buildings

MDPI

Article

RTR-Align: Rule Token-to-Region Alignment for Construction Safety Violation Recognition and Localization

Feifei Tao, Zanbo Gao * and Shuai Xu

College of Computer Science and Software Engineering, Hohai University, Nanjing 210098, China; tff@hhu.edu.cn (F.T.); 251607040065@hhu.edu.cn (S.X.)
* Correspondence: zbgao@hhu.edu.cn

Abstract

Construction safety inspections require identifying violated rules and localizing the associated workers or hazardous areas. Although multimodal large language models can describe hazards, supervision based solely on structured answers does not explicitly constrain the correspondence between rule records and visual targets. This study proposes RTR-Align, a Rule Token-to-Region Alignment method for construction safety violation recognition and localization. The method links the specific occurrence position of the Rule Token in each violation record to the corresponding target region, imposing separate spatial constraints on different instances associated with the same rule. It jointly learns rule assessment and target localization during LoRA adaptation of Qwen3-VL-8B-Instruct. The study uses the ConstructionSite10k dataset, which contains 10,013 construction site images with violation question-answering and localization annotations, and produces instance-level annotations for training and evaluation by separating grouped instances, adjusting bounding-box boundaries, and adding annotations for missed targets. In the main run, RTR-Align achieved an F1 score of 62.72% and a total intersection over union (Total IoU) of 39.40% on the 3004 test images, exceeding the base model by 48.11 and 32.55 percentage points, respectively. RTR-Align produced fewer false positives than the base model, but its recall was 1.19 percentage points lower. Analyses across rules and scenarios, along with parameter and qualitative analyses, revealed performance differences and remaining challenges. These results support joint learning of violation recognition and rule-associated localization, enabling the model to identify relevant targets with fewer false positives and provide spatial evidence for manual verification.

Keywords: construction safety; multimodal large language models; visual grounding; parameter-efficient fine-tuning

Check for updates

Academic Editor: Salvatore Verre

Received: 24 September 2026
Revised: 3 November 2026
Accepted: 5 November 2026
Published: 12 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

In recent years, the application of computational technologies to engineering monitoring, assessment, and decision support has continued to advance [1–4]. Construction safety inspection with multimodal large language models (MLLMs) aims to interpret construction images and natural-language safety rules jointly. Equipped with visual understanding and language reasoning capabilities, an MLLM-based inspector is expected to assess safety compliance from an image and explain its judgment in natural language.

Recent studies have demonstrated promising capabilities in generating rule-aware hazard assessments [5], supporting inspection tasks whose outputs consist solely of scene-level judgments and textual explanations. However, these capabilities do not ensure reliable

answers to instance-level questions about which worker violates a rule and where that worker is located [6]. Consider a crowded scene in which a worker without a hard hat stands between two workers wearing hard hats. To answer "Which worker violates the hard-hat rule?", the model must not only recognize the absence of a hard hat, but also associate this observation with the correct person and localize that person's visible body without including the neighboring workers. A scene-level answer such as "a worker is not wearing a hard hat" leaves this correspondence unresolved, making it difficult to verify whether the judgment refers to a genuine violation or a falsely accused worker. Therefore, explicitly linking rule-specific visual evidence to the corresponding person and spatial region is essential for verifiable MLLM-based construction safety inspection.

To investigate this correspondence, Chen and Zou [6] introduce ConstructionSite10k and evaluate pretrained vision-language models (VLMs) on safety violation question answering and visual grounding. Their framework extends inspection from reporting violations to requesting their locations, but the evaluation reveals that recognizing a violation does not ensure accurate localization of the associated region. To provide explicit spatial guidance, Adil et al. [7] use YOLOv11n to locate workers and machinery and incorporate the detected entities into structured prompts. This approach improves hazard assessment by providing object-level context, but relies on a separate detector to supply the spatial information used in subsequent reasoning. Moreover, detecting a worker does not establish whether that worker is responsible for the violation described by the model. Moving toward internally generated spatial evidence, Luo et al. [8] propose VEST, which uses reinforcement learning to generate hazard-related coordinates within reasoning sequences. Its image-level supervision and semantic rewards enable grounded reasoning without annotated bounding boxes. However, the supervision specifies the hazard category and its existence without providing an explicit target region for each rule-associated instance. In fact, the same rule may apply to several workers, while neighboring workers may have different compliance states. A correct image-level judgment therefore does not determine which individual should be associated with each occurrence of a rule in the answer, leaving the correspondence between rule records and specific visual regions implicit.

In this work, we propose RTR-Align, a training strategy that explicitly aligns rule tokens with visual regions for the recognition and localization of construction safety violations. Each violation is represented as a structured record containing a rule identifier, a bounding box, and an explanation. The rule token anchors the record to its corresponding visual target, with separate token positions distinguishing instances associated with the same rule. Through LoRA adaptation, we jointly optimize full-answer generation and regional attention alignment: the former teaches the model to generate structured violation records, while the latter aligns each rule token's visual attention with a target derived from its annotated bounding box. Regional supervision is used only during training, allowing the adapted model to generate violation records directly from images and textual rule prompts, without an additional detector or segmenter at inference time.

To implement the above rule–region constraints as effective instance-level supervision, data annotations must also accurately distinguish individual targets and delineate their spatial extents. ConstructionSite10k [6] provides a data foundation for construction safety violation question answering and visual grounding. However, some original annotations enclose multiple workers within a single bounding box, making it difficult to associate violated rules with specific workers. In addition, some bounding boxes fit their targets loosely and include adjacent workers or substantial irrelevant background, resulting in discrepancies between the supervised regions and the actual target extents, as shown in Figure 1. The former merges visual evidence from different workers, while the latter includes irrelevant regions in the supervision. Both hinder the model from learning clear

rule–instance correspondences. Therefore, we refine group-level annotations into individual instance annotations, improve the tightness of bounding boxes, and add annotations for missed targets, making ConstructionSite10k more suitable for training and evaluation at the level of rule-associated instances. Building on this dataset, this study adopts Qwen3-VL-8B-Instruct as the base model for adaptation in RTR-Align. The model supports vision-language understanding, instruction following, and two-dimensional localization, allowing it to express rule assessments, target locations, and associated explanations in a unified output. Its open weights and accessible decoder representations also facilitate region supervision for Rule Tokens and parameter-efficient adaptation. In subsequent evaluations, the unadapted Qwen3-VL-8B-Instruct model (denoted as Qwen3-VL-8B) serves as a direct reference based on the same underlying model. In contrast, other general-purpose vision-language models are used to compare model performance on the construction safety inspection task. The region loss ablation further examines the contribution of explicit correspondence supervision beyond full-answer supervision. This setup allows the overall effects of adaptation and the gains from rule–region constraints to be evaluated separately, with detailed experimental results presented in Section 5.

The main contributions of this work are as follows:

1. We identify limitations in existing dataset annotations and systematically refine ConstructionSite10k to address bounding boxes enclosing multiple workers, inaccurate boundaries, and omitted targets. These refinements establish explicit correspondences between safety rules and individual workers or hazardous regions, providing an instance-level supervision basis for violation recognition and localization.
2. We propose RTR-Align, a novel training strategy that uses rule tokens in structured outputs as supervision anchors and explicitly associates their visual attention with the corresponding bounding-box regions. This strategy jointly constrains rule identification and target localization without requiring an additional detector or regional supervision branch at inference.
3. We systematically evaluate RTR-Align across multiple dimensions, including model comparisons, safety rules, camera distances, illumination conditions, and training parameters, providing insights into its recognition and localization performance, scene adaptability, and practical applicability.

2. Related Work

2.1. Construction Safety Violation Recognition

Engineering safety monitoring draws on sensor measurements to assess structural condition and on site images to evaluate workers' use of personal protective equipment (PPE) [9–12]. In image-based construction safety inspection, the visual evidence needed to identify a violation depends on the rule being assessed. For example, Fang et al. [13] detected workers without hard hats in far-field surveillance videos. Assessing a worker's exposure to an excavator hazard requires additional context, including the machine's operating state and the worker's position relative to its operating radius.

Visual question answering (VQA) frames inspections as natural-language questions. Ding et al. [14] used such questions to assess image-level compliance without localizing violations. Later methods use different combinations of domain adaptation, safety ontologies, retrieval, and structured reasoning to improve scene understanding and compliance analysis [15–17]. Hazard reporting and conversational assessment further provide risk explanations [3,18], but answers and narratives alone do not necessarily link each judgment to a specific image region.

Construction systems combine VQA or hazard reasoning with detection or language-conditioned segmentation, but associations between judgments and regions depend on

information transfer between modules [19–21]. Construction-oriented VLMs and benchmarks also evaluate bounding-box outputs alongside safety judgments [6,22]. AVA-VLM uses adaptive local cropping for PPE violation recognition and localization, with a single target box enclosing all affected workers in each image that contains a PPE violation [23]. RTR-Align focuses on reliable associations between individual violation records and visual instances in multi-instance scenes. During generative VLM training, it uses each record's rule token, representing its rule identifier, as an alignment anchor to supervise this correspondence. The adapted model generates rule-associated boxes and explanations without an additional detector or segmenter at inference.

2.2. Visual Understanding Based on Multimodal Large Language Models

Multimodal large language models combine visual information with language instructions, providing a general foundation for construction safety inspection based on natural language rules. LLaVA-OneVision-2 focuses on unified perception and reasoning for images and videos. This study uses its 8B instruction-tuned model (LLaVA-OV-2-8B) to examine the transfer of general visual understanding to rule checking [24]. InternVL3.5-8B supports visual question answering, multimodal reasoning, and visual grounding, and can be used to compare object understanding and spatial representation capabilities [25]. GLM-4.6V-Flash is a lightweight vision-language model designed for efficient deployment, providing image understanding and multimodal reasoning capabilities [26]. GPT-5.6 Sol supports image input, text output, and structured output. This study uses its high reasoning configuration (denoted as GPT-5.6 Sol high) as a closed-source general-purpose reference model [27].

Qwen3-VL-8B-Instruct also supports vision-language understanding and 2D localization and provides open weights, making it suitable for supervision of internal representations and domain adaptation [28]. These models share the ability to generate responses based on images and language instructions. We compare their performance on rule-associated instance recognition and localization. RTR-Align uses Qwen3-VL-8B-Instruct as its backbone to investigate how the correspondence between rule records and spatial evidence can be explicitly incorporated into learning. Its contribution therefore also needs to be assessed through controlled comparisons using the same backbone and the same answer supervision.

2.3. Spatial Grounding in Vision-Language Models

Spatial grounding in vision-language models aims to associate linguistic expressions with corresponding image regions. Image–text contrastive learning and joint Transformer encoding provide a representational foundation for language-guided visual understanding [29,30]. When these representations are applied to construction safety inspection, correspondences still need to be established between specific safety judgments and the associated visual instances. Explicit grounding can also be achieved through language-conditioned detection or promptable segmentation, with Grounding DINO and Segment Anything being representative methods [31,32]. These two types of methods provide bounding boxes and masks, respectively, which can serve as spatial evidence within larger reasoning systems.

Generative grounding methods encode spatial representations through numerical coordinate strings, discrete location tokens, or combinations of coordinates and continuous region features [33–35]. Grounding annotations link expressions to regions, enabling multimodal models to generate bounding boxes without a separate detector at inference. LISA and GLaMM predict pixel-level masks associated with textual requests or responses [36,37], providing explicit spatial evidence beyond textual answers. Although these methods support language-guided grounding, reasoning segmentation, and region-grounded dialogue,

they do not specifically address construction safety rule assessment or correspondence between multiple violation records and specific instances.

Table 1 summarizes the above studies on construction safety inspection and general visual grounding, comparing their main model components, tasks, and spatial localization mechanisms. Construction safety methods support safety analysis through domain adaptation, collaboration between perception modules, semantic verification, or local visual inspection. The spatial information used includes site region labels, bounding boxes, and segmentation masks. General grounding methods associate linguistic expressions with image regions through numerical coordinates, discrete location tokens, or mask decoding. In the multi-instance scenarios considered in this study, the same rule may correspond to multiple targets, and the same target may be associated with multiple rules. Therefore, RTR-Align introduces a rule-to-region alignment objective in addition to the training objective for structured answer generation. Using the Rule Token representing the rule identifier in each violation record as an anchor, it uses instance-level bounding boxes to supervise the token's attention response to the corresponding visual region. This objective aims to reduce mismatches between rule assessments and visual instances and improve consistency between violation recognition and localization.

