informed reasoning during inference. This mechanism enables the system to progressively refine its decision policy and better adapt to diverse and dynamically changing degradation scenarios. Extensive experiments on eight challenging benchmarks demonstrate that LocAS-X significantly outperforms existing MLLM-based grounding models, achieving an average improvement of 30.59% in F1 score. These results highlight the potential of agentic localization and provide a promising foundation for robust object localization in complex and dynamic real-world environments.

1 INTRODUCTION

Object localization has witnessed substantial progress Liu et al. (2016); Carion et al. (2020); Zhang et al. (2022), yet existing models still face challenges in generalizing to diverse and rapidly changing real-world environments. In practice, images are frequently affected by diverse degradations, such as haze, rain, low-light conditions, and underwater distortion, which significantly alter visual appearance and degrade localization performance. At the same time, practical object localization involve highly diverse categories and scene distributions, further increasing task complexity. Consequently, localization models trained under standard conditions often struggle to generalize across diverse degradation types and application scenarios Sun et al. (2022).

Existing approaches to robust object localization under degraded conditions can be broadly categorized into two paradigms: representation learning and end-to-end pipeline. **Representation Learning** methods rely on curated datasets with fixed degradation types and object categories to learn specialized representations Sakaridis et al. (2025; 2018), as shown in Fig. 1(a). While effective in closed-vocabulary localization settings, these methods often fail to generalize even under slight shifts in degradation conditions or scene distributions, resulting in substantial performance drops. Alternatively, **End-to-End Pipeline** adopts a modular approach Liu et al. (2022); Zhang & Patel

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



(a) (b) (c)

Figure 1: Comparison of object localization paradigms: (a) representation learning with curated datasets, (b) end-to-end pipeline design, and (c) our proposed agentic localization framework that dynamically selects tools across diverse scenarios.

(2018), where a grounding model trained under standard conditions is combined with a dedicated image restoration module for specific scenarios, as shown in Fig. 1(b). Although this design avoids large-scale data collection and retraining, it typically relies on predefined restoration strategies and manually designed pipelines. As a result, suboptimal restoration may degrade localization performance, and the entire pipeline often requires redesign when adapting to new scenarios. Therefore, both paradigms remain inherently restricted to predefined degradation conditions and localization targets, making them difficult to generalize to diverse and dynamically changing environments. In particular, traditional object localization models struggle in dynamic real-world environments, where a single grounding model or fixed pipeline cannot simultaneously adapt to diverse environmental conditions and shifts in object distribution.

The limitations of existing paradigms suggest that robustness under diverse degradations requires adaptive, input-aware localization rather than improved representations or fixed pipelines. Recent advances in Multimodal Large Language Models (MLLMs) Wu et al. (2023a), particularly in grounding and visual reasoning Xiaomi & Team (2025); Guo et al. (2025), along with the emergence of agentic systems Schick et al. (2023); Zuo et al. (2025), provide a compelling foundation for this paradigm shift. Large Language Model (LLM) agents demonstrate strong capabilities in understanding complex visual scenes, reasoning over multiple candidates, and making context-aware decisions. Such properties naturally align with the requirements of real-world object localization, where both degradation conditions and target objects vary significantly across samples.

Building on this perspective, we propose an **Agentic Localization** framework that formulates object localization as a dynamic decision-making process, as illustrated in Fig. 1(c). Instead of treating localization as a static pipeline, our approach automatically composes a workflow by selecting from a diverse toolbox of image restoration modules and specialized detectors. Given an input image, the system first performs scene perception and then adaptively selects appropriate restoration strategies and localization modules. Based on this idea, we develop Locate in Any Scene (LocAS), which consists of Self-Adaptive Image Restoration (SAIR) and Multi-Expertise Localization (MEL). As shown in Fig. 2, SAIR dynamically selects restoration strategies through iterative LLM perception, while MEL integrates multiple specialized detectors to collaboratively handle diverse localization scenarios. To further enhance fine-grained reasoning, we introduce Experience-Aware Calibration (EAC) and extend LocAS into an enhanced framework, LocAS-X. LocAS-X extracts node-level decision experience from a small amount of annotated data and incorporates it into agent memory. During inference, the agent draws upon this accumulated experience to calibrate its reasoning process, thereby improving localization accuracy and facilitating rapid adaptation across diverse degradation conditions. Our contributions are summarized as follows:

- We propose an agentic localization framework, termed LocAS, which consists of SAIR and MEL for adaptive selection of restoration tools and detectors.
- We further introduce EAC and extend our framework to LocAS-X, which leverages past experience extracted from annotated data as agent memory to calibrate its reasoning process and improve localization performance.

- Extensive experiments demonstrate that LocAS-X consistently outperforms existing MLLM baselines and generalizes well across diverse degradations and scenarios. In particular, LocAS-X achieves an average F1-score improvement of 30.59% across eight datasets, demonstrating its effectiveness and adaptability across diverse real-world environments.

2 Related Works

Robust Object Localization under Degradation: Object localization under degraded visual conditions has progressed along two main directions. The first constructs degradation-specific datasets and learns specialized representations. Benchmarks such as DarkFace Yang et al. (2020) highlight the performance gap of standard detectors under low-light conditions, while datasets like BDD100K Yu et al. (2020) and HazyDet Feng et al. (2024) expand coverage to diverse weather conditions and viewpoints. Methods built on these datasets learn degradation-aware representations or adaptation strategies, such as HLA-Face Wang et al. (2021), AERIS Cui et al. (2022), and DAI-Net Du et al. (2024). However, these approaches are typically tied to specific degradation types or domains and generalize poorly across scenarios. The second direction develops restoration-assisted localization pipelines. Early work such as AOD-Net Li et al. (2017) couples dehazing with localization, and recent methods (e.g., VRD-IR Yang et al. (2023), SR4IR Kim et al. (2024), GEFU Wang et al. (2025b), UniRestore Chen et al. (2025)) explicitly optimize restoration for downstream recognition. Despite their effectiveness, these approaches typically rely on fixed restoration modules and detectors, and seldom consider whether restoration is necessary or how to select the optimal restoration-localization combination for a given input. Motivated by this limitation, we move beyond static pipelines and propose a more adaptive localization framework.

Open-Vocabulary Visual Grounding and Localization: Visual grounding and open-vocabulary localization underpin language-conditioned object localization, typically achieved through vision–language alignment. Methods such as GLIP Li et al. (2022) and Grounding DINO Liu et al. (2024) integrate large-scale language supervision into localization frameworks, while MLLMs including Qwen3-VL Bai et al. (2025), GLM-4.1V Hong et al. (2025), and InternVL3.5 Wang et al. (2025c) demonstrate strong grounding capabilities for open-world understanding. Rex-Omni Jiang et al. (2025b) further improves detection, particularly for small and dense objects. More recent work extends grounding from final prediction to intermediate reasoning. For example, Rex-Thinker Jiang et al. (2025a) introduces Chain-of-Thought (CoT) reasoning, P2G Chen et al. (2024) dynamically invokes external modules, GroundingAgent Luo et al. (2025) employs agentic reasoning, and VGR Wang et al. (2025a) enhances reasoning via region selection and reuse. Despite these advances, existing methods are primarily evaluated on clean and static scenarios, leaving robustness under severe degradation, occlusion, and dynamic conditions unexplored.

3 Methodology

To build an agentic localization system for diverse degradations and scenarios, we propose LocAS, a framework that automatically selects restoration modules and specialized detectors to construct effective pipelines. As shown in Fig. 2, LocAS includes two key components: SAIR, which produces localization-friendly images, and MEL, which leverages complementary detectors for diverse scenarios. To further improve decision quality, we introduce EAC and extend the framework to LocAS-X, which leverages experience derived from annotated data as agent memory to calibrate its decision-making process.

3.1 Self-Adaptive Image Restoration

Images in complex real-world environments frequently suffer from various degradations, posing significant challenges to reliable object localization. To address this issue, as shown in Fig. 2, LocAS introduces a SAIR pipeline that dynamically selects appropriate restoration strategies based on the input image condition. Specifically, an LLM perception module first analyzes the image and predicts a degradation-aware scene profile from a predefined set, including `normal`, `fog`, `rain`, `underwater`, `low-light`, and `noise`. Based on the predicted scene label, SAIR automatically activates the corresponding restoration tool from a restorer pool that includes dehazing, deraining, denoising, brightness enhancement, and other restoration tools.

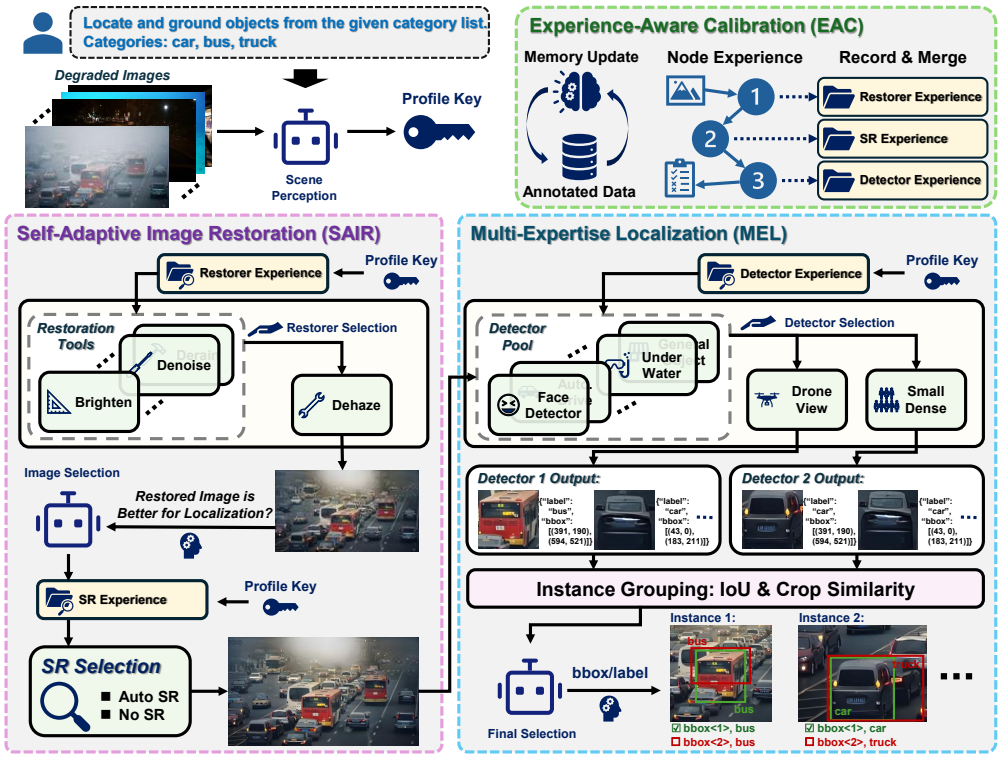


Figure 2: Overview of the LocAS-X framework. SAIR adaptively selects restoration strategies through LLM perception, while MEL integrates multiple specialized detectors to handle diverse localization scenarios. EAC harvests node-level decision experience from annotated data to enable experience-aware reasoning and improve localization performance.

However, restoration does not always improve downstream localization performance. In some cases, restoration may alter background colors or distort object boundaries, which can negatively affect localization accuracy. To address this issue, SAIR introduces an image selection mechanism that compares the original image with the restored version and selects the one that is more suitable for localization. If restoration degrades the image quality for localization, the original image is used instead. Importantly, the selection decision prioritizes localization-oriented cues such as object visibility, boundary clarity, and structural integrity rather than visual aesthetics, ensuring that the chosen image is more suitable for downstream object localization.

In the final restoration stage, SAIR applies Super-Resolution (SR) to increase spatial resolution and improve the visibility of target objects. This step is particularly useful when small or distant objects are present in the scene. Given a predefined target resolution, the SR module automatically selects an appropriate upscaling factor to enhance image clarity. Overall, through this adaptive restoration pipeline, SAIR produces a localization-friendly image representation that enhances object visibility and provides improved input representation for downstream localization modules.

3.2 Multi-Expertise Localization

Instead of relying on a single localization model, LocAS adopts a MEL pipeline that integrates multiple specialized detectors to handle diverse object distributions across a wide range of scenarios. During scene perception, the LLM also receives a target category list. Conditioned on both the image content and target categories, the LLM-based agent routes each query to the top-K most suitable detectors from a pool of specialized experts, including the `general-purpose detector`, `dense&small-object detector`, `autonomous-driving detector`, `drone-view detector`, `underwater detector`, and `face detector`. Each detector specializes in particular visual patterns or object distributions, enabling the system to handle a wide range of localization scenarios and generate comprehensive initial grounding proposals.

