LOOKING WHERE INFRARED MATTERS: INSTRUCTION-DRIVEN INSTANCE SEGMENTATION IN MULTISPECTRAL VISION

Anonymous authors
Paper under double-blind review

ABSTRACT

Thermal imaging complements RGB under poor illumination, supporting perception in night-time driving, search and rescue, and industrial inspection. In these settings, instances of the same category can play different roles, and identifying the specific task-relevant ones requires descriptions of their behavior, function, or relation to the scene. Yet existing RGB–thermal segmentation produces semantic masks of category labels, leaving such distinctions unresolved. To bridge this gap, we introduce instruction-driven multispectral instance segmentation, where natural-language instructions specify which instances to segment. We therefore construct MINT, a benchmark of 5,000 registered RGB–thermal pairs with referring and reasoning instructions spanning single, multiple, all, and absent targets. Instructions express the desired target without mentioning thermal evidence, leaving the model to determine its relevance. This relevance varies across both instructions and spatial locations: helpful thermal cues coexist with harmful ones within the same scene. So, we develop a framework that learns when to consult thermal evidence and where to integrate it. Starting from RGB, a language model can emit a learned `[LOOK]` token to request regional thermal evidence and resume reasoning. An instruction-conditioned learnable gate then selects where to apply deeper thermal processing within the requested region. We supervise these decisions using intervention-derived utility estimates from a frozen full-evidence model, without manual utility annotations. Experiments on MINT show that our model outperforms both RGB-only and always-fused baselines, while processing thermal evidence in depth on only a fraction of each image. *We will make the dataset and source code publicly available.*

1 INTRODUCTION

Thermal cameras provide complementary visibility when RGB imaging becomes unreliable, supporting applications such as night-time driving assistance, search and rescue, and inspection of unlit industrial sites. In driving assistance, a pedestrian entering the lane demands different attention from those remaining on the sidewalk. Delineating that pedestrian requires distinguishing a particular instance from others of the same category. The label "person" cannot specify which individual matters; natural-language instructions express the actions and relationships that designate the target. These descriptions vary with the task and scene, beyond what a vocabulary of category names can express. Their role becomes especially important under poor illumination, when colour and texture are difficult to discern and target identification depends on posture, apparent movement, and spatial relationships.

Existing segmentation paradigms address different parts of this need. RGB–thermal semantic segmentation improves scene parsing under challenging conditions, but assigns the same category label to pedestrians regardless of their relevance to the instruction (Zhang et al., 2023a; Liu et al., 2023b). Open-vocabulary approaches expand the range of recognizable categories while retaining category-based segmentation as their objective (Yu et al., 2025). Meanwhile, language-guided RGB segmentation interprets implicit requests and supports multiple or absent targets in generalized settings (Lai et al., 2024; Xia et al., 2024). These methods establish the ability to translate user intent into segmentation masks, but their reasoning relies on RGB evidence.