Table 1. Vision-language methods for construction safety and visual grounding.

Reference	Model	Task	Localization Mechanism
Tran et al. [20]	ViLT + CLIPSeg + SAM	Safety VQA; referring expression segmentation	Points selected from the CLIPSeg output prompt SAM to refine the segmentation.
Cheng et al. [21]	YOLOv10 + CLIP + DeepSeek-R1-Distill (7B) + KG-RAG	Worker detection; scene understanding; hazard reasoning	Detects workers and expands their boxes for contextual crops.
Wang et al. [22]	GPT-4o/LLaVA-13B + retrieval + CCV	Captioning; violation and object detection/localization	Generates boxes; CCV checks consistency between captions and predicted categories.
Kim et al. [23]	Qwen2.5-VL-7B + LoRA + adaptive cropping	PPE violation detection; object detection; captioning	Uses optional local crops. For images with PPE violations, one target box encloses all workers with violations.
Chen et al. [33]	CLIP + Vicuna-7B/13B	Referential dialogue; grounding; region QA	Generates boxes as numerical coordinate strings.
Peng et al. [34]	Kosmos-2	Phrase grounding; referring expression comprehension/generation	Links text spans to boxes encoded by discrete location tokens.
Rasheed et al. [37]	CLIP + Vicuna-7B + SAM	Grounded dialogue; referring segmentation; region captioning	Decodes masks from segmentation tokens to ground generated phrases.

Abbreviations. Vision-and-Language Transformer (ViLT), Contrastive Language–Image Pre-training (CLIP), Segment Anything Model (SAM), visual question answering (VQA), question answering (QA), knowledge graph-based retrieval-augmented generation (KG-RAG), collaborative cross-verification (CCV), low-rank adaptation (LoRA), and personal protective equipment (PPE).

3. Methodology

3.1. Dataset Annotation Optimization

This study investigates visual question answering for construction safety and the localization of objects involved in safety violations using the ConstructionSite10k dataset [6].

The dataset provides construction-site images annotated with safety-rule violations, corresponding explanations, and object bounding boxes. We focus on four safety rules concerning personal protective equipment, use of safety harnesses, edge protection, and human–machine interaction risks, as defined in Table 2. Rule 0 denotes the absence of violations and is not treated as a separate safety rule.

Table 2. Definitions of the safety rules and violation categories.

Safety Rule ID	Violation Category	Rule Description
Rule 0	No violation	No safety violations identified.
Rule 1	PPE adherence	Hard hats; Clothing covering shoulders and legs; Closed-toe footwear; High-visibility retroreflective vests at night; Face shields or safety glasses for cutting, welding, grinding, or drilling.
Rule 2	Safety harness usage	Safety harnesses required for work at heights of approximately $\geq$ 3 m when effective edge protection is absent.
Rule 3	Edge protection	For underground works or excavation areas deeper than three meters with steep retaining walls, guardrails, fences, barriers, or equivalent warning measures must be provided if workers may stand or work near exposed edges.
Rule 4	Human-machine interaction hazards	When an excavator is operating, or an operator is present in the cab, workers must not be located within the operator's blind spots or within the excavator's actual operating radius.

Issues in the original annotations. To assess the suitability of the original visual localization annotations for our task, we systematically examined the correspondence between violated safety rules and bounding boxes. This examination revealed three recurring types of annotation issues. Figure 1 illustrates the first two issues and the corresponding annotation refinements.

(a) **Multiple instances grouped within a single bounding box.** In some images, multiple workers are enclosed within a single annotated bounding box, making it difficult to determine which individuals the violation label applies to. For samples that require localization of individual workers, these grouped annotations are inconsistent with instance-level localization requirements. We therefore split the shared bounding boxes into individual boxes based on each worker's association with the rule under consideration, as shown in Figure 1a.

(b) **Insufficient spatial precision of some bounding boxes.** Some bounding boxes fit the targets loosely and include substantial irrelevant background, resulting in poor spatial alignment between the annotated regions and the actual targets. This imprecision affects localization evaluation based on intersection over union (IoU). We therefore adjusted the bounding boxes to reduce irrelevant background while retaining the full visible extent of each target, as shown in Figure 1b.

(c) **Incomplete target annotations.** The original annotations omit some targets involved in safety violations that are identifiable from visual evidence. These omissions may cause correct model predictions to be incorrectly counted as false positives during evaluation. We therefore added bounding boxes for the omitted targets based on the conditions under which each rule applies and the available visual evidence.

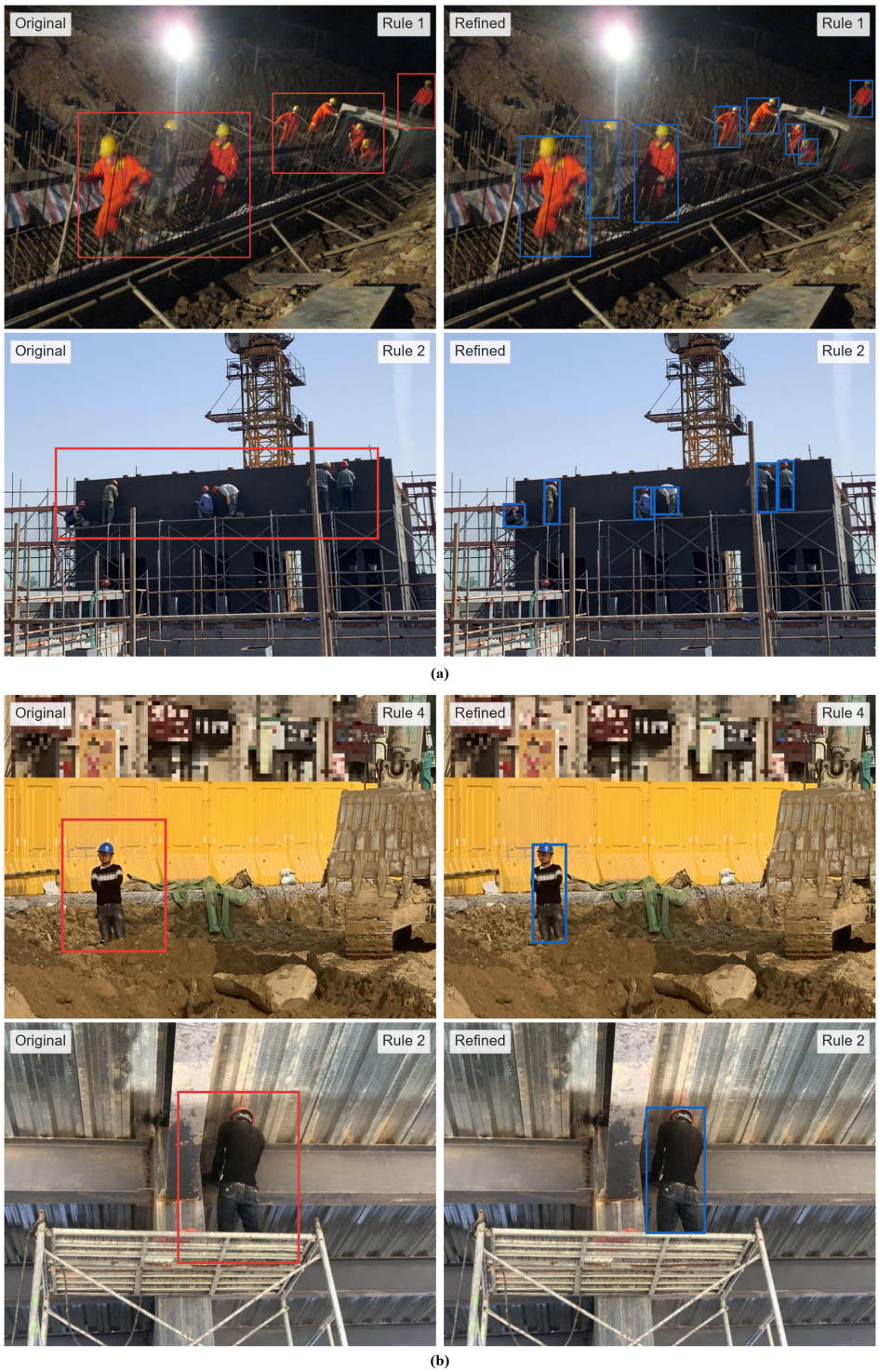


Figure 1. Comparison of original annotations (left, red boxes) and refined annotations (right, blue boxes). (**a**) Refinement of group-level annotations into individual instances. (**b**) Improvement in bounding-box tightness. Selected images are cropped and enlarged for visualization, with identical crops applied to each original–refined pair.

Model-assisted proposals. The overall annotation refinement workflow is illustrated in Figure 2. SAM3 and GPT5.5 assisted in identifying candidate targets to support checks for missed instances, multiple workers sharing one bounding box, and boundary deviations in the original annotations. Model-assisted results served as candidate information requiring human verification based on visible image evidence and the corresponding safety rules.

Manual correction. Two authors (F.T. and Z.G.) performed manual refinement. Annotators and reviewers followed the same annotation guidelines. Before starting, they jointly studied the guidelines and discussed representative cases to standardize rule assessment, instance separation, and bounding-box annotation. The guidelines specified one person instance or hazardous area per violation record. Separate records were created for multiple targets involving the same rule, while separate rule-associated records were retained for a target involving multiple rules. For person instances, bounding boxes covered visible body regions with as little irrelevant background as possible. For hazardous areas, boxes were determined from visible edges and the associated spatial extent. Annotators first manually checked target coverage and rule–instance correspondences using the original images and annotations, along with candidates from the Model-assisted proposals stage. For images with identified issues, they used Label Studio to manually draw and adjust instance bounding boxes and correct rule labels and violation reasons. Questions arising from insufficient evidence or unclear rule applicability were recorded and submitted for further review.

Inter-annotator agreement. To assess agreement between annotators, we selected 100 images from the dataset. Two annotators independently assigned a rule ID and drew a bounding box for each violation instance according to the same annotation guidelines. Following the instance-matching approach of Gupta et al. [38], we compared the two sets of annotations before adjudication. For each image and rule ID, we matched bounding boxes one to one to maximize the sum of their pairwise intersection-over-union (IoU) values. We retained only pairs with an IoU of at least 0.5. We pooled the annotation and matched-pair counts across all 100 images to calculate the instance-level F1 score. The resulting F1 score was 92%.

Quality verification. Two additional reviewers were recruited to ensure annotation quality. They had civil engineering knowledge and were familiar with image annotation. Following the same guidelines, they checked instance coverage in the corrected annotations and consistency among rule labels, bounding boxes, and violation reasons. Identified issues were recorded and returned to the Manual correction stage. The authors responsible for refinement corrected them and resubmitted the annotations for review. For disputed records, reviewers and authors discussed the basis for their judgments using the original images and annotation guidelines. Z.G. provided final confirmation, and the reasons for revisions and adjudication were recorded.

To quantify the changes resulting from annotation refinement, Table 3 reports the number of annotated bounding boxes and the mean bounding box area ratio for each safety rule. The mean bounding box area ratio is defined as the average ratio of each bounding box's area to the area of its corresponding image:

$$\overline{R}_r = \frac{100}{N_r} \sum_{i=1}^{N_r} \frac{w_i h_i}{W_i H_i}, \tag{1}$$

where N_r denotes the number of annotated bounding boxes for rule r, the width and height of the ith box are w_i and h_i, and those of its corresponding image are W_i and H_i, respectively. Before calculation, coordinates from different annotation versions are converted to a common representation normalized by the image width and height.