After detector selection, the chosen detectors are applied to the preprocessed image to generate candidate object proposals, as shown in Fig. 2. Since multiple detectors may produce duplicate predictions for the same object instance, LocAS performs an instance grouping step to consolidate localization results. Candidate bounding boxes are grouped based on spatial overlap and visual similarity between grounded regions, forming instance groups that likely correspond to the same object. Formally, let $\mathcal{B} = \{b_n\}_{n=1}^{N}$ denote the set of candidate bounding boxes predicted by multiple detectors. We first compute the Intersection-over-Union (IoU) between two candidate boxes b_i and b_j. If the IoU exceeds a predefined threshold, the two proposals are considered spatially consistent. However, spatial overlap alone is insufficient when detectors produce slightly shifted predictions, especially when objects appear in dense scenes. To improve grouping robustness, we additionally compute a visual similarity score between their cropped image regions. For each candidate box b_i, the bounding box is expanded with ratio α to obtain an enlarged crop region, which is subsequently resized to 32×32 patch and flattened into a vector representation. The visual similarity between two cropped regions is defined as:

$$S_{\text{vis}}(i, j) = \frac{\mathbf{v}_i^\top \mathbf{v}_j}{\|\mathbf{v}_i\|_2 \|\mathbf{v}_j\|_2}, \tag{1}$$

where $\mathbf{v}_i$ denotes the vectorized representation of the cropped region. This lightweight similarity computation avoids the overhead of using a vision encoder to extract latent embeddings, thereby maintaining high efficiency. Finally, a candidate proposal b_i is assigned to an existing group $\mathcal{G} = \{g_k\}_{k=1}^{K}$ if both spatial and visual consistency are satisfied:

$$\text{IoU}(b_i, b_j) > \theta_{\text{IoU}} \quad \wedge \quad S_{\text{vis}}(i, j) > \phi_{\text{vis}}, \tag{2}$$

where θ_{IoU} and ϕ_{vis} denote the IoU and visual similarity thresholds, respectively.

In the final stage of MEL, the system selects the best bounding box-label pair for each instance group. For each group, we extract the region defined by the grouped bounding boxes and provide it, together with the corresponding localization proposals, to the LLM for final decision-making. Compared with prompting on the entire image, instance-level crops are more token-efficient and encourage the LLM to focus on the target object. This design is particularly beneficial for small objects, where the LLM may otherwise fail to attend to relevant regions and produce suboptimal decisions. For illusory false positive groups, where detectors fail to identify valid objects or repeatedly generate background proposals, the system allows rejecting all candidates within the group. Overall, this MEL strategy enables LocAS to leverage complementary detector strengths while relying on LLM reasoning to resolve ambiguous localizations.

3.3 Experience-Aware Calibration

To this end, LocAS establishes a rule-based agentic localization flow that performs effectively across diverse scenarios. However, decisions based solely on predefined rules lack the flexibility to capture fine-grained characteristics of individual images. In practice, images with similar degradation types may still require different restoration or localization strategies. For example, applying dehazing to lightly foggy images may blur object features, while SR may amplify restoration artifacts and mislead detectors in MEL. In such cases, bypassing the corresponding restoration or SR step and directly performing localization on the original image can yield better results. Moreover, the optimal combination of detectors often varies across scenarios, highlighting the need for more fine-grained detector selection. To address these challenges, we introduce EAC, which enables the system to automatically collect, store, and summarize node-level decision experience from a small set of annotated data. The accumulated experience is then integrated into agent memory and leveraged to calibrate subsequent decision-making, thereby rapidly improving overall localization performance during inference.

As shown in Fig. 2, we introduce an experience retrieval mechanism for three critical decision nodes in LocAS: restorer selection, SR selection, and detector selection, resulting in an experience-aware framework, LocAS-X. Before deployment, we collect a small set of annotated images and perform exploratory runs to obtain decision knowledge via empirical evaluation. For each image, the LLM-based agent first infers a fine-grained scene profile during perception, including the scene label as well as attributes such as `object scale`, `object density`, `visibility`, and `illumination`. This profile serves as a key for organizing and retrieving experience. During

experience harvesting, the system executes the LocAS pipeline on annotated images under different decision configurations (right-top panel in Fig. 2), exploring restoration strategies, SR options, and detector selections. Localization performance is evaluated using ground truth, and for each decision node, the experience manager records metrics such as F1 scores. For images sharing the same profile, records are aggregated based on performance statistics to identify the most effective tools, forming a compact experience memory for calibrating future decisions. Unlike many memory-based methods that summarize experience as trajectories or textual descriptions, EAC directly records localization results at each decision node. This design avoids long contextual prompts that may distract the LLM, providing a concise and efficient reference for reliable decision-making.

During inference, LocAS-X incorporates experience-aware reasoning into the inference process. For each input image, the system first performs scene perception to obtain a fine-grained profile key describing its visual characteristics. The experience memory is queried to retrieve the top-K most similar profiles together with their associated recommendations at each decision node. These retrieved experiences are provided to the LLM-based agent as reference information, allowing the system to jointly consider current visual evidence and accumulated historical experience when selecting restoration strategies, SR configurations, and detectors. As more annotated data are accumulated, the experience memory is continuously updated and refined to reflect the evolving environment, enabling LocAS-X to progressively optimize its decision policy. With this memory-driven calibration mechanism, EAC turns the localization framework from a rule-based pipeline into an experience-guided system that learns from previous decisions and outcomes to improve performance across diverse degradation scenarios.

4 Experiments

4.1 Experimental Setup

Datasets and Baselines: We evaluate our methods on multiple benchmarks covering challenging real-world image degradations. Specifically, we include **HazyDet** Feng et al. (2024) for foggy-scene localization from drone viewpoints; **MARIS** Li et al. (2025) for underwater object localization; **DarkFace** Yang et al. (2020) for low-light face localization; the Night (**B-Night**) and Rainy (**B-Rainy**) subsets of BDD100K Yu et al. (2020) for nighttime and rainy driving scenarios; **DAWN** Kenk & Hassaballah (2020) for adverse weather conditions, including fog, rain, snow, and sandstorms; and **ExDark** Loh & Chan (2019) for object localization under diverse low-light conditions. We additionally report results on **COCO** Lin et al. (2014) as a reference under standard visual conditions. We compare our methods against several recent MLLMs capable of grounding-based object localization. For open-source models with fewer than 10B parameters, we consider **Ovis2.5-9B** Lu et al. (2025), **MiniCPM-V-4.5** Yu et al. (2025), **GLM-4.1V-9B** Hong et al. (2025), **InternVL3.5-8B** Wang et al. (2025c), **Qwen3-VL-8B** Yang et al. (2025), and **Qwen3.5-9B** Qwen Team (2026). We further compare against two closed-source frontier models, **GPT-4.1** Achiam et al. (2023) and **Gemini-2.5-Flash** Team et al. (2023), and a recent localization-specific model, **LocateAnything** Wang et al. (2026), to provide a more comprehensive set of baselines. For each open-source MLLM, we perform grounding-oriented Supervised Fine-Tuning (SFT) on the training split of each dataset for one epoch using LoRA with rank 8. For the closed-source models, we directly query their APIs with prompts specifying the required grounding output format.

Implementation Details: For restoration, we adopt several fast and effective models, including RIDCP Wu et al. (2023b) for dehazing, MPRNet Zamir et al. (2021) for deraining, SwinIR Liang et al. (2021) for denoising, LLFlow Wang et al. (2022) for brightness enhancement, and ESRGAN Wang et al. (2018) for SR. In the detector pool, GroundingDINO-B Liu et al. (2024) serves as the `general-purpose detector` because of its strong general grounding capability, whereas Rex-Omni Jiang et al. (2025b) is used as the `small&dense-object detector` due to its effectiveness in densely populated scenes. For the remaining domain-specific experts, we fine-tune Qwen3-VL-8B on non-degraded data from each target domain: the clear subset of BDD100K for the `autonomous-driving detector`, the pre-synthesis subset of HazyDet for the `drone-view detector`, the paired normal-light set of DarkFace for the `face detector`, and the open-vocabulary training split of MARIS for the `underwater detector`. For perception and reasoning, we employ the original Qwen3-VL-8B instruct model as the LLM-based agent. During instance grouping, the bounding-box expansion ratio α is set to 0.25, while

Table 1: Evaluation results on diverse datasets using F1 score (%) at IoU $= 0.5$. The best and second-best results are highlighted in bold and underlined, respectively. LocAS-X consistently ranks first and surpasses the competing baselines by a clear margin.

Method	COCO	HazyDet	MARIS	DarkFace	B-Night	B-Rainy	DAWN	ExDark	Average
GPT-4.1	9.05	2.52	30.72	0.00	0.70	2.27	3.43	15.49	8.02
Gemini-2.5-Flash	13.03	13.59	29.54	0.50	17.47	10.96	13.00	18.19	14.54
LocateAnything	25.79	21.73	0.56	20.94	13.90	16.00	22.99	11.12	16.63
Ovis2.5-9B	18.45	17.20	15.87	5.60	7.40	11.29	16.64	17.56	13.75
MiniCPM-V-4.5	17.78	12.00	36.66	0.94	5.66	5.18	11.71	16.92	13.36
GLM-4.1V-9B	30.11	14.35	38.41	0.00	5.48	2.83	14.01	28.79	16.75
InternVL3.5-8B	33.25	19.53	24.30	6.28	7.45	7.44	15.73	32.24	18.28
Qwen3-VL-8B	14.33	31.34	<u>51.56</u>	7.97	8.60	3.96	30.59	38.44	23.35
Qwen3.5-9B	23.85	3.25	48.14	0.29	0.09	2.00	1.16	27.65	13.30
LocAS	<u>67.46</u>	<u>45.72</u>	46.16	<u>44.87</u>	<u>37.41</u>	<u>40.03</u>	<u>47.71</u>	<u>64.96</u>	<u>49.29</u>
LocAS-X	**68.10**	**53.54**	**52.29**	**47.40**	**43.55**	**44.97**	**52.44**	**69.19**	**53.94**

the spatial and visual thresholds θ_{IoU} and ϕ_{vis} in Eq. 2 are set to 0.5. The top-K value for detector selection is set to 2, and the target SR resolution is 2K. For LocAS-X, the experience memory is initialized using 10 randomly sampled annotated examples from the training split of each dataset, yielding 80 samples in total. During inference, the top-K value for experience retrieval is set to 3.

4.2 Experimental Results

Benchmark Results: We evaluate localization performance using the F1 score (%) at IoU $= 0.5$, and report the results in Table 1 (Additional evaluation metrics are provided in the Appendix B). For CoT-based MLLMs, we disable the thinking mode, as we observe that CoT reasoning often collapses in object localization tasks (See the Appendix E for examples). Despite being fine-tuned on their respective datasets, baseline MLLMs still exhibit limited localization performance. As shown in Table 1, most models achieve low F1 scores on challenging datasets, especially under foggy, low-light, and rainy conditions. For instance, Qwen3.5-9B achieves only 3.25% on HazyDet, 0.29% on DarkFace, and 2.00% on B-Rainy. Even after fine-tuning on the respective datasets, the strongest grounding model, Qwen3-VL-8B, achieves only 31.34% on HazyDet, 7.97% on DarkFace, and 3.96% on B-Rainy. These results indicate that SFT alone is insufficient to bridge the domain gap caused by severe degradations, highlighting the need for adaptive restoration and expertise-aware localization strategies.

In contrast, LocAS significantly improves performance across all datasets through the cooperation of SAIR and MEL. For example, on HazyDet, DarkFace, and B-Rainy, LocAS improves the F1 score from 31.34% to 45.72%, from 7.97% to 44.87%, and from 11.29% to 40.03%, respectively, demonstrating the effectiveness of dynamically selecting restoration strategies. On COCO, where input images typically do not require restoration, adaptive selection of specialized detectors still yields substantial improvement, increasing the F1 score from 33.25% to 67.46%. Furthermore, with experience-aware reasoning by EAC, LocAS-X consistently improves upon LocAS, achieving additional gains such as 7.82% on HazyDet and 6.13% on MARIS. These results indicate that leveraging historical experience enables more fine-grained tool selection beyond rule-based decisions. Overall, LocAS-X outperforms the best baseline by an average of 30.59% across all datasets, demonstrating consistently superior performance and strong generalization across diverse scenarios. These results further validate the effectiveness of adaptive restoration, multi-expertise collaboration, and experience-aware reasoning for robust object localization in dynamic environments. Although the LocAS framework integrates multiple models to improve localization, its end-to-end latency remains comparable to that of a single MLLM-based grounding model. This is because SAIR restores degraded images and substantially reduces false-positive predictions, thereby shortening the output sequence generated by the MLLM. A detailed latency analysis is provided in Appendix G. Moreover, comparisons with closed-vocabulary object detectors are reported in Appendix C.