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

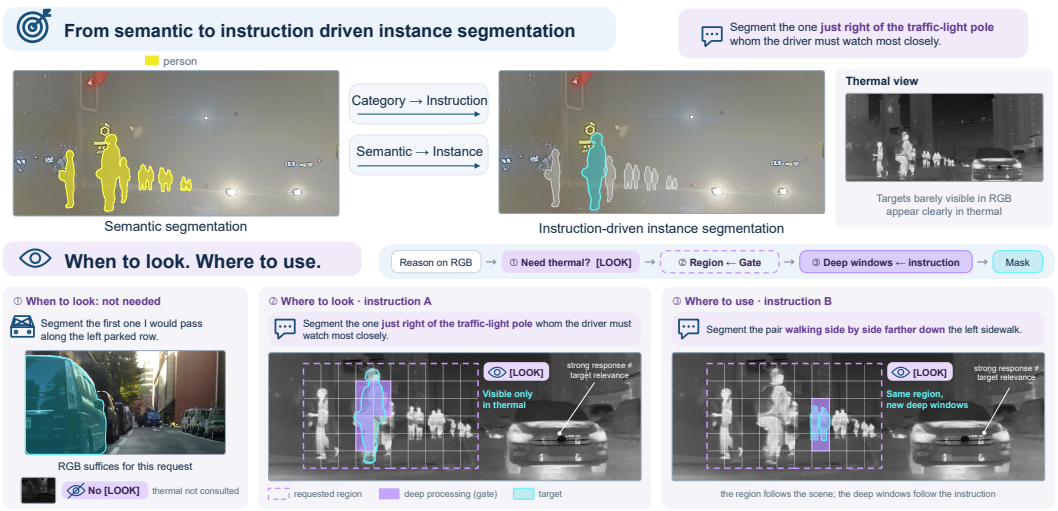

Figure 1: **From category labels to instruction-driven instance segmentation in multispectral vision. Top:** Instructions distinguish task-relevant instances within the same category, while thermal imagery reveals targets obscured in RGB. **Bottom:** Our model learns when to consult thermal evidence and where to process it. RGB suffices for the parked-vehicle request; nighttime requests trigger `[LOOK]`. Within a shared requested region, deep-processing windows shift with the instructed target, leaving the vehicle's strong but irrelevant thermal response unused.

To address this challenge, we introduce instruction-driven multispectral instance segmentation: given a registered RGB–thermal pair and a natural-language instruction, a model needs to segment the designated instance(s) or return an empty result when none satisfies the request. Developing this capability requires both data that pairs instructions with instance-level targets and a model that can determine which thermal evidence serves each instruction. A warm engine hood, for example, may produce a stronger thermal response than a nearby pedestrian while being irrelevant to a request about that pedestrian. Incorporating this response can interfere with target identification, despite its prominence in the thermal image. The value of thermal evidence therefore depends on what is being asked and where the evidence appears. This leads to the question: ***when should a model consult thermal evidence, and where should it use that evidence to satisfy the instruction?***

We design MINT (Multispectral INstruction Targets) to test target selection, contextual reasoning, and recognition of absent targets under varying visibility. Referring instructions specify targets through explicit attributes and spatial relations, while reasoning instructions require interpreting function, intent, or context. Four target-cardinality settings—one-to-one, one-to-many, one-to-all, and one-to-none—test whether a model can select the appropriate subset of instances, including the empty set. We curate 5,000 registered RGB–thermal image pairs, derive instance masks from their semantic annotations, and collect paired referring and reasoning instructions. The annotation collection combines generation by multiple vision-language models with a manually reviewed and corrected track. Crucially, supporting chains of thought may draw on both images, while instructions describe the desired targets without explicitly referring to thermal evidence. A model is therefore never told that thermal is relevant; it must determine that for itself.

Our method makes thermal consultation an explicit action within the reasoning process. The language model initially observes only RGB and can emit a learned `[LOOK]` token to request thermal evidence from a predicted region. Without a request, the thermal branch is not executed. Upon a request, shallow adapters introduce thermal information within the requested region, and their outputs are converted into spatially positioned tokens that allow the language model to continue reasoning. An instruction-conditioned gate then selects windows for deeper thermal processing according to their predicted utility for segmentation. Deep adapters execute only on these windows before the final mask is produced. To learn these decisions without manual utility annotations, we train and freeze a full-evidence model, then estimate how suppressing its adapter outputs changes the task loss. Suppressing both shallow and deep outputs measures the value of the requested evidence; suppressing deep outputs alone measures the additional value of deeper processing. These intervention-derived targets teach the model when to request thermal evidence and where to apply deeper processing according to their contribution to the task.

2 Related Work

Reasoning and Referring Segmentation. Referring segmentation grounds explicit attributes and spatial relations through dedicated vision–language architectures (Hu et al., 2016; Yang et al., 2022; Wang et al., 2022); reasoning segmentation extends this to instructions whose targets must be inferred from function, intent, or context (Li et al., 2025; Guo et al., 2026). LISA (Lai et al., 2024) links multimodal reasoning to mask prediction via a learned `[SEG]` token, and subsequent work adds instance-level outputs, multi-target pixel reasoning, grounded generation, and reinforcement learning (Yang et al., 2023; Ren et al., 2024; Rasheed et al., 2024; Liu et al., 2025). GRES (Liu et al., 2023a) generalizes the setting to multiple and absent targets, and GSVA (Xia et al., 2024) adds multiple mask references and explicit rejection. In parallel, active visual reasoning lets MLLMs search image regions (Wu & Xie, 2024) or invoke visual tools mid-reasoning (Zheng et al., 2026), establishing observation as an active component of inference, but always within RGB. None of this addresses when an instruction should trigger access to a second modality, or how that evidence should then be integrated. We make that decision explicit: the model reasons from RGB, requests regional thermal evidence when needed, and lets the instruction guide the subsequent spatial integration.