Table 3 summarizes the annotation statistics for all 10,013 images before and after refinement. The total number of rule-associated bounding boxes increased from 2078 to 2610, while the mean bounding box area ratio decreased from 5.20% to 3.66%. Rule 1 accounted for most of the increase in box count, rising from 1662 to 2108. Box counts also increased for Rules 2 and 4, while their mean bounding box area ratios decreased from 1.72% to 0.84% and from 1.89% to 1.07%, respectively. These changes are consistent with

splitting boxes containing multiple workers, adding missing instances, and adjusting box boundaries, reflecting the refinement of worker annotations into instance-level representations. In contrast, the box count for Rule 3 increased from 199 to 207, while its mean bounding box area ratio decreased from 20.04% to 18.96%. This ratio remained higher than those for the other rules, reflecting differences in annotation targets and spatial extent between continuous hazardous edges and individual workers.

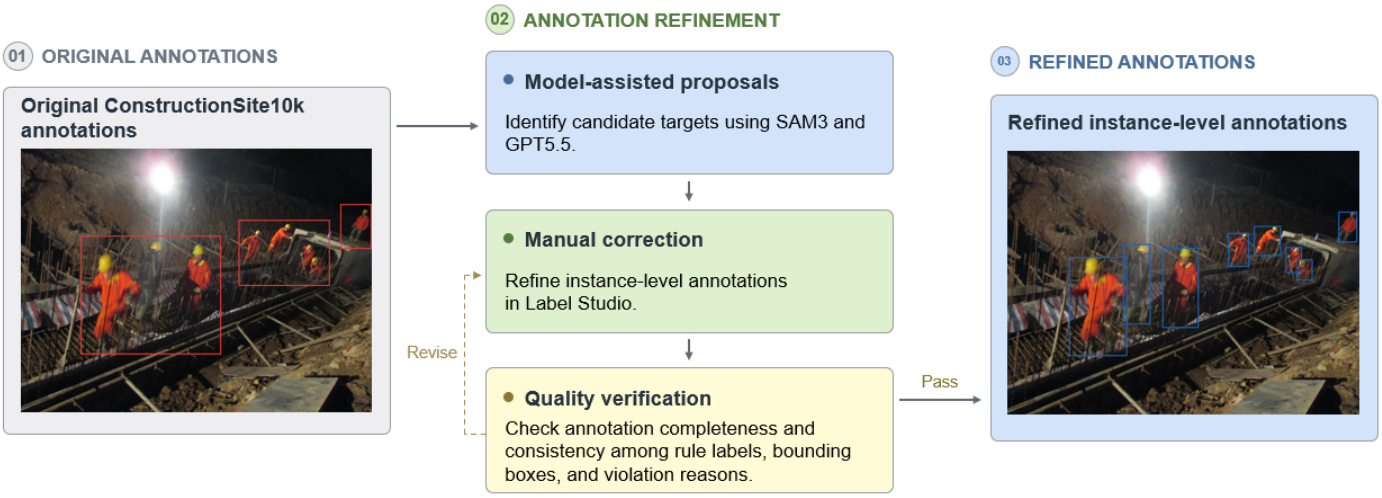


Figure 2. Dataset annotation refinement workflow. Red boxes on the left and blue boxes on the right indicate the original annotations and refined instance-level annotations of ConstructionSite10k, respectively.

Table 3. Annotation statistics before and after refinement across safety rules.

Safety Rule	Box Count		Mean Area Ratio (%)	
	Original	Refined	Original	Refined
Rule 1	1662	2108	3.87	2.54
Rule 2	146	208	1.72	0.84
Rule 3	199	207	20.04	18.96
Rule 4	71	87	1.89	1.07
Total	2078	2610	5.20	3.66

3.2. Overall Framework

We propose RTR-Align (Rule Token-to-Region Alignment), a training method for recognizing and localizing construction safety violations. As shown in Figure 3, the framework comprises annotation refinement and inputs, LoRA adaptation, and region supervision. Full-answer supervision trains the model to generate structured violation records, while an auxiliary region loss constrains the spatial correspondence between each violation record and its associated target.

Given a construction site image I and a task prompt p, the model generates a set of violation instances. The prompt p consists of system and user prompts. The system prompt assigns the model the role of a construction safety inspector and requires the specified output format. The user prompt includes four safety rules and their assessment criteria, instance bounding-box requirements, and JSON output constraints. It requires distinct violation targets to be reported separately, records to be sorted by rule identifier in ascending order, and brief violation explanations. When visual evidence is insufficient to support a violation judgment, the prompt instructs the model not to report that violation.

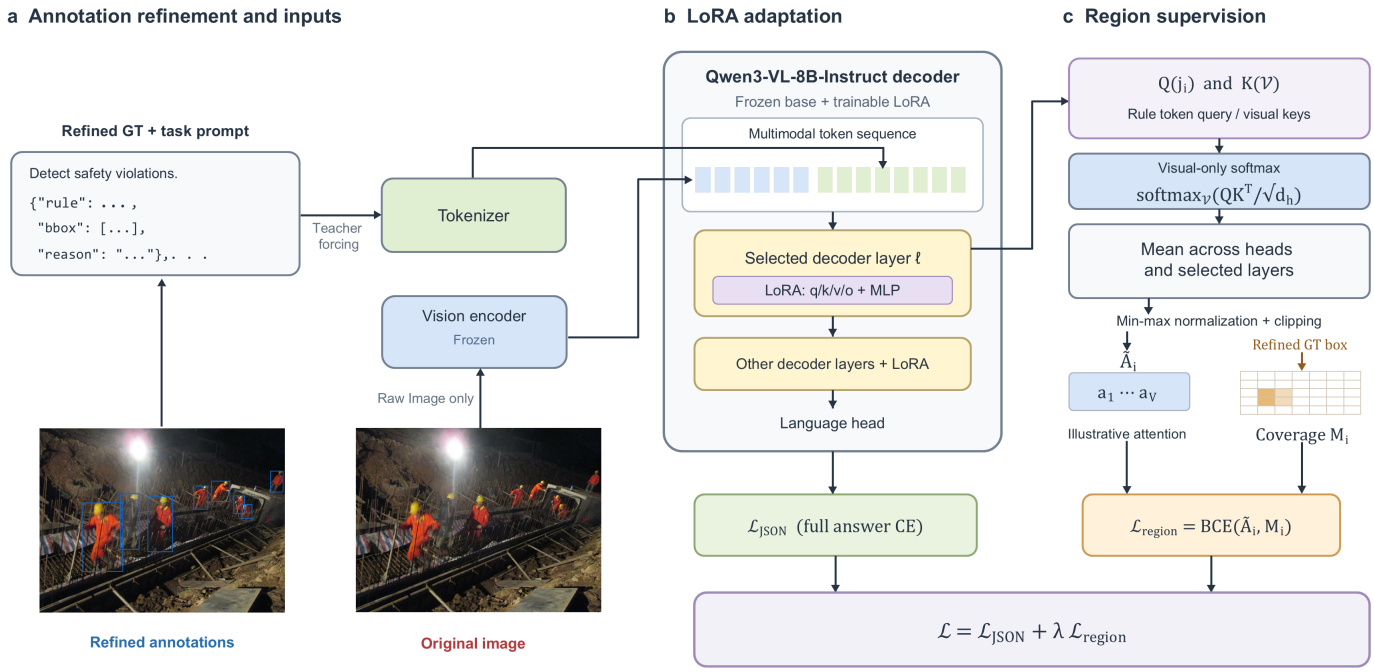


Figure 3. Overview of the RTR-Align framework. (**a**) Annotation refinement and inputs. The vision encoder receives raw RGB images, while refined annotations provide target JSON and bounding boxes for training. (**b**) LoRA adaptation. Base model weights are frozen, and only LoRA parameters in the language decoder are updated using full-answer cross-entropy supervision. (**c**) Region supervision used only during training. The Rule Token in each violation record serves as an anchor to align its visual response with the corresponding bounding-box coverage target. The region loss is averaged over instances and visual positions.

Predictions are represented as $\mathcal{Y} = \{(r_i, b_i, e_i)\}_{i=1}^{N}$, where N is the number of predicted violation records. The variables r_i, b_i, and e_i correspond to the rule, bbox, and reason fields, denoting the rule identifier, bounding box, and brief violation explanation, respectively. Bounding-box coordinates are normalized to 0–1000 relative to image width and height and expressed as an integer sequence $[x_{\min}, y_{\min}, x_{\max}, y_{\max}]$. When the same physical target is associated with multiple rules, separate records are used for each. The output follows a JSON structure. When no violations are reported, the model outputs {"violations":[]}.

During training, the raw RGB image, task prompt, and refined ground-truth (GT) annotations form a multimodal supervision sequence. The refined bounding boxes also provide targets for region supervision. During inference, only the raw RGB image and task prompt are provided. The adapted model autoregressively generates structured violation records based on visible evidence.

3.3. Structured Generation and LoRA Adaptation

RTR-Align uses Qwen3-VL-8B-Instruct as the base model [28]. The vision encoder and base model weights remain frozen, and only LoRA parameters in the language decoder are updated [39]. LoRA uses two smaller trainable matrices to represent a low-rank update to the base weights, enabling task adaptation with fewer trainable parameters. LoRA is applied to the query, key, value, and output projections in attention modules, and the gate, up, and down projections in multilayer perceptron (MLP) modules. The LoRA rank is set to $r = 16$, the scaling parameter to $\alpha = 32$, and dropout to 0.

Training uses teacher forcing: at each step, preceding ground-truth answer tokens are provided, and the model predicts the next token using the image and task prompt. The full-answer cross-entropy loss is

$$\mathcal{L}_{\text{JSON}} = -\frac{1}{T}\sum_{t=1}^{T}\log P_\theta(y_t \mid I, p, y_{<t}). \tag{2}$$

Here, θ denotes the trainable LoRA parameters, y_t the target token at position t, $y_{<t}$ the preceding answer tokens, and T the number of supervised answer tokens. Prompt tokens are excluded from the loss. Intuitively, a higher probability assigned to the correct token yields a lower loss. This loss averages token prediction errors over the complete JSON answer, jointly supervising rule identifiers, bounding-box coordinates, violation explanations, and JSON formatting.

3.4. Rule Token-to-Region Attention Alignment

Supervision over complete JSON answers trains the model to generate violation records but does not directly constrain the visual response associated with each record. During training, RTR-Align introduces region supervision, using the Rule Token in each record as an anchor and the corresponding instance's ground-truth bounding box to align the token's visual response with the target region. The method uses instance-level bounding boxes and requires no object segmentation masks.

For a training image containing N ground-truth violation records, let b_i denote the ground-truth bounding box of the ith record and j_i the position of its Rule Token in the complete multimodal sequence. During training, the sequence positions of Rule Tokens in the answer text are first identified then matched one-to-one with ground-truth bounding boxes in the order of violation records. The query vector is extracted from the Rule Token's own sequence position, at which point the token has already been fed into the model as part of the ground-truth answer. Under causal attention, this position can attend to the image and preceding sequence content, but not to coordinates or explanations appearing later in the same record.

Visual attention from Rule Token queries. To describe each violation record's spatial response in the image, RTR-Align computes association scores between its Rule Token query vector and the key vectors of visual tokens. Let $\mathcal{V}$ denote the set of visual token positions, with cardinality V. For each selected decoder layer and attention head, the computation retains the model's original Q/K normalization, rotary positional encoding, and grouped-query mapping. It applies softmax only over visual token positions to obtain the record's relative response distribution within the image. The response of record i to visual position v at decoder layer ℓ and attention head h is denoted by $a_{iv}^{\ell,h}$. Larger values indicate stronger responses relative to other visual positions.

To combine spatial responses across decoder layers and attention heads, we select the layer set $\mathcal{S} = \{2, 7, 12, 17, 22, 27\}$ using zero-based indexing and compute the unweighted mean:

$$\bar{a}_{iv} = \frac{1}{|\mathcal{S}|H}\sum_{\ell\in\mathcal{S}}\sum_{h=1}^{H} a_{iv}^{\ell,h}. \tag{3}$$

Here, H is the number of attention heads included in aggregation. The selected layers determine where auxiliary attention responses are extracted and do not restrict LoRA updates in other configured decoder layers.