Qualitative Analysis: We provide qualitative visualizations of the LocAS-X pipeline to illustrate how the system improves localization under different scenarios (Fig. 3). Each row shows one example and each column corresponds to a stage of the pipeline. In the first example, the scene is identified as low-light, and brightness enhancement followed by SR significantly improves the visibility of the face region, enabling correct localization through the collaboration of multiple detectors and LLM reasoning. For the second example, the system determines that restoration is unnecessary and directly applies SR, after which the underwater-specialized detector produces the final localization



Figure 3: Qualitative visualization of the LocAS-X pipeline under diverse scenarios.

result. In the third example, the image is recognized as a foggy scene; dehazing and SR substantially improve vehicle visibility, allowing the LLM to select tighter bounding boxes from multiple detector proposals. For the last example, although restoration is initially recommended, the restored image potentially degrades performance, and the LLM therefore selects the original image for SR, which leads to correct localization results. Overall, these examples demonstrate that LocAS-X can adaptively generate localization-friendly images across diverse scenarios and effectively combine multi-detector proposals with LLM reasoning to produce accurate final localizations.

Visualization of Localization Performance: We provide qualitative comparisons of localization results across different baselines in Fig. 4. Compared with baseline models such as Ovis2.5-9B, GLM-4.1V-9B, and Qwen3-VL-8B, LocAS-X produces more accurate and complete localizations under diverse degradation conditions. Baseline methods often suffer from imprecise localizations, incorrect labels, and excessive false positives in challenging scenarios such as fog, underwater, and low-light environments. In contrast, LocAS-X consistently identifies target objects with tighter bounding boxes and more accurate predictions. These qualitative results further validate the robustness and flexibility of LocAS-X in complex realistic settings. Additional qualitative comparisons of localization results on each dataset are provided in the Appendix H.

4.3 Ablation Studies

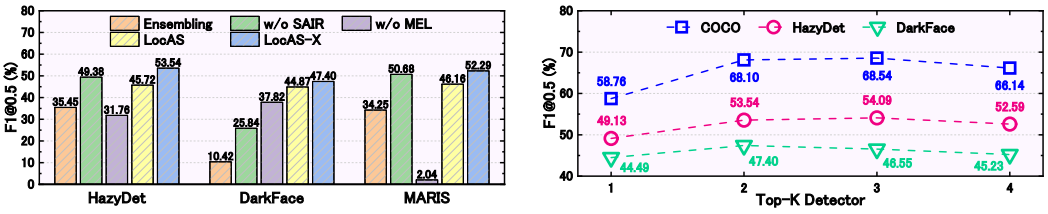
We conduct ablation studies on multiple datasets to evaluate the contribution of each module in LocAS-X and the impact of the number of detectors in MEL. Fig. 5(a) presents the component analysis, where we compare LocAS-X with LocAS and three additional ablated variants. To evaluate the effectiveness of SAIR, we remove the SAIR module and directly perform localization on the original input image (denoted as **w/o SAIR**). To assess the role of MEL, we replace it with a `general-purpose detector` (denoted as **w/o MEL**). To validate the contribution of agentic decision-making, we introduce a model-ensembling variant (denoted as **Ensembling**). In this variant, scene perception is followed by rule-based restoration and SR, and the resulting image is then processed by all six expert detectors in the pool. Their predictions are subsequently averaged and merged via class-wise IoU matching to obtain the final localization results. The results show that each module consistently contributes to performance across scenarios. In particular, on DarkFace and MARIS, the F1 score drops from 47.40% to 25.84% without SAIR and from 52.29% to 2.04% without MEL, respectively, demonstrating the importance of both modules. Furthermore, incorporating EAC refines the original rule-based pipeline with accumulated experience, leading to consistent performance improvements. The ensembling baseline performs poorly without agentic decision-making, as it cannot select the most appropriate restoration strategy or adaptively invoke



Figure 4: Qualitative comparison of localization results between LocAS-X and baseline MLLMs.

fallback operations when needed. Moreover, aggregating predictions from all detectors introduces substantial false positives, which outweigh the benefits of cross-detector complementarity. This observation confirms that both agent-based tool selection and final prediction reasoning are critical to the effectiveness of LocAS.

We further analyze the impact of the number of selected detectors (Top-K) in MEL on localization performance. We evaluate configurations with $K = 1$ to 4 across three datasets. Results show that even with a single detector, the LLM-based agent can select a suitable model and achieve reasonable performance. Increasing K initially improves performance by leveraging complementary detector expertise, with the best trade-off achieved at $K = 2$. Further increasing K to 4 introduces more false positives, leading to a slight degradation in localization performance. Therefore, to balance performance and computational cost, we set $K = 2$ in the LocAS-X framework. Additional ablations on the EAC sample size and a detailed comparison of tool-selection behavior between LocAS and LocAS-X are presented in the Appendices D and F, respectively.



(a) (b)

Figure 5: Ablation studies: (a) Component contributions. (b) Effect of detector count.

5 Conclusion

We propose LocAS, an agentic framework that formulates object localization as a dynamic decision process, enabling adaptive composition of restoration and localization pipelines for diverse real-world scenarios. By integrating SAIR and MEL, LocAS overcomes the limitations of static pipelines through dynamic restoration selection and complementary detector expertise. We further extend LocAS to LocAS-X with EAC, enabling experience-aware calibration of agent decisions under fine-grained scene variations. LocAS-X consistently outperforms existing baselines across diverse degradations, achieving substantial F1-score gains and strong generalization. These results position agentic localization as a promising direction for robust and scalable real-world visual understanding.

AI USE STATEMENT

In this work, we used generative AI tools for language polishing and text editing, as well as for assisting with code generation and the initial drafting of prompt templates. The AI-generated outputs were used only as auxiliary materials or preliminary drafts to improve efficiency. All subsequent modifications, refinement, integration, and maintenance of the generated text, code, and prompt templates were performed manually by the authors. We did not rely on generative AI tools to independently formulate the scientific conclusions or make final research decisions. All AI-assisted content was carefully reviewed and verified by the first author and the corresponding author, including checks for technical correctness, consistency with the actual implementation and experiments, and appropriateness of the final presentation. In particular, AI-assisted code and prompt templates were further modified and tested by the authors before being used in the experiments. We have reviewed all AI-assisted work and take full responsibility for the final content of this work, including all text, claims, code, and other artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects, personally identifiable information, or the collection of sensitive or private data. The datasets and models used in this study are publicly available and are used solely for research purposes. We do not identify any specific ethical concerns arising from the methodology, experiments, or intended applications presented in this work.

REPRODUCIBILITY STATEMENT

We have made efforts to ensure the reproducibility of our work. Detailed experimental settings, including the configurations and evaluation protocols used throughout our experiments, are provided in the corresponding experimental setup sections of the main paper and appendices. More complete implementation details and reference code are included in the Supplementary Material. To further facilitate the understanding and reproduction of our implementation, all prompts used in our framework are provided in Appendix I. In addition, upon acceptance, we will publicly release the complete source code as well as the fine-tuned adapter weights for our method and all implemented baselines.

REFERENCES

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.

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.

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

I Chen, Wei-Ting Chen, Yu-Wei Liu, Yuan-Chun Chiang, Sy-Yen Kuo, Ming-Hsuan Yang, et al. Unirestore: Unified perceptual and task-oriented image restoration model using diffusion prior. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 17969–17979, 2025.

Jiaxing Chen, Yuxuan Liu, Dehu Li, Xiang An, Weimo Deng, Ziyong Feng, Yongle Zhao, and Yin Xie. Plug-and-play grounding of reasoning in multimodal large language models. *arXiv preprint arXiv:2403.19322*, 2024.

Ziteng Cui, Yingying Zhu, Lin Gu, Guo-Jun Qi, Xiaoxiao Li, Renrui Zhang, Zenghui Zhang, and Tatsuya Harada. Exploring resolution and degradation clues as self-supervised signal for low quality object detection. In *European Conference on Computer Vision*, pp. 473–491. Springer, 2022.

Zhipeng Du, Miaojing Shi, and Jiankang Deng. Boosting object detection with zero-shot day-night domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12666–12676, 2024.

Changfeng Feng, Zhenyuan Chen, Xiang Li, Chunping Wang, Jian Yang, Ming-Ming Cheng, Yimian Dai, and Qiang Fu. Hazydet: Open-source benchmark for drone-view object detection with depth-cues in hazy scenes. *arXiv preprint arXiv:2409.19833*, 2024.

Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, et al. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*, 2025.

Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, et al. Glm-4.5 v and glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv preprint arXiv:2507.01006*, 2025.

Qing Jiang, Xingyu Chen, Zhaoyang Zeng, Junzhi Yu, and Lei Zhang. Rex-thinker: Grounded object referring via chain-of-thought reasoning. *arXiv preprint arXiv:2506.04034*, 2025a.

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*, 2025b.

Mourad A Kenk and Mahmoud Hassaballah. Dawn: vehicle detection in adverse weather nature dataset. *arXiv preprint arXiv:2008.05402*, 2020.

Rahima Khanam and Muhammad Hussain. Yolov11: An overview of the key architectural enhancements. *arXiv preprint arXiv:2410.17725*, 2024.

Jaeha Kim, Junghun Oh, and Kyoung Mu Lee. Beyond image super-resolution for image recognition with task-driven perceptual loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2651–2661, 2024.

Bingyu Li, Feiyu Wang, Da Zhang, Zhiyuan Zhao, Junyu Gao, and Xuelong Li. Maris: Marine open-vocabulary instance segmentation with geometric enhancement and semantic alignment. *arXiv preprint arXiv:2510.15398*, 2025.

Boyi Li, Xiulian Peng, Zhangyang Wang, Jizheng Xu, and Dan Feng. Aod-net: All-in-one dehazing network. In *Proceedings of the IEEE international conference on computer vision*, pp. 4770–4778, 2017.

Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10965–10975, 2022.

Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1833–1844, 2021.

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 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

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

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pp. 21–37. Springer, 2016.

Wenyu Liu, Gaofeng Ren, Runsheng Yu, Shi Guo, Jianke Zhu, and Lei Zhang. Image-adaptive yolo for object detection in adverse weather conditions. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 1792–1800, 2022.

Yuen Peng Loh and Chee Seng Chan. Getting to know low-light images with the exclusively dark dataset. *Computer vision and image understanding*, 178:30–42, 2019.

Shiyin Lu, Yang Li, Yu Xia, Yuwei Hu, Shanshan Zhao, Yanqing Ma, Zhichao Wei, Yinglun Li, Lunhao Duan, Jianshan Zhao, et al. Ovis2. 5 technical report. *arXiv preprint arXiv:2508.11737*, 2025.

Liqin Luo, Guangyao Chen, Xiawu Zheng, Yongxing Dai, Yixiong Zou, and Yonghong Tian. Connecting the dots: Training-free visual grounding via agentic reasoning. *arXiv preprint arXiv:2511.19516*, 2025.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126(9):973–992, 2018.

Christos Sakaridis, Haoran Wang, Ke Li, Arpit Jadon, Wim Abbeloos, Daniel Olmeda Reino, Luc Van Gool, Dengxin Dai, et al. Acdc: The adverse conditions dataset with correspondences for robust semantic driving scene perception. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36:68539–68551, 2023.

Tao Sun, Mattia Segu, Janis Postels, Yuxuan Wang, Luc Van Gool, Bernt Schiele, Federico Tombari, and Fisher Yu. Shift: a synthetic driving dataset for continuous multi-task domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 21371–21382, 2022.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Jiacong Wang, Zijian Kang, Haochen Wang, Haiyong Jiang, Jiawen Li, Bohong Wu, Ya Wang, Jiao Ran, Xiao Liang, Chao Feng, et al. Vgr: Visual grounded reasoning. *arXiv preprint arXiv:2506.11991*, 2025a.