Each instance's aggregated responses are then separately min–max normalized and clipped:

$$\tilde{a}_{iv} = \text{clip}\left(\frac{\bar{a}_{iv} - \min_{u\in\mathcal{V}}\bar{a}_{iu}}{\max_{u\in\mathcal{V}}\bar{a}_{iu} - \min_{u\in\mathcal{V}}\bar{a}_{iu} + \epsilon}, \delta, 1-\delta\right). \tag{4}$$

Here, $\epsilon = 10^{-6}$ is a small positive constant preventing division by zero, and $\delta = 10^{-4}$ defines the clipping bounds to ensure numerical stability in logarithmic operations. The normalized responses are used to compute the region loss and are not interpreted as calibrated violation probabilities.

Bounding-box coverage targets. Region supervision is defined on the spatial grid of the merged visual tokens. Let C_v denote the grid cell corresponding to visual token v. With C_v and the refined ground-truth (GT) bounding box b_i expressed in the same normalized image coordinate system, the coverage target is defined as

$$m_{iv} = \frac{\text{area}(C_v \cap b_i)}{\text{area}(C_v)}. \tag{5}$$

Cells entirely inside or outside the bounding box receive target values of 1 or 0, respectively, while cells intersecting the boundary retain their coverage fractions. This representation preserves partial overlaps at bounding-box boundaries, avoiding rounding small but nonzero overlaps to zero. The target describes rectangular bounding-box coverage, not pixel-level semantic segmentation.

Joint training objective. For instance i, let $\tilde{\mathbf{A}}_i = (\tilde{a}_{i1}, \ldots, \tilde{a}_{iV})$ and $\mathbf{M}_i = (m_{i1}, \ldots, m_{iV})$ denote the normalized visual response and coverage target, respectively, corresponding to the attention and coverage representations in Figure 3. The region loss is binary cross-entropy (BCE) averaged over all annotated instances and visual tokens:

$$\mathcal{L}_{\text{region}} = -\frac{1}{NV} \sum_{i=1}^{N} \sum_{v=1}^{V} [m_{iv} \log \tilde{a}_{iv} + (1 - m_{iv}) \log(1 - \tilde{a}_{iv})]. \tag{6}$$

The complete training objective is

$$\mathcal{L} = \mathcal{L}_{\text{JSON}} + \lambda \mathcal{L}_{\text{region}}, \tag{7}$$

where λ is a hyperparameter controlling the relative contribution of the region loss to the overall objective. For images without annotated violations ($N = 0$), the region loss is skipped, and only the generation loss for the empty violation list is computed. The joint loss optimizes the enabled LoRA parameters, while the vision encoder and base model weights remain frozen.

The trained model is evaluated in terms of both violation identification and spatial localization. For construction safety inspection, precision indicates how many reported violation instances are correct, while recall indicates how many actual violation instances are detected. Higher precision means fewer false alarms among reported violations, and higher recall means fewer missed violations. The F1-score is the harmonic mean of precision and recall, summarizing both aspects [40]. IoU measures the spatial overlap between predicted and ground-truth bounding boxes on a scale from 0 to 1, with higher values indicating closer spatial agreement [41]. The detailed matching protocol and metric definitions are provided in Section 4.4.

3.5. Rule Sampling, Training, and Inference Procedures

To increase sampling opportunities for infrequent rules, we construct five image pools: one without violations and four containing annotations for Rules 1, 2, 3, and 4, respectively. At each training microstep, the no-violation and Rule 1 pools are selected with probabilities of 20% and 40%, respectively. The remaining 40% is distributed equally among the Rule 2, Rule 3, and Rule 4 pools. One image is then sampled with replacement from the selected pool, retaining all its rule-associated annotations. Images involving multiple rules may

belong to multiple pools, so pool selection probabilities need not match the resulting proportions of annotated instances by rule during training.

Each training microstep constructs the multimodal supervision sequence, locates the Rule Token in each violation record, maps the associated bounding boxes onto the visual grid, and computes the full-answer cross-entropy and region losses. Gradients are accumulated before updating the LoRA parameters. Images without violations receive only generation supervision, with an empty violation list.

During inference, the adapted model and the same task prompt are retained, while region target construction and auxiliary loss computation are disabled. The model encodes the raw RGB image and autoregressively generates a structured answer. Rule identifiers and bounding boxes are then parsed according to a unified evaluation protocol. Thus, spatial supervision is confined to training, and inference requires no GT annotations.

4. Experimental Design

4.1. Overall Experimental Design

This section evaluates RTR-Align's violation recognition and localization capabilities from four perspectives: overall performance, differences across rules and scenarios, ablation experiments, and qualitative cases. We compare models on the same test set and analyze performance across four safety rules and groups defined by distance, lighting, viewpoint, and information richness. We then examine the effects of annotation refinement, the number and positions of supervised layers, image-pool sampling probabilities, and region loss weights, and use success and failure cases to analyze target association, localization errors, false positives, and missed detections.

4.2. Dataset and Experimental Splits

The experiments in this study use the ConstructionSite10k dataset [6], which contains 10,013 construction site images. The official split comprises 7009 training images and 3004 test images. Each violation record includes a rule identifier, a natural language explanation of the violation, and a bounding box corresponding to the object involved. We retain the same set of images as the official test split. Table 4 presents the distribution of image attributes in the official training and test splits.

Table 4. Distribution of dataset parameters in the test and training sets.

Split	Measure	Camera Distance			Illumination				View		Quality	
		Long	Mid	Short	Night	Normal	Over-exposed	Under-exposed	Elevation	Plan	Rich Info	Poor Info
Test	Count	334	1309	1361	43	2426	154	381	2895	109	1291	1713
	Ratio	11.12%	43.58%	45.31%	1.43%	80.76%	5.13%	12.68%	96.37%	3.63%	42.98%	57.02%
Train	Count	386	5063	1560	159	5885	273	692	6286	723	3980	3029
	Ratio	5.51%	72.24%	22.26%	2.27%	83.96%	3.89%	9.87%	89.68%	10.32%	56.78%	43.22%

Count denotes the number of images in each category. Ratio denotes the percentage of images in that category within the corresponding dataset split (3004 test images and 7009 training images). Percentages may not sum to 100% due to rounding.

4.3. Model and Experimental Setup

The models evaluated include LLaVA-OV-2-8B [24], InternVL3.5-8B [25], GLM-4.6V-Flash [26], Qwen3-VL-8B [28], GPT-5.6 Sol high [27], and the proposed RTR-Align. RTR-Align uses LoRA to adapt Qwen3-VL-8B-Instruct, keeping the base model weights frozen and updating only the adapter parameters.

Training used the AdamW optimizer with a peak learning rate of 5×10^{-5}, with the model loaded in BF16 precision and cosine learning-rate decay applied after warm-up.

The region alignment loss weight was set to $\lambda = 0.05$ in the main run. During inference, the Qwen3-VL-8B baseline and RTR-Align used identical rule prompts and generation parameters in Table 5, with a maximum generation length of 1024 tokens. One prediction was generated per test image and scored using a unified output parsing and evaluation protocol. Training ran for 8192 iterations with an effective batch size of 8. Linear learning-rate warm-up was used with a warm-up ratio of 5%. Gradients were accumulated over 8 steps. Training used a single NVIDIA RTX PRO 6000 GPU with 96 GB of memory. GPT-5.6 Sol was evaluated with the high reasoning effort setting.

Table 5. Inference settings shared by Qwen3-VL-8B and our method.

Parameter Type	Value	Description
do_sample	True	Stochastic sampling
max_new_tokens	1024	Maximum output tokens
temperature	0.2	Sampling randomness
top_p	1.0	Nucleus sampling threshold
top_k	20	Candidate token count
batch_size	8	Images per batch

4.4. Evaluation Metrics

We use precision, recall, and F1 score to evaluate the identification of safety-violation instances. Spatial localization is assessed using the mean intersection over union of matched bounding boxes (Matched IoU) and instance-level overall intersection over union (Total IoU). Each instance is represented by both a rule category and a bounding box. Performance evaluation in this study focused on violation instance recognition and localization. The explanatory content of the reason field in the model output was not evaluated.

4.4.1. Violation Instance Identification Metrics

For each image, predicted and ground-truth instances are eligible for one-to-one matching only if they share the same rule category and their bounding box IoU is ≥ 0.5. Matching first maximizes the number of valid pairs and, among solutions with the same number of pairs, maximizes the sum of IoU values. Matched pairs are counted as true positives (TP), unmatched predictions as false positives (FP), and unmatched ground-truth instances as false negatives (FN). Counts are aggregated across the entire test set to calculate micro-averaged metrics:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{8}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{9}$$

$$\text{F1} = \frac{2TP}{2TP + FP + FN} \tag{10}$$

4.4.2. Spatial Localization Metrics

To avoid overestimating localization quality by evaluating only instances that meet the IoU threshold, localization evaluation uses a separate matching procedure without an IoU threshold. For each image, instances are first grouped by rule category, then matched one-to-one within each category. The number of pairs equals the smaller of the predicted and ground-truth instance counts, and the assignment maximizing the sum of IoU values is selected.

Matched IoU measures localization quality after pairing instances within the same rule category:

$$\text{Matched IoU} = \frac{1}{M}\sum_{k=1}^{M}\text{IoU}_k \tag{11}$$

Total IoU incorporates unmatched predictions and unmatched ground-truth instances as zero-overlap terms, capturing the combined effects of localization errors, false positives, and false negatives:

$$\text{Total IoU} = \frac{\sum_{k=1}^{M}\text{IoU}_k}{M + U_p + U_g} \tag{12}$$

where M is the total number of matched pairs across the test set. For the kth pair of predicted and ground-truth bounding boxes, IoU_k is the ratio of their intersection area to their union area. U_p and U_g denote the numbers of unmatched predictions and unmatched ground-truth instances, respectively, in this matching procedure. Both localization metrics are computed by aggregating across the entire test set, without averaging per-image scores. Image–rule combinations with neither ground-truth instances nor predictions contribute to neither the numerator nor the denominator of the localization metrics. All metrics are treated as undefined when their denominators are zero.

5. Experimental Results and Analysis

5.1. Overall Performance

Table 6 compares the overall performance of different models in construction safety violation recognition and localization. In the main run, RTR-Align achieves the highest precision, F1, Matched IoU, and Total IoU among the models in Table 6, while its recall is lower than that of the Qwen3-VL-8B baseline. These results indicate that task-specific adaptation mainly improves the accuracy of violation predictions and overall localization performance, although a trade-off between precision and recall remains.

For recognition, the Qwen3-VL-8B baseline achieves a recall of 69.29% but a precision of only 8.16%. This indicates that it detects many actual violation instances but also produces numerous false positives. Compared with the Qwen3-VL-8B baseline, RTR-Align in the main run increases precision from 8.16% to 58.13% and F1 from 14.61% to 62.72%, while recall decreases by 1.19 percentage points, from 69.29% to 68.10%.

Thus, the recognition gains mainly arise from reducing false positives while retaining most of the baseline's ability to detect actual targets. GPT-5.6 Sol high achieves higher precision and F1 than the other models without task-specific adaptation in the table, but lower values than RTR-Align's main run. These results indicate that RTR-Align achieves a better precision–recall balance in this evaluation.

The Qwen3-VL-8B baseline achieves a Matched IoU of 70.62% but a Total IoU of only 6.85%, indicating that bounding-box overlap for matched instances alone does not fully reflect the effects of unmatched predictions and missed detections on overall localization performance. RTR-Align improves these metrics to 77.64% and 39.40%, respectively, indicating gains in both bounding-box fit for matched instances and overall localization quality when unmatched instances are included. Across five random seeds, F1, Matched IoU, and Total IoU are 60.61% ± 1.28%, 77.38% ± 0.84%, and 37.89% ± 0.96%, respectively. The means of all three metrics exceed the corresponding results of the other comparison models in the table. The overall advantage observed in the main run is maintained across random seeds. Under identical inference conditions, RTR-Align processed the entire test set in 48.00 min compared with 154.75 min for Qwen3-VL-8B, a difference that may be partly explained by fewer false positives leading to fewer predicted violation records and thus less time required for autoregressive decoding.

Table 6. Overall performance comparison of different models on construction safety violation recognition and localization.

Model	Precision	Recall	F1	Matched IoU	Total IoU	Time
LLaVA-OV-2-8B [24]	3.77	33.69	6.78	43.46	3.57	270.60
InternVL3.5-8B [25]	6.16	22.50	9.67	35.44	5.60	83.65
GLM-4.6V-Flash [26]	7.75	60.60	13.74	69.49	6.33	99.53
Qwen3-VL-8B [28]	8.16	69.29	14.61	70.62	6.85	154.75
GPT-5.6 Sol high [27]	**32.89**	65.12	**43.71**	**73.68**	**26.38**	—
Ours (main run)	58.13	**68.10**	62.72	77.64	39.40	48.00
Ours (5 seeds)	58.11 ± 2.57	63.52 ± 3.18	60.61 ± 1.28	77.38 ± 0.84	37.89 ± 0.96	47.59 ± 0.91

Time denotes inference time for the entire test set, in minutes. Training for the main RTR-Align run took approximately 85 min. The five-seed row reports the mean ± sample standard deviation across random seeds 1, 2, 3, 4, and 337187. The main run used random seed 337187. For recognition and localization metrics, underlining and boldface denote the best and second-best values, respectively. Inference with GPT-5.6 Sol (high) resumed after an interruption and was completed for the entire test set. The dash (—) indicates that total inference time across sessions could not be reliably determined.

5.2. Performance Across Safety Rules

Table 7 compares the recognition and localization performance of the Qwen3-VL-8B baseline and RTR-Align across four safety rules. RTR-Align achieves higher precision, F1, and Total IoU for every rule, indicating that task-specific adaptation improves performance across rules. However, performance levels and changes in individual metrics vary by rule. Rule 1 shows the best overall performance, whereas Rule 3 remains relatively weak in recognition and localization. Rules 2 and 4 exhibit more pronounced precision–recall trade-offs. Thus, per-rule evaluation reveals performance differences not fully captured by overall metrics.

For recognition, Rule 1 achieves higher precision while largely maintaining recall, raising F1 from 17.35% to 73.89%. This indicates that fewer false positives are the main source of improvement for this rule. For Rule 3, both precision and recall increase, indicating that the model detects more actual violation instances while reducing false positives.

Table 7. Performance comparison across safety rules (all metrics expressed as %).

Model	Rule ID	Precision	Recall	F1	Matched IoU	Total IoU
Qwen3-VL-8B [28]	Rule 1	9.79	76.06	17.35	74.37	8.19
	Rule 2	8.31	69.33	14.84	72.34	6.53
	Rule 3	0.57	5.71	1.03	19.70	1.13
	Rule 4	3.58	68.57	6.80	66.89	3.07
Ours	Rule 1	73.39	74.39	73.89	79.82	50.82
	Rule 2	30.66	56.00	39.62	70.98	19.67
	Rule 3	25.00	40.00	30.77	54.57	15.37
	Rule 4	16.67	31.43	21.78	86.88	10.62

For localization, RTR-Align achieves higher Matched IoU for Rules 1, 3, and 4. The improvement for Rule 3 indicates better spatial localization of matched instances. Rule 2 shows a different pattern: Matched IoU decreases slightly from 72.34% to 70.98%, while Total IoU increases from 6.53% to 19.67%. For Rule 2, Total IoU increased because matched pairs accounted for a larger share of its denominator, even though their mean IoU decreased. Rule 4 achieves a Matched IoU of 86.88% but a Total IoU of only 10.62%. Together with

its low precision and recall, these results indicate that accurate localization of matched instances cannot fully offset the effects of false positives and missed detections. Therefore, the two localization metrics should be interpreted together to distinguish the localization quality of matched boxes from overall performance across all predictions.

5.3. Performance Comparison Across Scenario Groups

Table 8 compares the Qwen3-VL-8B baseline and RTR-Align across groups defined by distance, lighting, viewpoint, and information richness. RTR-Align achieves higher precision, F1, and Total IoU in all groups, indicating that task-specific adaptation improves performance under different scene conditions.

Across distance groups, RTR-Align's F1 and Total IoU increase from 53.85% and 29.10% for distant targets to 66.58% and 44.61% for nearby targets, respectively. However, recall in the long-distance group decreases from the baseline's 76.00% to 56.00%, indicating that precision gains are accompanied by more missed detections. Among lighting groups, the night group achieves the highest F1 and Total IoU, at 87.85% and 69.51%, respectively, but its Matched IoU is slightly lower than the baseline's. This indicates that localization quality for the full prediction set and bounding-box overlap for matched instances should be examined separately. In the Plan view group, Matched IoU reaches 90.68%, while Total IoU is only 17.55%, with precision and recall at 33.33% and 31.58%, respectively. These results indicate that false positives and missed detections still limit overall performance in this group. Furthermore, both F1 and Total IoU are lower for the Rich info group than for the Poor info group, indicating that the group with richer information in this test set does not achieve better recognition and localization results.

Table 8. Performance comparison across distance, lighting, view, and information quality groups (all metrics expressed as %).

Model	Scenario	Precision	Recall	F1	Matched IoU	Total IoU
Qwen3-VL-8B [28]	Long distance	5.82	76.00	10.82	70.35	4.70
	Mid distance	6.90	63.09	12.44	65.39	5.56
	Short distance	10.96	74.73	19.11	75.89	9.61
	Night	74.77	72.73	73.73	83.10	51.47
	Normal lighting	7.23	70.19	13.10	68.91	6.10
	Overexposed	7.17	73.47	13.07	65.12	5.79
	Underexposed	6.66	58.65	11.96	71.58	5.60
	Elevation view	8.27	69.06	14.77	70.42	6.93
	Plan view	5.49	78.95	10.27	78.78	4.87
	Poor info	12.47	77.48	21.48	74.06	10.44
	Rich info	5.82	61.70	10.64	67.03	4.91
Ours	Long distance	51.85	56.00	53.85	74.08	29.10
	Mid distance	56.71	65.34	60.72	72.79	36.65
	Short distance	60.77	73.63	66.58	83.18	44.61
	Night	90.38	85.45	87.85	80.74	69.51
	Normal lighting	53.73	66.20	59.32	77.09	36.26
	Overexposed	64.81	71.43	67.96	76.28	40.98
	Underexposed	53.04	58.65	55.71	77.16	34.01
	Elevation view	58.59	68.94	63.35	77.51	39.97
	Plan view	33.33	31.58	32.43	90.68	17.55
	Poor info	61.67	73.27	66.97	78.92	43.69
	Rich info	54.76	63.30	58.72	76.28	35.58

Table 4 lists the number of test images in each scene group.

5.4. Effects of the Number and Positions of Supervised Layers

To examine how the number and positions of decoder layers used for region supervision affect performance, we compare configurations with 1, 3, 6, 9, and 12 layers, as shown in Table 9. For the single-layer, three-layer, and six-layer settings, we also evaluate different layer positions. All other training and evaluation settings remain unchanged.

F1 is the primary metric for comparing supervision configurations because it combines precision and recall and is suitable for evaluating violation instance detection when both false positives and missed detections matter. Under our evaluation protocol, a correct detection also requires a matching rule and bounding-box IoU ≥ 0.5. Thus, F1 reflects violation instance detection performance that meets this localization requirement. Reporting precision and recall also helps explain performance differences between configurations.

The main configuration selects layers $2, 7, 12, 17, 22, 27$, with an index interval of 5 between adjacent supervised positions. This distributes region alignment supervision across multiple network depths rather than concentrating it at a single position. Other configurations assess whether shifting supervised positions or increasing the number of supervised layers provides additional gains.

Moving the single supervised layer from index [2] to [19] or [33] changes F1 from 62.12% to 61.61% and 61.75%, respectively, indicating that later supervision positions do not necessarily improve violation recognition. The main six-layer configuration achieves an F1 of 62.72% and a recall of 68.10%. Shifting all six supervised positions reduces recall to 64.52% or 64.40%. In contrast, shifting the three-layer configuration to later positions improves F1, indicating that the effect of supervision position depends on the selected layer combination. The nine-layer configuration achieves an F1 of 62.69%, close to the main configuration, whereas the twelve-layer configuration drops to 60.99%. Thus, increasing the number of supervised layers does not yield monotonic gains across the tested combinations of layer counts and positions. The main six-layer configuration achieves an F1 close to that of the nine-layer configuration with fewer supervised positions while retaining higher recall.

Table 9. Ablation study on the number and positions of supervised layers. All other training and evaluation settings remain unchanged. Layer indices are zero-based. Precision, Recall, and F1 are expressed as percentages. ΔF1 denotes the F1 difference from the main configuration [2, 7, 12, 17, 22, 27], in percentage points. The highest values for precision, recall, and F1 are underlined.

Layer Number	Layer Indices	Precision	Recall	F1	ΔF1
1	[2]	59.48	65.00	62.12	−0.60
1	[19]	58.75	64.76	61.61	−1.11
1	[33]	57.78	66.31	61.75	−0.97
3	[2, 15, 27]	55.57	62.98	59.04	−3.68
3	[5, 18, 30]	58.35	64.88	61.44	−1.28
3	[8, 21, 33]	60.05	62.98	61.48	−1.24
6	[2, 7, 12, 17, 22, 27]	58.13	<u>68.10</u>	<u>62.72</u>	—
6	[5, 10, 15, 20, 25, 30]	59.50	64.52	61.91	−0.81
6	[8, 13, 18, 23, 28, 33]	60.11	64.40	62.18	−0.54
9	[2, 5, 8, 11, 14, 17, 20, 23, 26]	<u>60.53</u>	65.00	62.69	−0.03
12	[2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24]	56.91	65.71	60.99	−1.73

5.5. Ablation Experiments

5.5.1. Effect of Annotation Refinement

To evaluate the contribution of annotation refinement, we compare two training configurations: A1 uses the original training annotations, and A2 uses refined training annotations. The trained models are evaluated on the same test images and annotations, with results reported below:

Table 10 compares training annotation versions using identical evaluation annotations. Refined annotations improve precision, recall, and F1 by 6.13, 6.08, and 6.15 percentage points, respectively, confirming recognition gains from instance-level training annotation refinement. Boxes enclosing multiple workers merge evidence from different people, while loose boundaries include background in the target region. Separating instances and correcting boundaries improve consistency between rule records and spatial targets, while adding annotations for missed targets provides more effective instance supervision. The simultaneous gains in precision and recall support the benefits of this overall annotation refinement and link the data issues in Figure 1 to model performance.

Table 10. Effect of training annotation refinement. All performance metrics are expressed as percentages.

Group	Training Annotations	λ	Precision	Recall	F1
A1	Original GT	0.05	52.00	62.02	56.57
A2	Refined GT	0.05	58.13	68.10	62.72

5.5.2. Effect of the Rule–Target Sampling Strategy

Table 11 compares violation recognition performance under different image-pool sampling probabilities. With the no-violation sampling proportion fixed at 20%, changing the Rule 1 and Rules 2–4 pool probabilities from C3's 20%/60% to C0's 40%/40% raises F1 from 57.80% to 62.72%. Further changing them to C4's 60%/20% increases precision but reduces recall to 62.74%. C1 has the highest recall and lowest precision, while C2 shows the opposite, indicating that sampling allocations change the model's tendency to report violations. C0 achieves the highest F1 among the tested configurations and retains higher recall than C4, making it suitable as the main configuration balancing effective detection and false-positive control.

Table 11. Overall test-set performance under different rule–target sampling ratios.

Group	No-Violation Samples	Rule 1 Samples	Rules 2–4 Samples (Combined)	Precision (%)	Recall (%)	F1 (%)
C0	20%	40%	40%	58.13	68.10	62.72
C1	10%	45%	45%	46.44	69.88	55.80
C2	40%	30%	30%	64.79	54.76	59.35
C3	20%	20%	60%	51.29	66.19	57.80
C4	20%	60%	20%	62.37	62.74	62.55

5.5.3. Effect of Region Alignment Loss and Sensitivity to Its Weight

In Table 12, the $\lambda = 0$ control retains full JSON answer supervision and LoRA adaptation. Bounding-box coordinates remain part of the training answers. Only the region alignment loss is disabled to assess its additional contribution. Compared with this control, the main configuration increases precision by 7.74 percentage points and recall by 13.58 percentage points, with F1 rising from 52.37% to 62.72%. These results show that adding explicit alignment constraints between the Rule Token and visual regions to full JSON answer supervision further improves violation instance detection in this experiment, directly supporting our rationale that implicit correspondence alone is insufficient. All tested positive weights yield higher F1 than the control without region loss, but F1 does not increase monotonically with the weight. Among the weights tested, $\lambda = 0.05$ yielded the highest observed F1 score (62.72%) and recall (68.10%). Because this comparison was based on a single run per weight, it remains unclear whether the F1 ranking would be consistent across random seeds.