Sen Wang, Shao Zeng, Tianjun Gu, Zhizhong Zhang, Ruixin Zhang, Shouhong Ding, Jingyun Zhang, Jun Wang, Xin Tan, Yuan Xie, et al. From enhancement to understanding: Build a generalized bridge for low-light vision via semantically consistent unsupervised fine-tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13804–13814, 2025b.

Shihao Wang, Shilong Liu, Yuanguo Kuang, Xinyu Wei, Yangzhou Liu, Zhiqi Li, Yunze Man, Guo Chen, Andrew Tao, Guilin Liu, et al. Locateanything: Fast and high-quality vision-language grounding with parallel box decoding. *arXiv preprint arXiv:2605.27365*, 2026.

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

Wenjing Wang, Wenhan Yang, and Jiaying Liu. Hla-face: Joint high-low adaptation for low light face detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16195–16204, 2021.

Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pp. 0–0, 2018.

Yufei Wang, Renjie Wan, Wenhan Yang, Haoliang Li, Lap-Pui Chau, and Alex Kot. Low-light image enhancement with normalizing flow. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 2604–2612, 2022.

Jiayang Wu, Wensheng Gan, Zefeng Chen, Shicheng Wan, and Philip S Yu. Multimodal large language models: A survey. In *2023 IEEE International Conference on Big Data (BigData)*, pp. 2247–2256. IEEE, 2023a.

Rui-Qi Wu, Zheng-Peng Duan, Chun-Le Guo, Zhi Chai, and Chongyi Li. Ridcp: Revitalizing real image dehazing via high-quality codebook priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22282–22291, 2023b.

LCT Xiaomi and Core Team. Mimo-vl technical report. *arXiv preprint arXiv:2506.03569*, 1(2):5, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Wenhan Yang, Ye Yuan, Wenqi Ren, Jiaying Liu, Walter J Scheirer, Zhangyang Wang, Taiheng Zhang, Qiaoyong Zhong, Di Xie, Shiliang Pu, et al. Advancing image understanding in poor visibility environments: A collective benchmark study. *IEEE Transactions on Image Processing*, 29:5737–5752, 2020.

Zizheng Yang, Jie Huang, Jiahao Chang, Man Zhou, Hu Yu, Jinghao Zhang, and Feng Zhao. Visual recognition-driven image restoration for multiple degradation with intrinsic semantics recovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14059–14070, 2023.

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*, pp. 2636–2645, 2020.

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.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14821–14831, 2021.

Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.

He Zhang and Vishal M Patel. Density-aware single image de-raining using a multi-stream dense network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 695–704, 2018.

Yushen Zuo, Qi Zheng, Mingyang Wu, Xinrui Jiang, Renjie Li, Jian Wang, Yide Zhang, Gengchen Mai, Lihong V Wang, James Zou, et al. 4kagent: agentic any image to 4k super-resolution. *arXiv preprint arXiv:2507.07105*, 2025.

A LIMITATIONS

Our framework has several limitations. First, LocAS inherits the limitations of its underlying MLLM-based detectors, which may be less accurate than specialized closed-vocabulary detectors in fixed-scene settings. At the same time, MLLM-based detectors provide open-vocabulary localization and stronger adaptability to domain shifts, which are central to the scenarios considered in this work. Improvements in future grounding models are therefore likely to further enhance LocAS. Second, the current implementation covers only a limited set of restoration operations and localization domains. Although these components address several common visual degradation scenarios, specialized domains such as infrared, X-ray, and medical imaging are not evaluated. The modular structure of LocAS allows new tools and detectors to be incorporated, and extending the framework to broader domains remains an important direction for future research.

B EVALUATION ON ADDITIONAL METRICS

We additionally evaluate all methods using Recall at IoU $= 0.5$ (R@0.5), which is widely used in MLLM-based grounding and localization. As reported in Table 2, the conclusions under R@0.5 remain consistent with those based on F1 score: LocAS-X achieves the strongest performance across all datasets and competing baselines.

As a supplementary comparison with traditional object localization, we report Average Precision (AP) at IoU $= 0.5$ in Table 3. Since MLLM-based grounding models do not provide calibrated confidence scores, all predictions are assigned a uniform confidence score of 1.0. Consequently, the resulting AP values do not fully capture confidence-ranking quality and should be interpreted primarily as a reference for comparison with conventional object detectors.

Table 2: Evaluation results on diverse datasets using R@0.5 (%). The best and second-best results are highlighted in bold and underlined, respectively. LocAS-X consistently ranks first and surpasses the competing baselines by a clear margin.

Method	COCO	HazyDet	MARIS	DarkFace	B-Night	B-Rainy	DAWN	ExDark	Average
GPT-4.1	7.21	2.04	34.16	0.00	0.82	1.67	2.82	14.34	7.88
Gemini-2.5-Flash	8.13	8.61	24.81	0.42	9.07	9.22	10.18	13.67	10.51
LocateAnything	43.58	21.07	0.92	12.52	12.87	16.00	23.32	7.46	17.22
Ovis2.5-9B	35.21	20.23	18.55	1.45	6.24	6.99	22.45	28.70	17.48
MiniCPM-V-4.5	45.42	12.11	40.38	4.93	8.59	7.24	13.69	21.20	19.20
GLM-4.1V-9B	36.26	18.44	39.46	0.00	16.32	8.39	21.00	30.77	21.33
InternVL3.5-8B	56.19	19.58	25.43	16.00	15.43	10.66	25.69	32.67	25.21
Qwen3-VL-8B	52.84	33.11	<u>56.16</u>	30.09	23.76	10.54	37.71	55.25	37.43
Qwen3.5-9B	39.29	4.75	52.82	1.22	0.05	1.22	0.60	21.69	15.21
LocAS	<u>70.82</u>	<u>46.15</u>	51.12	<u>46.23</u>	<u>44.42</u>	<u>43.33</u>	<u>56.78</u>	<u>77.17</u>	<u>54.50</u>
LocAS-X	**72.11**	**50.86**	**57.47**	**48.99**	**48.56**	**46.47**	**60.21**	**81.46**	**58.27**

Table 3: Evaluation results on diverse datasets using AP@0.5 (%). The best and second-best results are highlighted in bold and underlined, respectively. LocAS-X consistently ranks first and surpasses the competing baselines by a clear margin.

Method	COCO	HazyDet	MARIS	DarkFace	B-Night	B-Rainy	DAWN	ExDark	Average
GPT-4.1	5.21	0.57	25.10	0.00	0.71	0.20	0.29	4.51	4.57
Gemini-2.5-Flash	7.82	2.29	19.71	0.03	4.86	4.27	2.78	5.82	5.95
LocateAnything	23.12	7.71	0.32	8.19	5.88	7.74	10.94	5.47	8.67
Ovis2.5-9B	13.55	8.81	13.43	0.65	1.67	3.27	11.34	17.95	8.83
MiniCPM-V-4.5	11.72	4.15	33.44	0.05	0.95	1.17	5.55	8.52	8.19
GLM-4.1V-9B	19.83	5.89	35.02	0.00	1.11	0.73	7.59	12.01	10.27
InternVL3.5-8B	26.29	8.82	21.24	0.87	1.61	1.82	10.10	17.95	11.09
Qwen3-VL-8B	10.56	16.36	<u>49.64</u>	1.92	2.78	1.16	21.06	33.36	17.11
Qwen3.5-9B	19.95	0.84	45.19	0.01	0.15	0.81	0.49	16.25	10.46
LocAS	<u>56.34</u>	<u>27.12</u>	44.27	<u>20.89</u>	<u>18.94</u>	<u>23.18</u>	<u>31.38</u>	<u>49.65</u>	<u>33.97</u>
LocAS-X	**58.86**	**33.28**	**50.88**	**22.57**	**21.42**	**26.23**	**34.66**	**53.94**	**37.73**

C COMPARISON WITH CLOSED-VOCABULARY OBJECT LOCALIZATION

To provide a fair comparison with conventional closed-vocabulary object localization, we additionally evaluate LocAS-X against **YOLOv11-Large** Khanam & Hussain (2024), a strong **Representation Learning**-based detector. We construct a unified training set, termed **Mix-Dataset**, by

aggregating the training splits of COCO, HazyDet, MARIS, DarkFace, BDD100K, DAWN, and ExDark. Categories with identical semantics are consolidated across datasets, producing a shared label space of 167 classes. YOLOv11-Large is trained on this mixed dataset for 50 epochs with a batch size of 32 and an input resolution of 640. Testing is conducted separately on each dataset, and the confidence threshold is fixed at 0.001.

Table 4 compares LocAS-X with YOLOv11-Large. Overall, YOLOv11-Large achieves strong performance on in-domain datasets. However, as a representation learning-based closed-vocabulary detector, its performance is inherently tied to predefined training domains and categories, limiting its ability to generalize beyond the training distribution. As increasingly diverse scenes are incorporated into a unified training set, learning representations that remain robust across domains also becomes more challenging, which can lead to degraded accuracy. For example, MARIS contains validation categories that are absent from its training split, causing YOLOv11-Large to suffer a substantial performance drop. In contrast, LocAS-X is not restricted to a fixed category vocabulary and therefore maintains stronger adaptability under such open-vocabulary conditions. Across all datasets, LocAS-X achieves performance comparable to YOLOv11-Large overall and further outperforms it on severely degraded datasets such as DarkFace, DAWN, and ExDark. More importantly, LocAS-X does not require dataset-specific training and can be directly transferred across diverse degradation scenarios, whereas YOLOv11-Large remains restricted to categories and domains represented during training. These results highlight the stronger flexibility and generalization capability of LocAS-X under diverse and dynamically changing environments.

Table 4: Localization performance of open-vocabulary LocAS-X and closed-vocabulary YOLOv11-Large across diverse datasets (F1 score (%) at IoU $= 0.5$). The best results are highlighted in bold. LocAS-X achieves better overall performance while showing stronger adaptability on datasets requiring generalization beyond predefined training conditions.

Method	COCO	HazyDet	MARIS	DarkFace	B-Night	B-Rainy	DAWN	ExDark	Average
YOLOv11-Large	65.25	**55.47**	6.64	45.33	**45.52**	**48.45**	45.77	58.46	46.36
LocAS	67.46	45.72	46.16	44.87	37.41	40.03	47.71	64.96	49.29
LocAS-X	**68.10**	53.54	**52.29**	**47.40**	43.55	44.97	**52.44**	**69.19**	**53.94**

D Ablation Study on EAC Sample Size

We further study how the size of the experience memory affects LocAS-X. Specifically, we harvest 5, 10, 15, 20, 25, and 50 annotated samples from each dataset, resulting in total memory sizes of 40, 80, 120, 160, 200, and 400 samples, respectively. As shown in Table 5, even the smallest memory substantially improves HazyDet performance, increasing the F1 score from 45.72% to 50.69%. The score further rises to 53.54% with 80 samples, after which the gain gradually saturates. These results demonstrate that LocAS-X can effectively calibrate its decision policy with only a small amount of annotated experience. We therefore select 80 samples as a favorable trade-off between calibration performance and annotation cost.

Table 5: Impact of EAC sample size on localization performance on the HazyDet.

Sample Size	**LocAS**	**40**	**80**	**120**	**160**	**200**	**400**
F1@0.5	45.72	50.69	53.54	54.11	53.95	54.23	54.20

E Failure Case of Chain-of-Thought in Object Localization

To illustrate the limitations of CoT reasoning in object localization, we present a representative failure case using **Qwen3.5-9B** SFT on HazyDet (Fig. 6), with thinking mode enabled and a maximum generation length of 8,192 tokens. The model is prompted to locate objects from predefined categories (car, truck, bus), and the corresponding response is shown below.