Table 12. Comparison of violation identification performance across region loss weights.

Region Weight	Precision (%)	Recall (%)	F1 (%)
0	50.39	54.52	52.37
0.03	55.52	65.24	59.99
0.04	57.94	62.98	60.35
0.05	58.13	68.10	62.72
0.06	56.71	66.43	61.18
0.07	59.63	65.24	62.31
0.10	59.19	64.76	61.85
0.15	59.58	64.76	62.07
0.20	59.52	62.50	60.98

5.6. Qualitative Comparison and Failure Analysis

Figure 4 compares ground-truth annotations with predictions from the Qwen3-VL-8B baseline and RTR-Align, showing differences in rule assessment, target selection, and instance coverage. In (b), the base model focuses its predictions on people, whereas RTR-Align outputs the hazardous edge region associated with Rule 3, illustrating that different safety rules require different localization targets. In (e), the base model points to people below, whereas RTR-Align localizes the two workers at height in the upper part of the image associated with Rule 2. Its target selection matches the ground truth. These two sets of cases show that construction safety violation inspection requires selecting the relevant people or hazardous areas according to the rules. Identifying people in an image alone is insufficient for violation assessment. In the partially occluded and nighttime scenes in (c,d), RTR-Align detects person instances missed by the base model. In (a,f), RTR-Align reduces predicted boxes without corresponding ground-truth annotations, and the rule categories of retained targets agree with the ground truth. These qualitative results show RTR-Align's performance in reducing false positives, recovering missed instances, and establishing rule–target correspondences, consistent with the overall recognition gains.

Figure 5 presents RTR-Align's failure cases involving partially visible people, occlusion, small targets, and multiple coexisting rules. In (a), the person at the left image edge is only partly visible. The model detects the person on the right associated with Rule 1 but misses this edge instance. In (b), the person behind is occluded by a foreground person and is not detected as a separate instance associated with Rule 1. These cases show that identifying every violation instance remains challenging when people are incompletely visible or multiple instances occlude one another. In (c), the two workers at height appear small in the image, and the scene involves both Rules 2 and 3. In (d), the hazardous edge region corresponding to Rule 3 and the people associated with Rule 4 require simultaneous inspection.

In both cases, the model outputs a Rule 3 region box but misses the people associated with Rules 2 and 4, respectively. This indicates that predicting a box for one rule does not guarantee complete detection of instances associated with other rules in the same scene. Joint inspection of multiple rules requires the model to consider the judgment criteria and corresponding targets of different rules simultaneously. These missed detections indicate that the task remains difficult in complex scenes. Moreover, the predicted region boxes in (c,d) differ in extent from the ground truth, indicating that hazardous-area localization accuracy also needs improvement alongside more complete detection of instances associated with multiple rules.

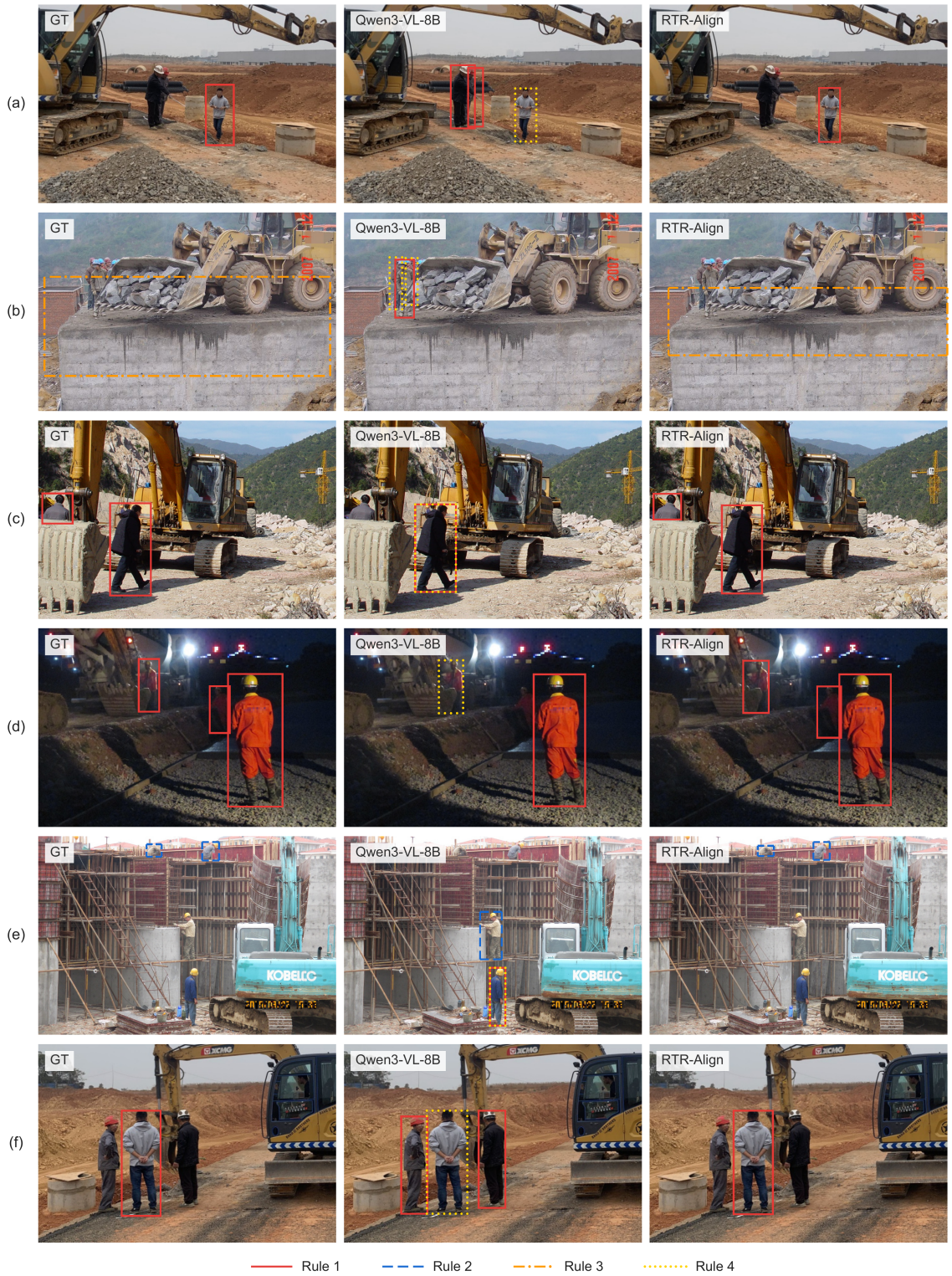


Figure 4. Qualitative comparison across construction scenes. Columns show ground truth (GT), Qwen3-VL-8B predictions, and RTR-Align predictions, respectively. Red solid, blue long-dashed, orange dash-dotted, and yellow dotted boxes denote Rules 1, 2, 3, and 4, respectively. The same crop is used across columns in each row.The rows illustrate (**a**) Rule 1 target selection near an excavator; (**b**) localization of a hazardous edge region associated with Rule 3; (**c**) detection of a partially occluded Rule 1 instance; (**d**) detection of Rule 1 instances in a nighttime scene; (**e**) localization of two workers at height associated with Rule 2; and (**f**) Rule 1 target selection in a scene with multiple people.



Figure 5. Failure cases in challenging construction scenes. The left column shows ground truth (GT), and the right column shows RTR-Align predictions, covering cases of partially visible people, occlusion, small targets, and coexisting rules. Bounding-box colors and line styles for each rule are shown in the legend. Both columns in each row use the same crop. The panels show (**a**) a missed partially visible Rule 1 instance at the left image edge; (**b**) a missed Rule 1 instance occluded by a foreground person; (**c**) missed detections of both Rule 2 and Rule 3 violations in a scene where the two rules coexist; and (**d**) missed Rule 4 instances in a scene with coexisting Rules 3 and 4.

Liu et al. [42] evaluated word–region alignment by providing ground-truth captions and measuring the fraction of visual attention that fell within the corresponding annotated region as the model predicted each word of interest. Che et al. [43] measured the fraction

of a generated object token's visual attention that fell inside the ground-truth bounding box of the corresponding object. Hu et al. [44] supplied Qwen3-VL with the ground-truth answer and measured the share of visual attention that the answer tokens assigned to the annotated evidence box. We adapted this analysis to construction-site violations by giving the base model (Qwen3-VL-8B-Instruct) and RTR-Align the same image and task prompt. For each image, we used teacher forcing with the same ground-truth JSON response to compare the two models at the rule-ID token for each annotated violation. At that position, we computed attention to the visual tokens and normalized the weights over those tokens within each head. We then averaged the weights across all heads and the six selected decoder layers $[2, 7, 12, 17, 22, 27]$ to obtain an attention map for each violation.

Figure 6 compares the visual attention maps before and after adaptation for three test scenes. In all three scenes, a larger share of visual attention fell within the union of the annotated violation boxes after adaptation. This share rose from 11.29% to 18.48% in the top row, from 8.46% to 15.77% in the middle row, and from 16.32% to 28.11% in the bottom row. Across the test images with annotated violations, the within-box attention ratio decreased after adaptation in only seven images and increased in all others. With each image weighted equally, the mean ratio increased from 7.16% for the base model to 11.69% for RTR-Align, a gain of 4.53 percentage points (Table A2, Appendix C).

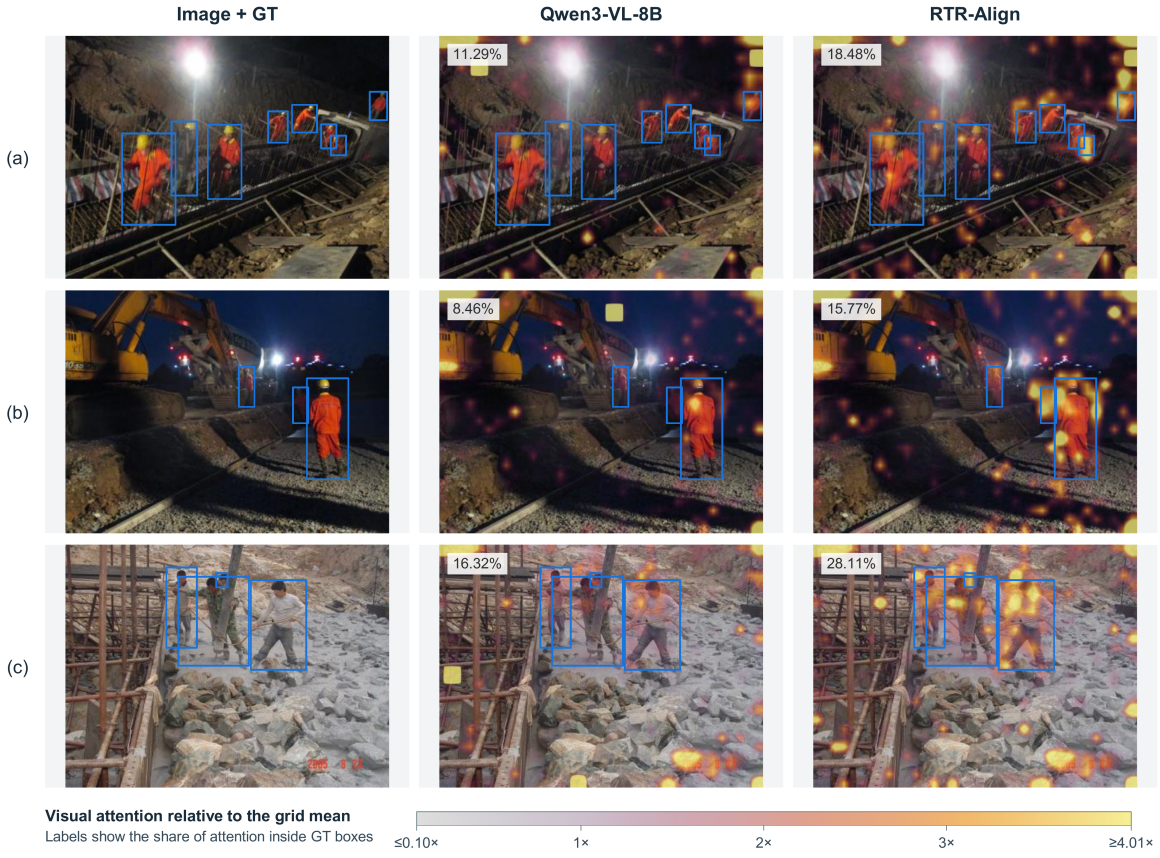


Figure 6. Visual attention maps of Qwen3-VL-8B and RTR-Align for three construction-site scenes. Percentages show the share of attention inside the blue ground-truth boxes.Panels (**a**,**b**) show nighttime scenes with workers not wearing high-visibility vests, and panel (**c**) shows a daytime scene with workers not meeting personal protective equipment requirements.

6. Discussion

The overall comparison and region loss ablation jointly indicate that improvements in rule checking depend not only on general visual reasoning but also on how judgment–

target correspondences are supervised. Detection-guided methods use externally supplied worker and machinery locations to organize the reasoning context [7]. VEST learns hazard reasoning with coordinates through image-level labels and semantic rewards [8]. RTR-Align's instance bounding-box supervision further specifies the region referenced by each violation record. These approaches organize spatial evidence through object context, weakly supervised localization, and record-level correspondence, respectively. Our ablation supports the additional value of the last constraint beyond full-answer supervision.

Rule 1 assessment focuses on people and their protective equipment. The supervised targets have direct spatial links to the assessed attributes, and annotations for this rule are relatively abundant in the dataset. Rule 2 additionally requires understanding height and edge protection, whereas Rule 3 concerns extended hazardous edges. These differences indicate that the benefits of instance-level supervision must be considered alongside the contextual understanding required by each rule. Analysis across scenario groups further shows that high localization quality for matched instances at long distances and in top-down views cannot replace complete detection, consistent with the small-target and occlusion issues in the failure cases. Performance in the nighttime and information richness groups may also depend on their target and rule composition. These results are therefore better suited to identifying visual evidence that needs strengthening than to attributing differences to a single scene factor.

Rule 4 checks whether workers enter an excavator's operating radius or the operator's blind spots when the excavator is operating or the operator is present. RTR-Align's precision increases from 3.58% to 16.67%, while recall decreases from 68.57% to 31.43%, showing fewer false positives alongside reduced detection coverage. This provides more targeted review cues but also means that more annotated human–machine interaction risk instances are not correctly recognized and localized, limiting its ability to independently warn of human–machine proximity risks. The absence of a Rule 4 prediction should therefore not be interpreted as evidence that no human–machine interaction risk is present. Such rules rely on joint evidence of people, machinery, and operating states. Future work should strengthen relational and temporal modeling based on instance localization and assess its value in practical inspection workflows through cross-site validation.

Training and evaluation in this study were conducted on the ConstructionSite10k dataset [6], without cross-dataset validation. In addition, we did not conduct a quantitative comparison between RTR-Align and methods that pair a separate object detector with a vision-language model using a common test set and evaluation protocol. Their relative performance in recognizing and localizing construction safety violations therefore remains to be assessed.

7. Conclusions

This study proposes RTR-Align, a Rule Token-to-Region Alignment method for construction safety violation recognition and localization. Using Qwen3-VL-8B-Instruct and instance-level annotation refinement of ConstructionSite10k, it incorporates the correspondence between each violation record and its visual region into model training. Given construction site images and natural language safety rules, the model can generate structured records containing violated rules, target locations, and related explanations, grounding safety judgments in specific targets. In the main run, RTR-Align achieved higher precision and better overall localization than the unadapted base model, but its recall was 1.19 percentage points lower. Region loss ablation further shows that explicit correspondence supervision provides additional gains beyond full-answer and coordinate supervision, supporting joint learning of rule assessment and instance association.

This study provides a research foundation for AI-assisted construction safety inspection in methodology, data, and application. Methodologically, modeling rule-associated instances connects safety judgments with spatial evidence, providing a concrete approach to adapting general-purpose vision-language models for specialized rule checking. At the data level, refining boxes shared by multiple workers into individual instance annotations and improving bounding-box fit align training supervision more closely with the objects requiring verification. This also provides more detailed data for future comparisons of supervision strategies, analysis of instance association errors, and evaluation of localization capabilities. At the application level, outputs combining rules, locations, and explanations help inspectors identify the people or hazardous areas referenced by model judgments, supporting item-by-item review and organization of violation records. The study also shows that rules involving operational relationships, such as human–machine interactions, still face insufficient target coverage. Thus, the current method is intended to provide professionals with verifiable visual cues, while on-site personnel assess risks and appropriate response measures based on actual operating conditions.

Future research should expand data coverage across construction sites, work stages, and environmental conditions and conduct independent cross-site validation while maintaining consistent instance-level annotations. This would test generalization across target scales, occlusion levels, and scene layouts. For distant and partially visible targets, combining local high-resolution observations with multiscale information could be explored to improve the discernibility of small person targets and protective equipment. For human–machine interaction rules, sequential images, relative positions of people and machinery, and equipment operating states should be incorporated to improve understanding of how hazardous relationships change over time and test whether violation instance coverage can increase while controlling false positives. Research could also cover more safety rules involving relationships among people, equipment, and the environment and incorporate inspectors' reviews of model outputs into the improvement process. Practical workflow trials with site management staff and safety inspectors, evaluating output comprehensibility, review efficiency, and the transition to corrective action, would help clarify the conditions for and value of integrating this method into construction safety inspection workflows.

Author Contributions: Conceptualization, F.T. and S.X.; methodology, F.T., Z.G. and S.X.; software, F.T. and Z.G.; validation, F.T., Z.G. and S.X.; formal analysis, F.T. and Z.G.; investigation, Z.G. and S.X.; resources, F.T. and Z.G.; data curation, F.T. and Z.G.; writing—original draft preparation, F.T., Z.G. and S.X.; writing—review and editing, F.T. and Z.G.; visualization, F.T. and S.X.; supervision, F.T. and Z.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 62302151.

Data Availability Statement: The annotation guidelines and refined annotations for this study are available at https://github.com/zbgao-01/RTR-Align (accessed on 4 November 2026).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this paper:

RTR-Align	Rule Token-to-Region Alignment
VLM	Vision-Language Model
LoRA	Low-Rank Adaptation
SAM	Segment Anything Model
GT	Ground Truth

IoU	Intersection over Union
PPE	Personal Protective Equipment

Appendix A. Glossary of Key Technical Terms

LoRA (Low-Rank Adaptation)

Low-rank adaptation (LoRA) enables parameter-efficient model adaptation by representing weight updates in selected linear layers using trainable low-rank matrices while keeping the original weights frozen [39]. In this study, only the LoRA parameters in the language decoder are updated. Training combines the generation loss over the complete JSON output with the loss over rule-token regions, while the base visual encoder remains frozen.

SAM (Segment Anything Model)

SAM refers to the Segment Anything model family. In this study, SAM3 assists annotation by identifying and segmenting objects in images using concept prompts [45]. The segmentation capabilities of the annotation tool should be distinguished from the supervision used to train RTR-Align. Its regional targets are derived from the proportion of each grid cell covered by the final bounding boxes, rather than directly from SAM3 pixel masks.

Appendix B. Cross-Model Evaluation of RTR-Align

To assess the applicability of RTR-Align across vision-language models, we applied it to three additional backbones: LLaVA-OV-2-8B [24], InternVL3.5-8B [25], and GLM-4.6V-Flash [26]. We adapted the implementation to account for differences in how the backbones arrange visual tokens and compute attention. For all four backbones, RTR-Align training used the same rule-token-to-region alignment objective proposed in this work. Table A1 in Appendix B reports the performance of each of the three additional backbones before and after RTR-Align fine-tuning.

Table A1. Cross-model evaluation of RTR-Align on construction safety violation recognition and localization. All metrics are expressed as percentages.

Backbone	Setting	Precision	Recall	F1	Matched IoU	Total IoU
LLaVA-OV-2-8B [24]	Base	3.77	33.69	6.78	43.46	3.57
	RTR-Align	41.20	44.05	42.58	59.15	26.52
InternVL3.5-8B [25]	Base	6.16	22.50	9.67	35.44	5.60
	RTR-Align	53.24	51.90	52.56	71.30	31.83
GLM-4.6V-Flash [26]	Base	7.75	60.60	13.74	69.49	6.33
	RTR-Align	58.52	61.31	59.88	75.13	37.34

Appendix C. Visual Attention Inside Violation Bounding Boxes

The uniform reference is the fraction of the total image area covered by the union of the ground-truth bounding boxes for violation instances and represents the proportion of visual attention expected within this region under a uniform distribution. Attention inside boxes is the sum of the visual attention weights assigned by the model to this region divided by the sum of the visual attention weights over the entire image. The change from baseline is calculated by subtracting the mean proportion of attention inside the boxes for Qwen3-VL-8B from the corresponding value for RTR-Align and is expressed in percentage points.

Table A2. Share of visual attention within the union of ground-truth boxes, averaged equally across all test images with annotated violations.

Model	Uniform Reference	Attention Inside Boxes	Change from Baseline
Qwen3-VL-8B [28]	8.29	7.16	—
RTR-Align	8.29	11.69	+4.53

Appendix D. Prompt Availability

The full system and user prompts used in this study are available at https://github.com/zbgao-01/RTR-Align (accessed on 4 November 2026).