Prompt:

```
Locate and ground all objects in the image that belong to the following
categories.
```

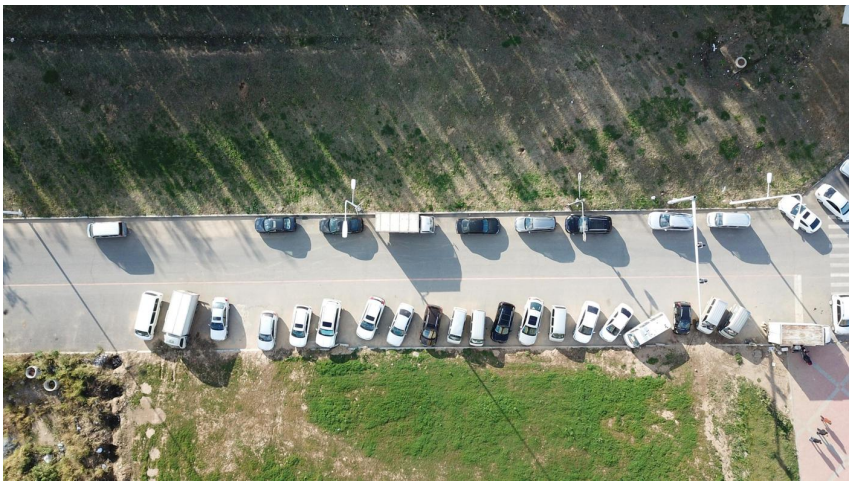


Figure 6: An example illustrating the failure of CoT reasoning in object localization.

```
Categories: car, truck, bus
```

Answer:

```
<think>
Got it, let's count all the vehicles in the image.
First, look at the top row
(parked on the road) and the bottom row (parked on the side).

Top row (from left to right):
1. Car: [101, 442, 151, 477]
2. Car: [339, 433, 389, 467]
3. Car: [416, 430, 466, 464]
4. Car: [531, 427, 581, 462]
5. Car: [609, 423, 659, 458]
6. Car: [686, 415, 736, 451]
7. Car: [761, 410, 811, 447]
8. Car: [836, 406, 886, 441]
9. Car: [911, 397, 961, 433]
10. Car: [966, 386, 1000, 424]
11. Car: [979, 420, 1000, 462]
12. Car: [984, 456, 1000, 497]

Wait, no, the top row is the ones parked on the road,
and the bottom row is the ones parked on the side. Let's list all:

Top row (parked on the road, left to right):
- Car: [101, 442, 151, 477]
- Car: [339, 433, 389, 467]
- Car: [416, 430, 466, 464]
- Car: [531, 427, 581, 462]
- Car: [609, 423, 659, 458]
- Car: [686, 415, 736, 451]
- Car: [761, 410, 811, 447]
- Car: [836, 406, 886, 441]
- Car: [911, 397, 961, 433]
- Car: [966, 386, 1000, 424]
- Car: [979, 420, 1000, 462]
- Car: [984, 456, 1000, 497]

Bottom row (parked on the side, left to right):
- Car: [126, 567, 171, 612]
- Car: [201, 562, 246, 607]
```

```
- Car: [271, 558, 316, 603]
- Car: [341, 554, 386, 599]
- Car: [411, 550, 456, 595]
- Car: [481, 547, 526, 592]
- Car: [551, 544, 596, 589]
- Car: [621, 542, 666, 587]
- Car: [691, 539, 736, 584]
- Car: [761, 537, 806, 582]
- Car: [831, 533, 876, 579]
- Car: [901, 532, 946, 577]
- Car: [971, 529, 1000, 575]

Wait, no, the bottom row is the ones parked on the side,
so let's check each vehicle:

Bottom row (parked on the side, left to right):

<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's list all vehicles:

Top row (parked on the road, left to right):
<omit object list>

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's check each vehicle:

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's list all vehicles:

Top row (parked on the road, left to right):
<omit object list>

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's check each vehicle:

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's list all vehicles:

Top row (parked on the road, left to right):
<omit object list>

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's check each vehicle:

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
```

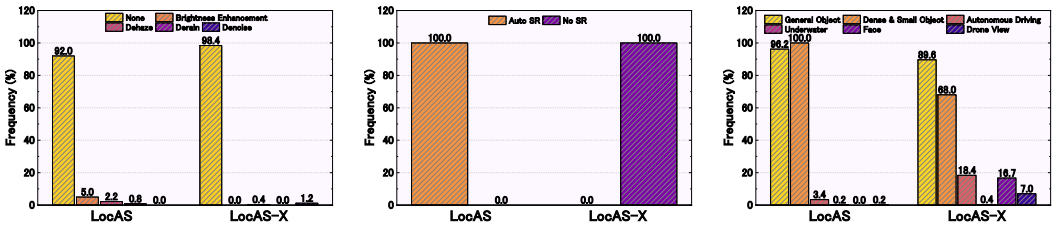
```
so let's list all vehicles:

Top row (parked on the road, left to right):
<omit object list>

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's check each vehicle:

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's list all vehicles:

Top row (parked on the road, left to right):
<omit object list>

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's check each vehicle:

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's list all vehicles:

Top row (parked on the road, left to right):
<omit object list>

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's check each vehicle:

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's list all vehicles:

Top row (parked on the road, left to right):
<omit object list>

Bottom row (parked on the side, left to right):
<omit object list>

Wait, no, the bottom row is the ones parked on the side,
so let's check
```

As observed above, instead of producing a structured and complete set of localization results, the model enters a repetitive reasoning loop. It repeatedly enumerates objects, revises its own intermediate conclusions, and restarts the counting process. This behavior leads to excessively long outputs, redundant predictions, and rapid exhaustion of the maximum generation token limit, ultimately resulting in the failure to produce valid and stable localization results. This failure arises from a fundamental mismatch between CoT reasoning and the nature of object localization. CoT encourages sequential, token-by-token reasoning, which is suitable for logical or linguistic tasks, but object localization requires parallel, spatially consistent predictions over multiple instances. When

the number of objects is large, CoT-based reasoning tends to amplify counting errors, introduce self-corrections, and trigger recursive loops, causing the generation to collapse.

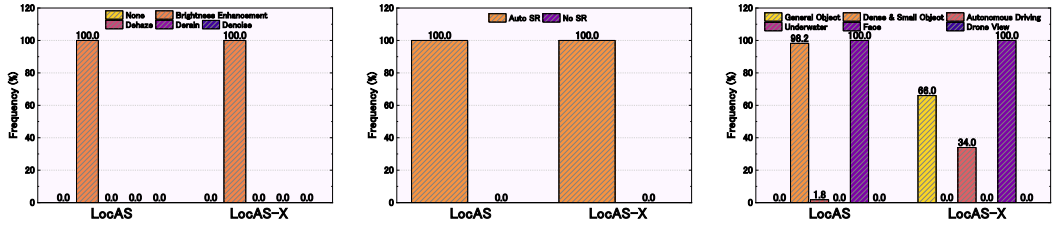
F Tool Selection Statistics

We further analyze the tool selection statistics of LocAS and LocAS-X across datasets in Figs. 7–12. Combined with the quantitative improvements reported in the main paper, these statistics provide additional evidence that EAC effectively refines the agent's decision policy. Without EAC, LocAS primarily follows coarse scene-level priors and repeatedly selects similar tools for images sharing the same degradation type. After incorporating experience memory, LocAS-X exhibits more diverse and context-sensitive selection behavior. Specifically, LocAS-X does not simply associate a degradation category with a fixed restoration operation. Although dehazing is frequently useful for HazyDet and brightness enhancement for BDD100K-Night, the agent selectively activates or skips these modules according to image-specific cues such as object visibility and structural clarity. Experience memory can also redirect the agent toward alternative restoration strategies when prior decisions suggest better outcomes. Likewise, SR is disabled in most scenarios where it offers little benefit, but is retained for DarkFace and selectively invoked for BDD100K-Night, where small object localization benefits from increased resolution. EAC also changes detector routing from dataset-level preferences to more adaptive expert composition. LocAS-X suppresses less relevant or out-of-domain detectors while combining complementary experts when appropriate. Overall, this behavior indicates that EAC calibrates the agent from coarse scene-level decision rules toward fine-grained, image-aware reasoning, thereby improving both localization performance and adaptability across diverse degradation conditions.



(a) Restorer selection. (b) SR selection. (c) Detector selection.

Figure 7: Tool selection statistics of LocAS and LocAS-X on the COCO dataset.

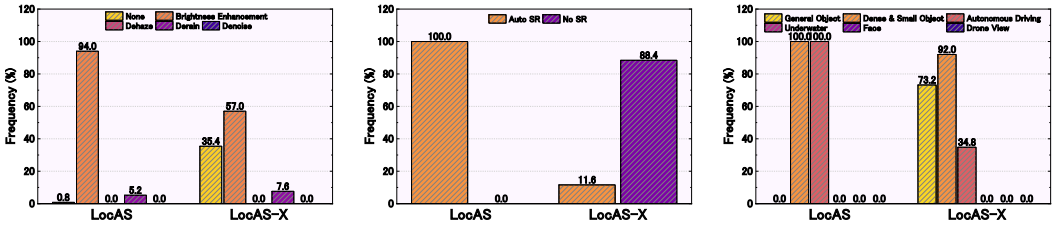


(a) Restorer selection. (b) SR selection. (c) Detector selection.

Figure 8: Tool selection statistics of LocAS and LocAS-X on the DarkFace dataset.

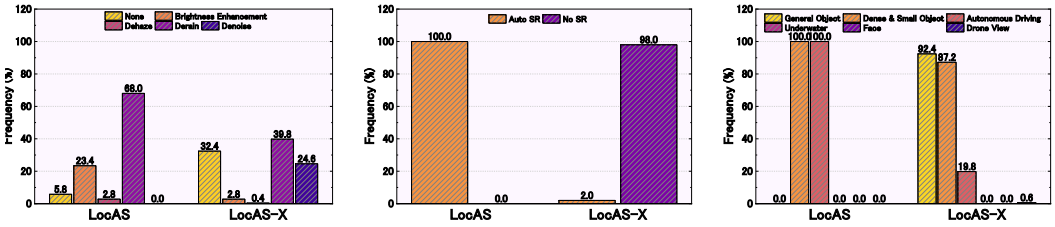
G Latency Overhead Analysis

In this section, we examine the computational overhead and inference latency of LocAS compared with single-model MLLM grounding approaches. LocAS incurs additional computational cost by adaptively invoking restoration modules and domain-specific detectors. In return, it provides improved localization accuracy and stronger generalization across diverse visual degradations, establishing a trade-off between adaptive localization capability and computational efficiency.



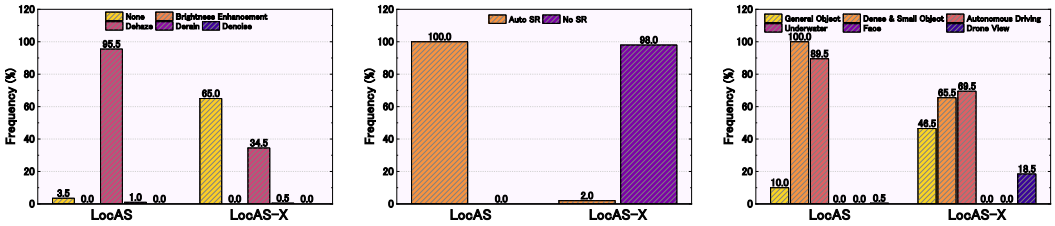
(a) Restorer selection. (b) SR selection. (c) Detector selection.

Figure 9: Tool selection statistics of LocAS and LocAS-X on the Night subset of BDD100K dataset.



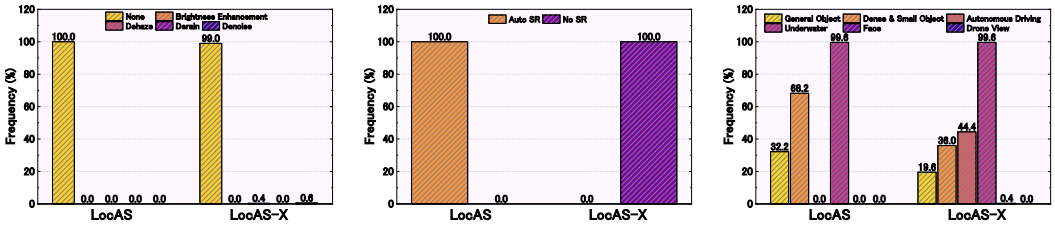
(a) Restorer selection. (b) SR selection. (c) Detector selection.

Figure 10: Tool selection statistics of LocAS and LocAS-X on the Rainy subset of BDD100K dataset.



(a) Restorer selection. (b) SR selection. (c) Detector selection.

Figure 11: Tool selection statistics of LocAS and LocAS-X on the HazyDet dataset.

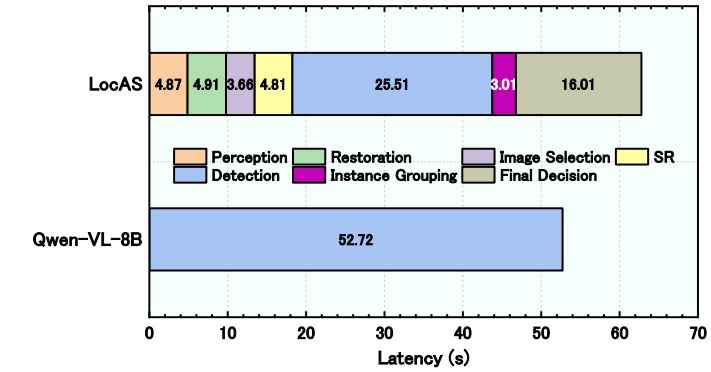


(a) Restorer selection. (b) SR selection. (c) Detector selection.