References

1. Li, L.; Huang, Z.; Wang, J.; Du, B.; Dai, L. Automated construction monitoring based on computer vision: A comprehensive review. *Dev. Built Environ.* **2026**, *25*, 100832. https://doi.org/10.1016/j.dibe.2025.100832.
2. Wang, Q.A.; Huang, X.Y.; Wang, J.F.; Ni, Y.Q.; Ran, S.C.; Li, J.P.; Zhang, J. Concise Historic Overview of Rail Corrugation Studies: From Formation Mechanisms to Detection Methods. *Buildings* **2024**, *14*, 968. https://doi.org/10.3390/buildings14040968.
3. de Marco, G.; Niederwieser, E.; Siegele, D. Exploring a Multimodal Conversational Agent for Construction Site Safety: A Low-Code Approach to Hazard Detection and Compliance Assessment. *Buildings* **2025**, *15*, 3352. https://doi.org/10.3390/buildings15183352.
4. Wang, Q.A.; Cui, G.Y.; Sun, G.Q.; Ding, B.D.; Ma, Z.G.; Yang, J.M.; Gong, P.; Liu, J.; Zhu, H.Y. Multi-Level Spatial Design Decision-Making Model for Block Caving Systems in Super-Large Open-Pit Mines. *Appl. Sci.* **2026**, *16*, 6753. https://doi.org/10.3390/app16136753.
5. Adil, M.; Lee, G.; Gonzalez, V.A.; Mei, Q. Using Vision Language Models for Safety Hazard Identification in Construction. *arXiv* **2025**, arXiv:2504.09083. https://doi.org/10.48550/arXiv.2504.09083.
6. Chen, X.; Zou, Z. Are large pre-trained vision language models effective construction safety inspectors. *Data-Centric Eng.* **2026**, *7*, e11. https://doi.org/10.1017/dce.2026.10044.
7. Adil, M.; Ahmed, M.; Aqib, M.; Gonzalez, V.A.; Lee, G.; Mei, Q. Integration of Object Detection and Small VLMs for Construction Safety Hazard Identification. *arXiv* **2026**, arXiv:2604.05210. https://doi.org/10.48550/arXiv.2604.05210.
8. Luo, X.; Li, C.; Gu, B.; Fang, D. Toward fine-grained construction safety inspection via vision-language grounded reasoning. *Adv. Eng. Inform.* **2026**, *76*, 104870. https://doi.org/10.1016/j.aei.2026.104870.
9. Wang, Q.A.; Huang, X.Y.; Ni, Y.Q.; Feng, W.S.; Wang, J.F.; Qiu, S.; Zhou, P.; Li, J.P.; Luo, Y. Uncertainty-aware, high-precision multi-step prediction of structural health monitoring sensor streams under extreme typhoon events: An enhanced Bayesian dynamic linear model leveraging the kernel regression basis function for severe environmental adaptation. *Measurement* **2026**, *262*, 120050. https://doi.org/10.1016/j.measurement.2025.120050.
10. Nath, N.D.; Behzadan, A.H.; Paal, S.G. Deep learning for site safety: Real-time detection of personal protective equipment. *Autom. Constr.* **2020**, *112*, 103085. https://doi.org/10.1016/j.autcon.2020.103085.
11. Jiao, X.; Li, C.; Zhang, X.; Fan, J.; Cai, Z.; Zhou, Z.; Wang, Y. Detection Method for Safety Helmet Wearing on Construction Sites Based on UAV Images and YOLOv8. *Buildings* **2025**, *15*, 354. https://doi.org/10.3390/buildings15030354.
12. Wang, Q.A.; Lu, A.W.; Ni, Y.Q.; Wang, J.F.; Ma, Z.G. Bayesian Network in Structural Health Monitoring: Theoretical Background and Applications Review. *Sensors* **2025**, *25*, 3577. https://doi.org/10.3390/s25123577.
13. Fang, Q.; Li, H.; Luo, X.; Ding, L.; Luo, H.; Rose, T.M.; An, W. Detecting non-hardhat-use by a deep learning method from far-field surveillance videos. *Autom. Constr.* **2018**, *85*, 1–9. https://doi.org/10.1016/j.autcon.2017.09.018.
14. Ding, Y.; Liu, M.; Luo, X. Safety compliance checking of construction behaviors using visual question answering. *Autom. Constr.* **2022**, *144*, 104580. https://doi.org/10.1016/j.autcon.2022.104580.
15. Chan, C.F.; Wong, P.K.Y.; Guo, X.; Cheng, J.C.P.; Chan, J.P.C.; Leung, P.H.; Tao, X. Context-aware vision-language model agent enriched with domain-specific ontology for construction site safety monitoring. *Autom. Constr.* **2025**, *177*, 106305. https://doi.org/10.1016/j.autcon.2025.106305.
16. Guo, K.X.; Wong, P.K.Y.; Cheng, J.C.P.; Chan, C.F.; Leung, P.H.; Tao, X. Enhancing visual-LLM for construction site safety compliance via prompt engineering and Bi-stage retrieval-augmented generation. *Autom. Constr.* **2025**, *179*, 106490. https://doi.org/10.1016/j.autcon.2025.106490.
17. Wang, Y.; Luo, H.; Fang, W. An integrated approach for automatic safety inspection in construction: Domain knowledge with multimodal large language model. *Adv. Eng. Inform.* **2025**, *65*, 103246. https://doi.org/10.1016/j.aei.2025.103246.

18. Chen, Q.; Yin, X. Tailored vision-language framework for automated hazard identification and report generation in construction sites. *Adv. Eng. Inform.* **2025**, *66*, 103478. https://doi.org/10.1016/j.aei.2025.103478.
19. Wen, S.; Park, M.; Tran, D.Q.; Lee, S.; Park, S. Automated construction safety reporting system integrating deep learning-based real-time advanced detection and visual question answering. *Adv. Eng. Softw.* **2024**, *198*, 103779. https://doi.org/10.1016/j.advengsoft.2024.103779.
20. Tran, D.Q.; Aboah, A.; Jeon, Y.; Do, M.T.; Abdel-Aty, M.; Park, M.; Park, S. Visual Question Answering-based Referring Expression Segmentation for construction safety analysis. *Autom. Constr.* **2025**, *174*, 106127. https://doi.org/10.1016/j.autcon.2025.106127.
21. Cheng, S.; Qi, Y.; Wu, R.; Guan, Y. A Multimodal Agent Framework for Construction Scenarios: Accurate Perception, Dynamic Retrieval, and Explainable Hazard Reasoning. *Buildings* **2025**, *15*, 4439. https://doi.org/10.3390/buildings15244439.
22. Wang, H.; Guo, F.; Wang, Q.; Liu, J. Tandem CCV-VLM: Visual construction safety inspection based on Collaborative Cross-Verification VLM. *Adv. Eng. Inform.* **2026**, *75*, 104841. https://doi.org/10.1016/j.aei.2026.104841.
23. Kim, Y.; Kim, T.; Kim, Y.; Park, S. AVA-VLM: Adaptive Visual Attention-Vision Language Model for In-the-Wild Construction Site Monitoring. *arXiv* **2026**, arXiv:2607.05859v1. https://doi.org/10.48550/arXiv.2607.05859.
24. An, X.; Xie, Y.; Tang, F.; Yan, Y.; Tan, H.; Zhu, D.; Chen, C.; Zhao, X.; Qin, B.; Yang, K.; et al. LLaVA-OneVision-2: Towards Next-Generation Perceptual Intelligence. *arXiv* **2026**, arXiv:2605.25979. https://doi.org/10.48550/arXiv.2605.25979.
25. Wang, W.; Gao, Z.; Gu, L.; Pu, H.; Cui, L.; Wei, X.; Liu, Z.; Jing, L.; Ye, S.; Shao, J.; et al. InternVL3.5: Advancing Open-Source Multimodal Models in Versatility, Reasoning, and Efficiency. *arXiv* **2025**, arXiv:2508.18265. https://doi.org/10.48550/arXiv.2508.18265.
26. Z.ai. GLM-V: GLM-4.6V/4.5V/4.1V-Thinking. Available online: https://github.com/zai-org/GLM-V (accessed on 20 September 2026).
27. OpenAI. GPT-5.6 Sol Model. 2026. Available online: https://developers.openai.com/api/docs/models/gpt-5.6-sol (accessed on 14 October 2026).
28. Bai, S.; Cai, Y.; Chen, R.; Chen, K.; Chen, X.; Cheng, Z.; Deng, L.; Ding, W.; Gao, C.; Ge, C.; et al. Qwen3-VL Technical Report. *arXiv* **2025**, arXiv:2511.21631. https://doi.org/10.48550/arXiv.2511.21631.
29. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning, Virtual Event, 18–24 July 2021*; Proceedings of Machine Learning Research: New York, NY, USA, 2021; Volume 139, pp. 8748–8763.
30. Kim, W.; Son, B.; Kim, I. ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision. In *Proceedings of the 38th International Conference on Machine Learning, Virtual Event, 18–24 July 2021*; Proceedings of Machine Learning Research: New York, NY, USA, 2021 Volume 139, pp. 5583–5594.
31. Liu, S.; Zeng, Z.; Ren, T.; Li, F.; Zhang, H.; Yang, J.; Jiang, Q.; Li, C.; Yang, J.; Su, H.; et al. Grounding DINO: Marrying DINO with Grounded Pre-training for Open-Set Object Detection. In *Computer Vision—ECCV 2024*; Lecture Notes in Computer Science; Leonardis, A., Ricci, E., Roth, S., Russakovsky, O., Sattler, T., Varol, G., Eds.; Springer: Cham, Switzerland, 2025; Volume 15105, pp. 38–55. https://doi.org/10.1007/978-3-031-72970-6_3.
32. Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A.C.; Lo, W.Y.; et al. Segment Anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023*; IEEE: New York, NY, USA, 2023; pp. 3992–4003. https://doi.org/10.1109/ICCV51070.2023.00371.
33. Chen, K.; Zhang, Z.; Zeng, W.; Zhang, R.; Zhu, F.; Zhao, R. Shikra: Unleashing Multimodal LLM's Referential Dialogue Magic. *arXiv* **2023**, arXiv:2306.15195. https://doi.org/10.48550/arXiv.2306.15195.
34. Peng, Z.; Wang, W.; Dong, L.; Hao, Y.; Huang, S.; Ma, S.; Ye, Q.; Wei, F. Grounding Multimodal Large Language Models to the World. In Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024. Available online: https://proceedings.iclr.cc/paper_files/paper/2024/hash/e112a4671e8779aa9f640a0e3f81bd26-Abstract-Conference.html (accessed on 23 September 2026).
35. You, H.; Zhang, H.; Gan, Z.; Du, X.; Zhang, B.; Wang, Z.; Cao, L.; Chang, S.F.; Yang, Y. Ferret: Refer and Ground Anything Anywhere at Any Granularity. In Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024. Available online: https://proceedings.iclr.cc/paper_files/paper/2024/hash/fd6613131889a4b656206c50a8bd7790-Abstract-Conference.html (accessed on 23 September 2026).
36. Lai, X.; Tian, Z.; Chen, Y.; Li, Y.; Yuan, Y.; Liu, S.; Jia, J. LISA: Reasoning Segmentation via Large Language Model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 17–21 June 2024*; IEEE: New York, NY, USA, 2024; pp.9579–9589. https://doi.org/10.1109/CVPR52733.2024.00915.

37. Rasheed, H.; Maaz, M.; Shaji, S.; Shaker, A.; Khan, S.; Cholakkal, H.; Anwer, R.M.; Xing, E.; Yang, M.H.; Khan, F.S. GLaMM: Pixel Grounding Large Multimodal Model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 17–21 June 2024*; IEEE: New York, NY, USA, 2024; pp. 13009–13018. https://doi.org/10.1109/CVPR52733.2024.01236.
38. Gupta, A.; Dollár, P.; Girshick, R. LVIS: A Dataset for Large Vocabulary Instance Segmentation. In *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019*; IEEE: New York, NY, USA, 2019; pp. 5351–5359. https://doi.org/10.1109/CVPR.2019.00550.
39. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. LoRA: Low-Rank Adaptation of Large Language Models. In Proceedings of the Tenth International Conference on Learning Representations (ICLR 2022), Virtual Event, 25–29 April 2022. Available online: https://openreview.net/forum?id=nZeVKeeFYf9 (accessed on 23 September 2026).
40. Sokolova, M.; Lapalme, G. A systematic analysis of performance measures for classification tasks. *Inf. Process. Manag.* **2009**, *45*, 427–437. https://doi.org/10.1016/j.ipm.2009.03.002.
41. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The PASCAL Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. https://doi.org/10.1007/s11263-009-0275-4.
42. Liu, C.; Mao, J.; Sha, F.; Yuille, A. Attention Correctness in Neural Image Captioning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017*; AAAI Press: Palo Alto, CA, USA, 2017; pp. 4176–4182. https://doi.org/10.1609/aaai.v31i1.11197.
43. Che, L.; Liu, T.Q.; Jia, J.; Qin, W.; Tang, R.; Pavlovic, V. Hallucinatory Image Tokens: A Training-free EAZY Approach to Detecting and Mitigating Object Hallucinations in LVLMs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Honolulu, HI, USA, 19–23 October 2025*; IEEE: New York, NY, USA, 2025; pp. 21635–21644. https://doi.org/10.1109/ICCV51701.2025.02009.
44. Hu, Z.; Xu, G.; Hu, G. The Answer Knows Where to Look: Zooming before MLLMs Answer under Visual-Token Budgets. *arXiv* **2026**, arXiv:2610.32664. https://doi.org/10.48550/arXiv.2610.32664.
45. Carion, N.; Gustafson, L.; Hu, Y.T.; Debnath, S.; Hu, R.; Suris Coll-Vinent, D.; Ryali, C.; Alwala, K.V.; Khedr, H.; Huang, A.; et al. SAM 3: Segment Anything with Concepts. In Proceedings of the Fourteenth International Conference on Learning Representations (ICLR 2026), Rio de Janeiro, Brazil, 23–27 April 2026. Available online: https://proceedings.iclr.cc/paper_files/paper/2026/hash/e0982cbc81401df3430ee1ff780dc7a2-Abstract-Conference.html (accessed on 23 September 2026).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.