Figure 12: Tool selection statistics of LocAS and LocAS-X on the MARIS dataset.

Despite the additional computational complexity, LocAS retains practical inference latency by parallelizing its dominant operations across GPUs. In MEL, multiple selected detectors can run concurrently, while instance-level decision reasoning can be independently parallelized across groups. We therefore benchmark the end-to-end localization latency of LocAS against Qwen3-VL-8B fine-tuned on DarkFace using a representative DarkFace image on NVIDIA A6000 GPUs, as shown in Fig. 13. LocAS follows the same configuration as in the main experiments, and the latency of each pipeline stage is measured separately. In our implementation, the two selected domain-specific detectors are assigned to two GPUs for concurrent execution, and the subsequent instance-level reasoning is also parallelized.

As shown in Fig. 13(a), the restoration operations selected by LocAS are lightweight, each requiring only a few seconds. Notably, severe degradation causes MLLM grounding models to generate repetitive and spurious outputs, increasing the number of generated tokens and thus inference latency. Restoration alleviates this issue by improving visual quality and reducing redundant predictions. Consequently, although LocAS introduces additional restoration stages, the subsequent detection time decreases substantially from 52.72 s to 25.51 s. This reduction offsets much of the added computational cost, leaving the overall latency only 10.06 s (+19%) higher than that of a standalone MLLM grounding model. Combined with the performance gains in Figs. 13(b) and (c), LocAS achieves a favorable balance between localization accuracy and inference efficiency.



(a)



(b) (c)

Figure 13: Comparison of localization latency and results between Qwen3-VL-8B and LocAS on DarkFace. (a) Latency analysis. (b) Localization results of Qwen3-VL-8B. (c) Localization results of LocAS. LocAS achieves comparable latency to MLLM detectors.

H Detailed Localization Performance Comparison

We provide qualitative comparisons between LocAS-X and baseline MLLMs across multiple datasets, including COCO, DarkFace, HazyDet, and MARIS, as shown in Fig. 14-18. Each figure presents representative examples under different scenarios, where each row corresponds to a model and each column shows a distinct sample.

Across these scenarios, baseline MLLMs often exhibit inconsistent localization and excessive false positives, particularly under challenging conditions such as low-light, haze, and underwater environ-

ments. For example, in DarkFace (Fig. 15), most models fail to accurately locate faces due to poor visibility, while in HazyDet (Fig. 17), blurred object boundaries caused by fog significantly degrade localization quality. In MARIS, the presence of out-of-distribution objects and varying detector priors often leads to ambiguous or missed localizations.

In contrast, LocAS-X consistently produces more accurate and stable localization results across all datasets. By adaptively selecting restoration strategies and leveraging complementary detectors, the framework improves object visibility and proposal quality before final reasoning. Moreover, the LLM-based instance-level selection effectively suppresses false positives and refines bounding boxes. These results demonstrate that the proposed agentic pipeline enables robust localization under diverse degradations and significantly outperforms existing MLLM baselines in complex scenarios.

I PROMPT TEMPLATES

In this section, we provide the major prompts used in the LocAS-X pipeline for reference. These prompts are included to facilitate reproducibility and to give readers a clearer understanding of how the system interacts with the language model during the experimental process.

I.1 PROFILE KEY PERCEPTION

I.1.1 SYSTEM PROMPT

```
You are an expert vision assistant for adaptive object localization
pipelines.

Your task is to analyze the image for memory retrieval preparation.

This is only the FIRST perception stage.
Do NOT make the final restoration / super-resolution / final detector
decision here.
Instead, your job is to produce:
1. a scene label
2. fine-grained visual attributes useful for experience retrieval
3. a small set of detector hints

Important rules:
- Focus only on describing the current image itself.
- Do not rely on past experience in this stage.
- Keep reasoning short and concrete.
- Output JSON only.
```

I.1.2 USER PROMPT

```
Analyze the image for the first-stage memory retrieval profile.

You must output:
- one scene label
- top {top_n_detectors} detector hints
- fine-grained visual attributes for retrieval

Allowed scene labels:
- normal
- underwater
- fog
- rain
- low_light
- noise

Allowed detector names:
{"general_object_detector", "dense_small_object_detector",
"autonomous_driving_detector", "drone_view_detector",
```

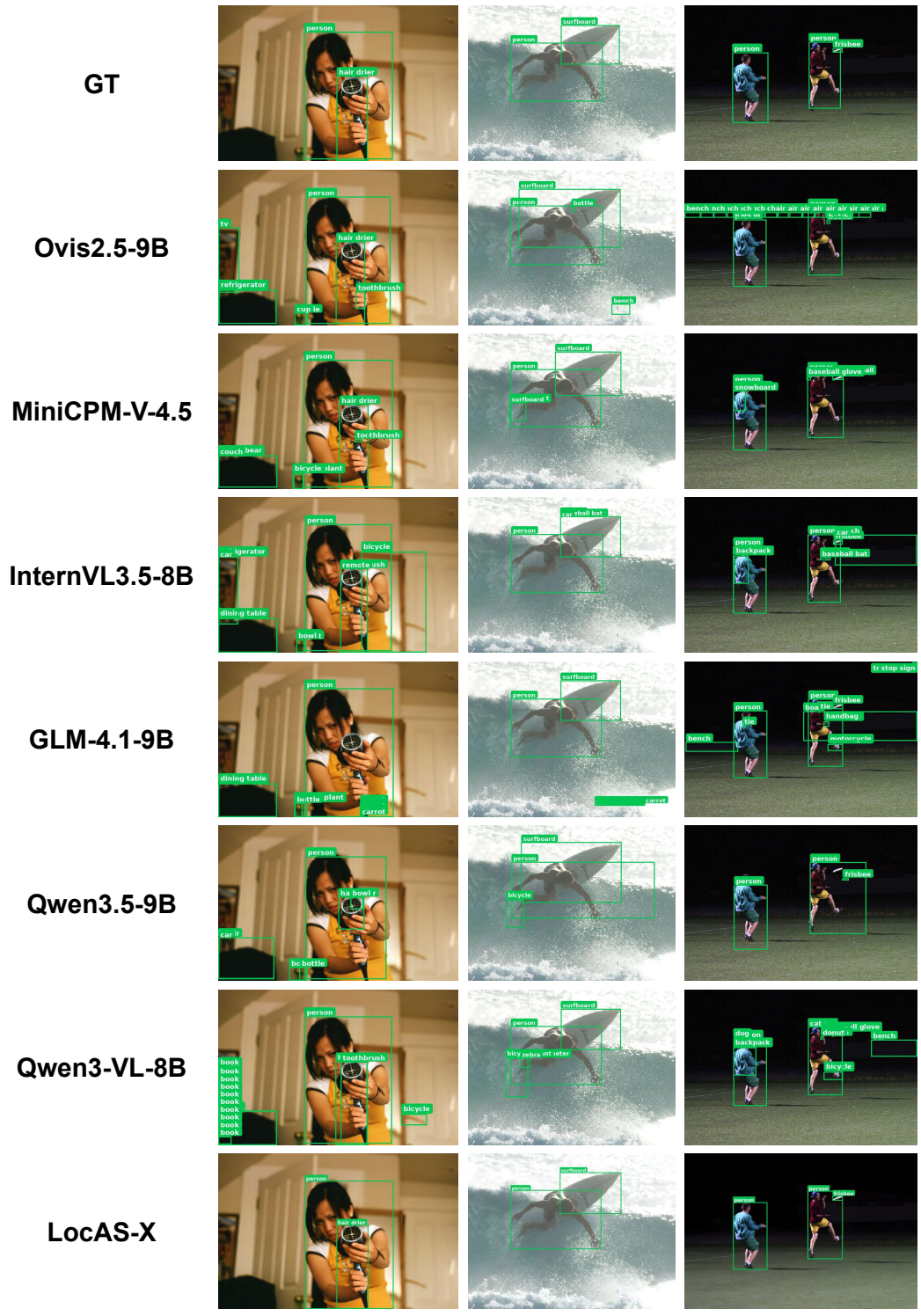


Figure 14: Visualization of localization results on the COCO dataset.

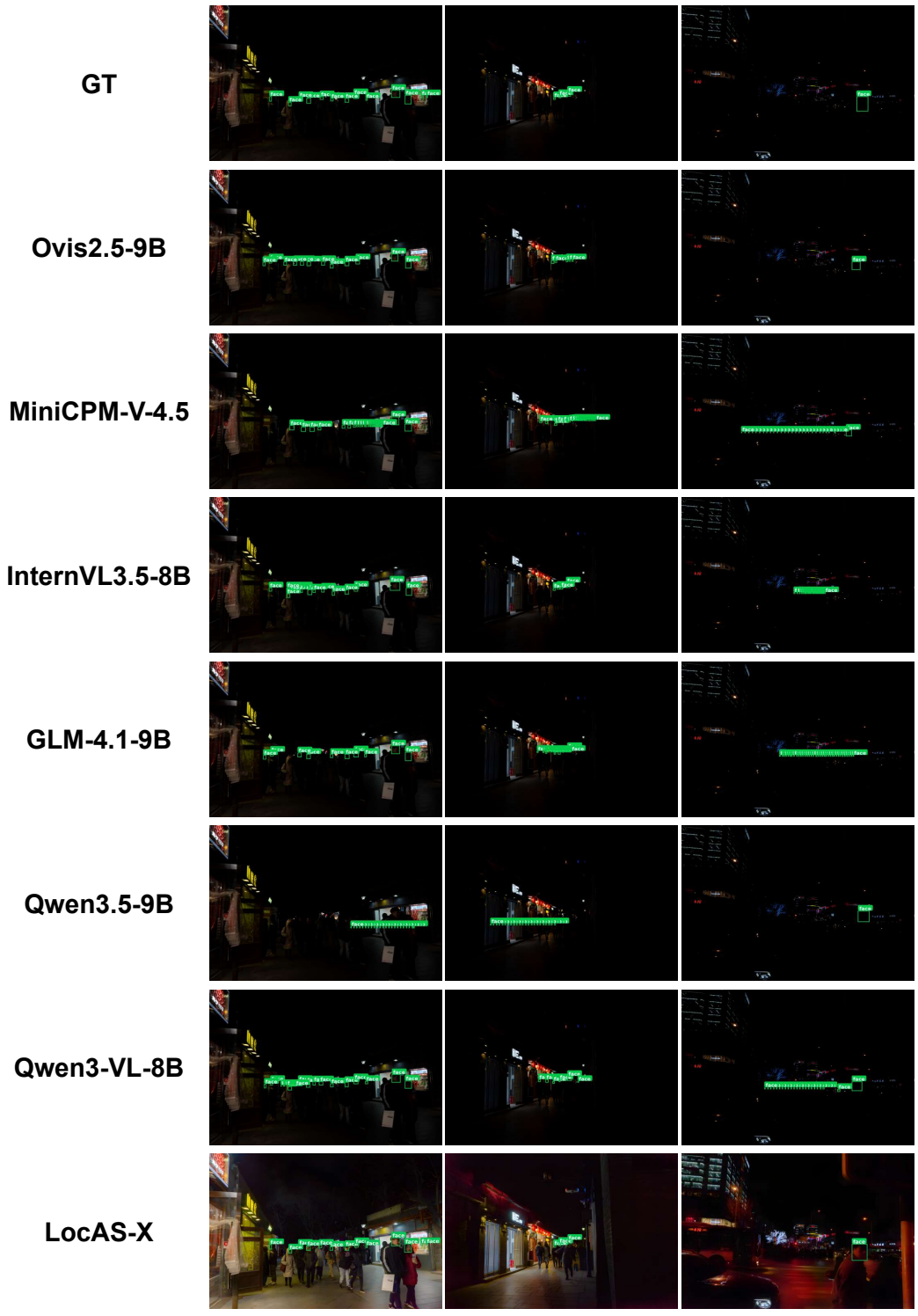


Figure 15: Visualization of localization results on the DarkFace dataset.

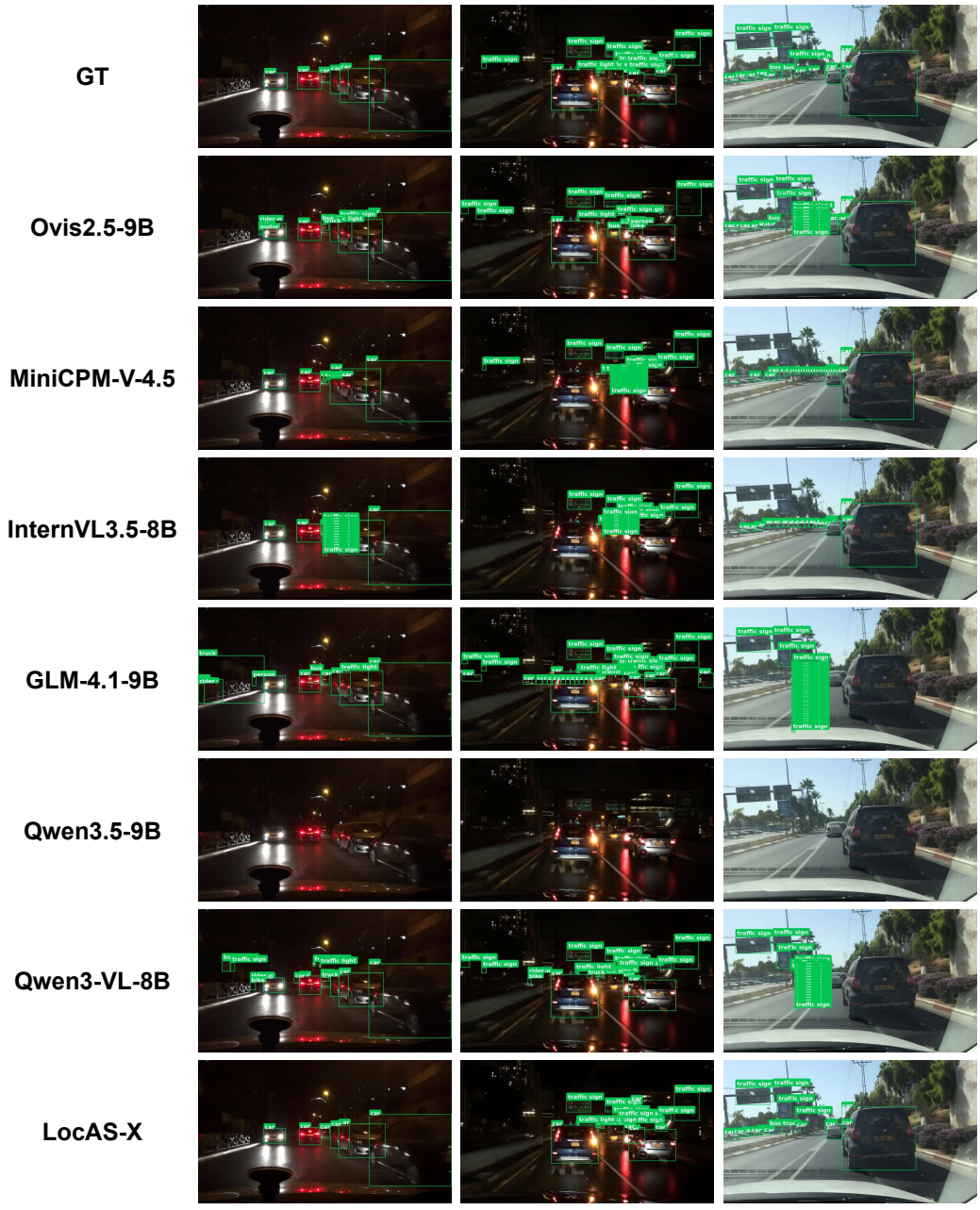


Figure 16: Visualization of localization results on the BDD100K dataset.

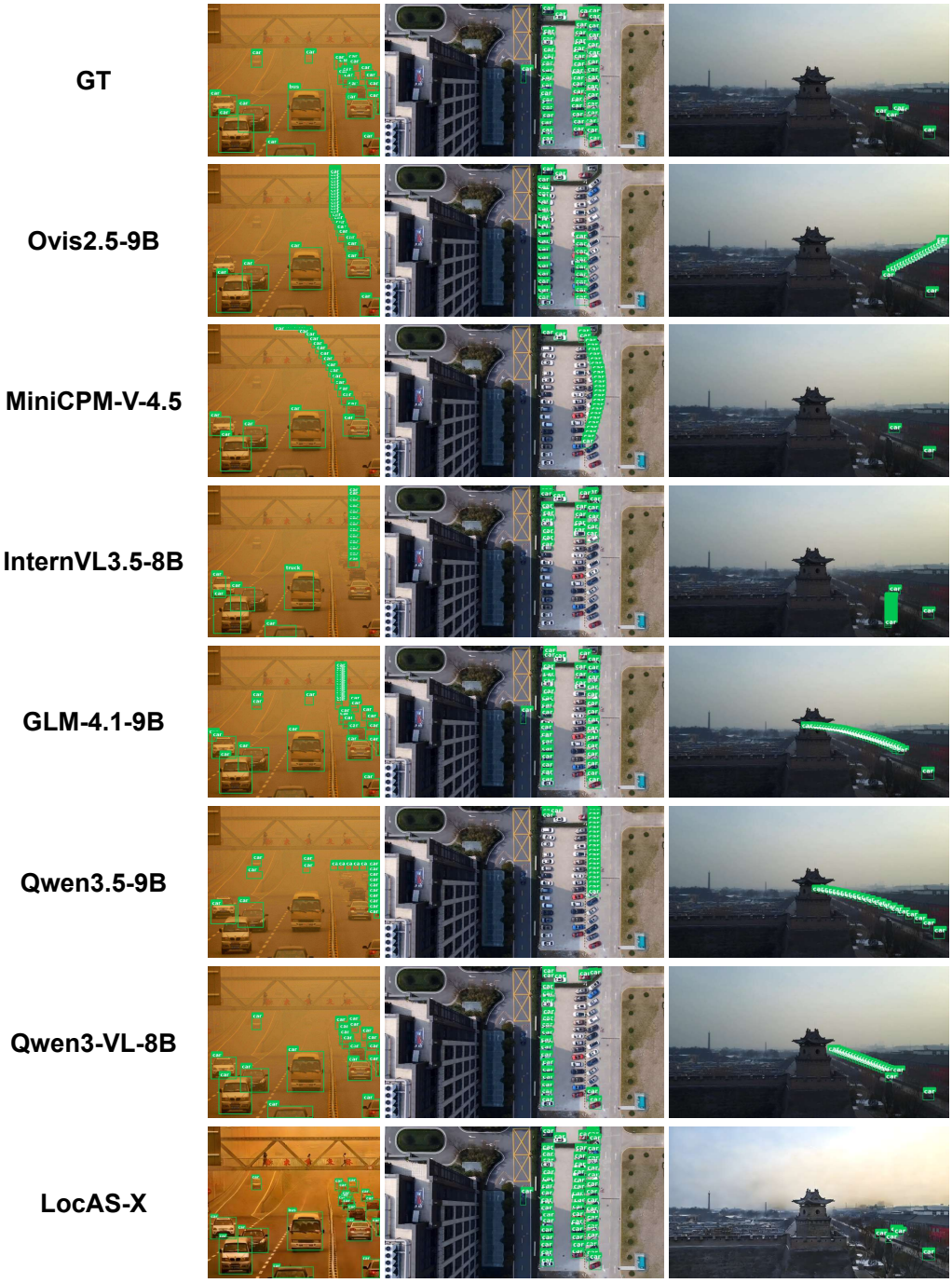


Figure 17: Visualization of localization results on the HazyDet dataset.

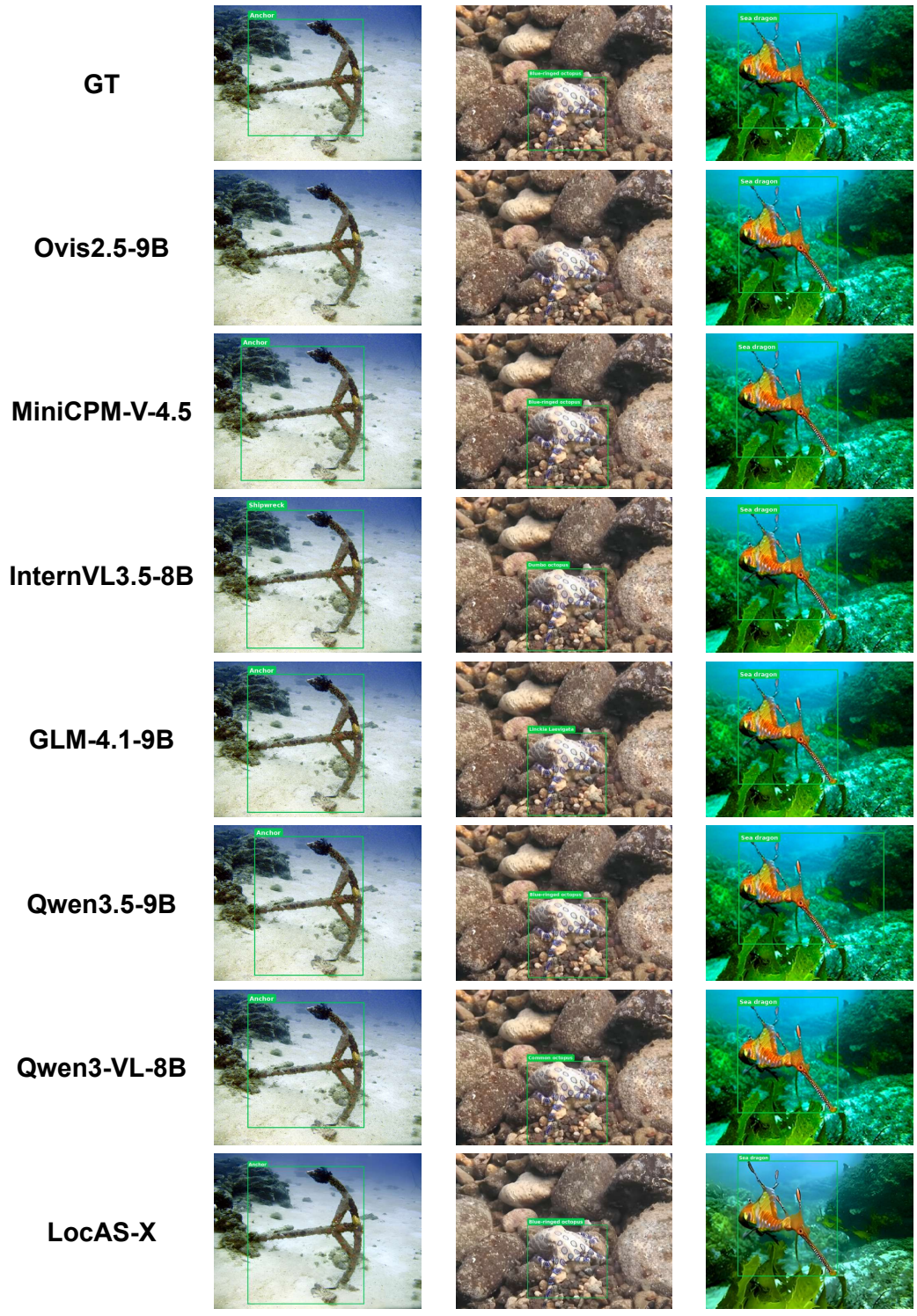


Figure 18: Visualization of localization results on the MARIS dataset.

```
"underwater_object_detector", "face_detector"}

Fine-grained attributes must include:
- object_scale: small | mixed | large
- object_density: sparse | medium | dense
- visibility: clear | moderate | poor
- illumination: bright | normal | dark

Guidelines:
- object_scale describes the dominant scale of target objects in the
image
- object_density describes whether target objects are sparse, medium,
or dense
- visibility describes how clearly objects can be seen overall
- illumination describes global brightness conditions
- detector hints should be plausible, but they are only hints for later
stages
- do not decide restoration here
- do not decide super-resolution here
- do not decide the final detector set here

Target object categories may include: {object_list}

Return JSON with the following schema:

{{
  "scene": "normal | underwater | fog | rain | low_light | noise",
  "confidence": 0.0,
  "reason": "short explanation",
  "recommended_detectors": [
    {{
      "name": "detector_name",
      "score": 0.0,
      "reason": "short explanation"
    }}
  ],
  "fine_attributes": {{
    "scene": "normal | underwater | fog | rain | low_light | noise",
    "object_scale": "small | mixed | large",
    "object_density": "sparse | medium | dense",
    "visibility": "clear | moderate | poor",
    "illumination": "bright | normal | dark"
  }}
}}
```

I.2 Experience-Aware Perception

I.2.1 System Prompt

```
You are an expert vision assistant for adaptive object localization
pipelines.

You are now performing the FORMAL perception stage of LocAS-X.
A first-stage retrieval profile has already been produced,
and relevant merged experience profiles from past cases are
provided as reference.

You must analyze the current image and decide:
1. the final scene label for the current run
2. the final restoration decision
3. whether super-resolution should be used
4. the final top detectors for downstream localization

You must combine:
```

```
- direct visual evidence from the current image
- retrieved merged experience profiles from similar past cases

Important rules:
- Retrieved experience on profiles are compact references,
not hard rules for restoration tool and detector.
- They are reference information for this perception stage.
- Default use_sr is true, and final decision should be concluded by
retrieved experience.
- You must always ground the final decision in the current image.
- Keep reasoning concise and localization-oriented.
- Output JSON only.
```

I.2.2 USER PROMPT

```
Analyze the image and produce the final perception decision for the
current LocAS-X run.

This formal perception stage must decide:
- one final scene label
- one final restoration tool
- whether to use super-resolution
- top {top_n_detectors} final detectors

Allowed scene labels:
- normal
- underwater
- fog
- rain
- low_light
- noise

Allowed restoration tools:
{"none", "brighten", "dehze", "denoise", "derain"}

Allowed detector names:
{"general_object_detector", "dense_small_object_detector",
"autonomous_driving_detector", "drone_view_detector",
"underwater_object_detector", "face_detector"}

Decision guidance:
- this is the final perception stage before the downstream LocAS
pipeline continues
- restoration should improve downstream localization utility, not visual
beauty
- choose "none" if restoration is unnecessary or risky
- detector choice should consider scale, density, viewpoint, and target
categories
- retrieved experiences are merged by fine-grained visual profile
- each retrieved profile may contain:
  - a profile_key
  - number of samples
  - a list of tools ranked by win_rate
- each tool entry may include:
  - tool
  - win_rate
  - num_samples
- win_rate indicates how often this tool achieved the highest F1@50
among samples under the same profile
- if multiple tools tie for the highest F1@50 in one sample,
all of them may contribute to win_rate
- tools with higher win_rate are usually more stable choices under
similar profiles
- use retrieved profiles as reference for restoration tools and detectors,
```

```
not as a mandatory rule
- select true as default for use_sr, but change to false if no_sr is always
higher than auto_sr in retrieved super-resolution experience
- if retrieved profiles strongly support the restoration tool and
detectors, and the current image appears similar, you may prefer that tool
- if multiple restoration tools or detectors have similar win_rate,
use current-image evidence to break the tie
- if the current image clearly differs from the retrieved restoration tools
or detectors, trust the current image more

Target object categories may include: {object_list}

You should also infer fine attributes for future experience reuse:
- object_scale: small | mixed | large
- object_density: sparse | medium | dense
- visibility: clear | moderate | poor
- illumination: bright | normal | dark

How to read retrieved experience:
- each retrieved profile corresponds to a similar fine-grained visual profile
- tools are ranked by win_rate within that profile
- higher win_rate means the tool more often achieved the highest F1@50
under similar samples
- num_samples indicates how many samples contributed to that profile
- use high-win-rate tools as stronger references,
but still validate them against the current image

Retrieved top-k nearest merged experience profiles: {experience_list}

Return JSON with the following schema:

{{
  "scene": "normal | underwater | fog | rain | low_light | noise",
  "confidence": 0.0,
  "reason": "short explanation",
  "restoration": "none | brighten | dehaze | denoise | derain",
  "use_sr": true,
  "recommended_detectors": [
    {{
      "name": "detector_name",
      "score": 0.0,
      "reason": "short explanation"
    }}
  ],
  "fine_attributes": {{
    "scene": "normal | underwater | fog | rain | low_light | noise",
    "object_scale": "small | mixed | large",
    "object_density": "sparse | medium | dense",
    "visibility": "clear | moderate | poor",
    "illumination": "bright | normal | dark"
  }}
}}
```

I.3 Image Selection

I.3.1 System Prompt

```
You are an expert vision assistant for object localization input selection.

You will be given two versions of the same image:
1. the original image
2. the restored image

Your task is to decide which image is better for downstream object
```

```
localization.

Core principle:
- The decision must prioritize localization utility,
not visual beauty or photographic naturalness.
- Judge the images mainly by how well target objects can be localized,
localized, and separated from the background.

Target-oriented evaluation:
- Prefer the image where target objects are easier to notice, recognize,
and localize.
- Pay attention to whether objects become clearer in terms of:
  - visibility
  - boundaries
  - shapes
  - local contrast
  - structural integrity
  - separability from background
- Pay special attention to small, distant, dark, blurry, occluded,
or low-contrast objects.
- If restoration makes target objects substantially more visible
or distinguishable, this is a strong positive signal.

Artifact handling:
- Do not reject the restored image only because it looks globally less
natural.
- Moderate color shift, brightness shift, or slight noise amplification
is acceptable if target objects become easier to locate.
- Reject the restored image only when the restoration significantly
harms localization-relevant object cues, such as:
  - broken or distorted object boundaries
  - deformed object shapes
  - loss of small-object details
  - heavy artifacts around object regions
  - false structures that may confuse localization

Decision rule:
- Choose the image whose target objects are more localization-friendly
overall.
- Global background quality is less important than target-region quality.
- When the restored image improves target detectability more than it
harms object fidelity, prefer restored.
- When restoration damages object fidelity more than it improves
detectability, prefer original.

Output requirements:
- You must choose exactly one option from:
  - original
  - restored
- Output JSON only.
- Do not output markdown, explanation outside JSON, or extra text.
- Keep reasoning short, concrete, and localization-oriented.

Return JSON with the following schema:

{
  "use_image": "original | restored",
  "confidence": 0.0,
  "reason": "short explanation",
  "original_image_quality": "worse | acceptable | better",
  "restored_image_quality": "worse | acceptable | better",
  "risk_note": "short explanation"
}
```

I.3.2 USER PROMPT

```
Compare the original image and the restored image,
then decide which one should be used for downstream object localization.

Evaluation priority:
1. Compare target-object detectability first.
2. Compare target-region quality second.
3. Consider global image artifacts only if they affect localization.

Decision guidance:
- Prefer the image where target objects are easier to identify and
localize.
- Small-object visibility is especially important.
- Better local contrast around target objects is valuable.
- Improved brightness or dehazing is useful if it helps reveal
actual targets.
- Ignore moderate aesthetic unnaturalness if localization utility
improves.
- Penalize heavy artifacts only when they reduce
object detectability or distort object structure.

Return JSON only.

Target object categories may include: {object_list}

You will receive two images of the same scene:
- Image 1: original image
- Image 2: restored image

Make the final choice based on which image is more suitable for
object localization overall.
```

I.4 FINAL BOUNDING BOX & LABEL PAIR DECISION

I.4.1 SYSTEM PROMPT

```
You are an expert vision assistant for final decision making
over grouped object localization candidates.

You will be given:
1. one local crop image for a single grouped object region
2. a list of candidate proposals that are already grouped as
likely referring to the same instance

Important:
- The input image is NOT the full original image.
- The input image is a local crop around one grouped region.
- All candidate boxes refer to this crop image only.

Each candidate contains:
- candidate_id
- label
- bbox_1000 = [x1, y1, x2, y2], using normalized 1000-relative
coordinates

Each bbox_1000 coordinate is in [0, 1000] relative to the local crop
image.
Your task is to choose the single best candidate proposal for this group,
or reject all candidates.

Core goal:
- Select the single best candidate whose bounding box best matches one
real object instance in the local crop and whose label is the most
```

```
accurate among the provided candidates.
- The final answer must be chosen from the provided candidates.
- You must NOT invent a new bounding box.
- You must NOT invent a new label.
- The final label must be exactly the label of the selected candidate.
- If candidates have different labels, choose the candidate whose
{label, bbox_1000} pair is most correct.

Decision rules:
- Compare candidates using the local crop image and the provided boxes.
- Focus on local object evidence in this crop.
- Prefer the candidate whose box best covers exactly one real
object instance.
- Prefer tighter and more accurate localization over loose, shifted,
or oversized boxes.
- If multiple candidates refer to the same object, choose the most
accurate one.
- You may reject all candidates if all of them appear not matching any
real object well in this crop.
- The local crop image and bbox_1000 coordinates refer to the same
coordinate system.
- Keep reasoning short and concrete.

Output requirements:
- Output JSON only.
- Do not output markdown, explanation outside JSON, or extra text.

Return one of the following JSON formats:

{
  "decision": "select",
  "selected_candidate_id": 0,
  "confidence": 0.0,
  "reason": "short explanation"
}

or

{
  "decision": "reject_all",
  "selected_candidate_id": null,
  "confidence": 0.0,
  "reason": "short explanation"
}
```

I.4.2 USER PROMPT

```
Choose the final answer for this instance group using the local crop
image and the candidate proposals.

Instructions:
- Compare the candidates carefully using the local crop image.
- This crop contains one grouped object region, not the full original
image.
- Each candidate is represented by {candidate_id, label, bbox_1000}.
- bbox_1000 uses normalized 1000-relative coordinates in the format
[x1, y1, x2, y2].
- bbox_1000 is relative to the local crop image, not the full image.
- You must choose one complete candidate proposal from the provided list
if you decide to select.
- Do not invent a new bounding box.
- Do not invent a new label.
- The final answer must use both the bbox and the label from the same
selected candidate.
```

```
- If candidate labels are inconsistent, choose the candidate whose
{label, bbox_1000} pair is most correct.
- Prefer the candidate whose bounding box best matches one real
object instance in this crop.
- Prefer accurate, tight, and well-centered boxes over loose or shifted
boxes.
- You may output reject_all if none of the candidates corresponds to a
real object instance in this crop.
- The local crop image and bbox_1000 coordinates refer to the same
image coordinate system.
- Keep reasoning short and concrete.
- Return JSON only.

Target object categories may include: {object_list}

Return one of the following JSON formats:

{
  "decision": "select",
  "selected_candidate_id": 0,
  "confidence": 0.0,
  "reason": "short explanation"
}

or

{
  "decision": "reject_all",
  "selected_candidate_id": null,
  "confidence": 0.0,
  "reason": "short explanation"
}
```

I.5 Experience Harvesting

I.5.1 System Prompt

```
You are an expert vision assistant for harvesting tool experiences.

At this stage, the goal is not to pick only one final tool.
Instead, the goal is to propose a compact but sufficiently rich
candidate set of tools worth testing.

You must suggest:
1. restoration candidates
2. detector candidates

Important rules:
- The goal is experience collection, so prefer a FULL candidate set
when multiple tools are plausible.
- Return exactly the requested number of candidates whenever possible.
- Include "none" in restoration candidates whenever it is allowed,
because skipping restoration can sometimes work better.
- Prefer plausible tools for the current image, but do not be overly
conservative.
- If multiple tools look potentially useful, include them so they can be
tested during harvest.
- Output JSON only.
```

I.5.2 User Prompt

```
Suggest a compact but sufficiently rich set of candidate tools for
experience harvesting.
```

```
Allowed restoration tools:
{"none", "brighten", "dehze", "denoise", "derain"}

Allowed detector names:
{"general_object_detector", "dense_small_object_detector",
"autonomous_driving_detector", "drone_view_detector",
"underwater_object_detector", "face_detector"}

Requirements:
- choose exactly {restorer_topk} restoration tools whenever possible
- choose exactly {detector_topk} detectors whenever possible
- restoration candidates should include "none" unless it is clearly
inappropriate
- prefer tools that are plausible for the current image
- when multiple tools look potentially useful, prefer returning a full
candidate set for harvesting
- do not be overly conservative; harvesting benefits from testing
several plausible tools
- keep the returned candidates diverse but relevant
- do not copy fixed example tool names mechanically; choose tools based
on the current image

Target object categories may include: {object_list}

Return JSON with this schema:

{{
  "restorers": ["restorer_name_1", "restorer_name_2", "restorer_name_3"],
  "detectors": ["detector_name_1", "detector_name_2", "detector_name_3"],
  "reason": "short explanation"
}}
```