RGB–Thermal Segmentation. MFNet and RTFNet establish multispectral scene parsing (Ha et al., 2017; Sun et al., 2019), with PST900, MSRS, and SemanticRT providing semantic annotations (Shivakumar et al., 2020; Tang et al., 2022; Ji et al., 2023). Subsequent work develops Transformer-based feature rectification (Zhang et al., 2023a), flexible multimodal extraction (Zhang et al., 2023b), and joint fusion and segmentation (Tang et al., 2022; Liu et al., 2023b). SpiderMesh (Fan et al., 2023) masks regions where RGB semantics are inadequate; M-SpecGene (Zhou et al., 2025) learns representations under cross-modal imbalance; Open-RGBT (Yu et al., 2025) extends recognition to open-vocabulary categories. These methods adapt fusion to the scene, while thermal utility also depends on the instruction: evidence supporting one target may distract from another. We learn this utility through adapter-output interventions, guiding evidence requests and spatial processing. MM-RIS (Ma et al., 2025) introduces RGB–thermal referring segmentation, but constructs targets as connected components of semantic masks. Components distinguish separated regions, yet cannot separate touching instances of the same category. Our SAM-guided watershed construction and human correction resolve these instance boundaries, enabling instructions to select individual objects and subsets beyond disconnected semantic regions.

3 Instruction-driven Multispectral Instance Segmentation

3.1 Problem Definition

Instruction-driven multispectral instance segmentation asks a model to segment the instances that satisfy a natural-language request using registered RGB–thermal imagery. The instruction can designate targets through observable attributes and spatial relations, or require interpreting their function, behavior, and role in the scene.

Given an RGB image $I_{\mathrm{rgb}} \in \mathbb{R}^{H\times W\times 3}$, a registered thermal image $I_{\mathrm{t}} \in \mathbb{R}^{H\times W}$, and an instruction S, let $\mathcal{O} = \{o_i\}_{i=1}^{N}$ denote the annotated instances, with binary masks $m_i \in \{0,1\}^{H\times W}$. The instruction determines the target index set

$$\mathcal{J}(S) = \big\{\, i \in \{1,\dots,N\} \;:\; o_i \text{ satisfies } S \,\big\}. \tag{1}$$

The desired output is the corresponding set of instance masks, $\mathcal{Y}(S) = \{\, m_i : i \in \mathcal{J}(S) \,\}$, whose aggregate foreground is

$$M_S(p) = \max_{i\in\mathcal{J}(S)} m_i(p), \tag{2}$$

where p indexes a pixel and the maximum over an empty set is taken as zero.

Equation 1 places no constraint on the size of $\mathcal{J}(S)$: the instruction alone decides how many instances of a category are wanted. This selection scope is a property of the request rather than of the number of instances a scene happens to contain, and we distinguish four scopes: *one-to-one* singles out an individual among its peers; *one-to-many* selects a qualifying subset; *one-to-all* takes the entire category; and *one-to-none* matches no instance, leaving $\mathcal{J}(S) = \varnothing$ and M_S all-zero.

Table 1: Comparison with existing segmentation benchmarks. Ours is the only one with instance-level targets, referring and reasoning instructions, all four selection scopes, and thermal evidence. †Targets are connected components of the semantic masks, so touching objects of one category remain a single target.

Benchmark	Target	Instruction level		Instruction granularity				Thermal	#Images	#Masks	#Instructions
	Instance	*Referring*	*Reasoning*	*1-to-1*	*1-to-many*	*1-to-all*	*1-to-none*				
RefCOCO	✔	✔	✗	✔	✗	✗	✗	✗	20.0k	50.0k	142k
gRefCOCO	✔	✔	✗	✔	✔	✗	✔	✗	20.0k	60.3k	278k
ReasonSeg	✔	✗	✔	✔	✗	✗	✗	✗	1.2k	1.2k	1.2k
MUSE	✔	✗	✔	✔	✔	✗	✗	✗	74k	910k	246k
MFNet	✗	✗	✗	✗	✗	✗	✗	✔	1.6k	–	–
PST900	✗	✗	✗	✗	✗	✗	✗	✔	0.9k	–	–
MSRS	✗	✗	✗	✗	✗	✗	✗	✔	1.4k	–	–
SemanticRT	✗	✗	✗	✗	✗	✗	✗	✔	11.4k	–	–
MM-RIS	✗†	✔	✗	✔	✔	✗	✗	✔	2.9k	16.2k	16.2k
MINT (Ours)	✔	✔	✔	✔	✔	✔	✔	✔	5.0k	37.8k	42.0k

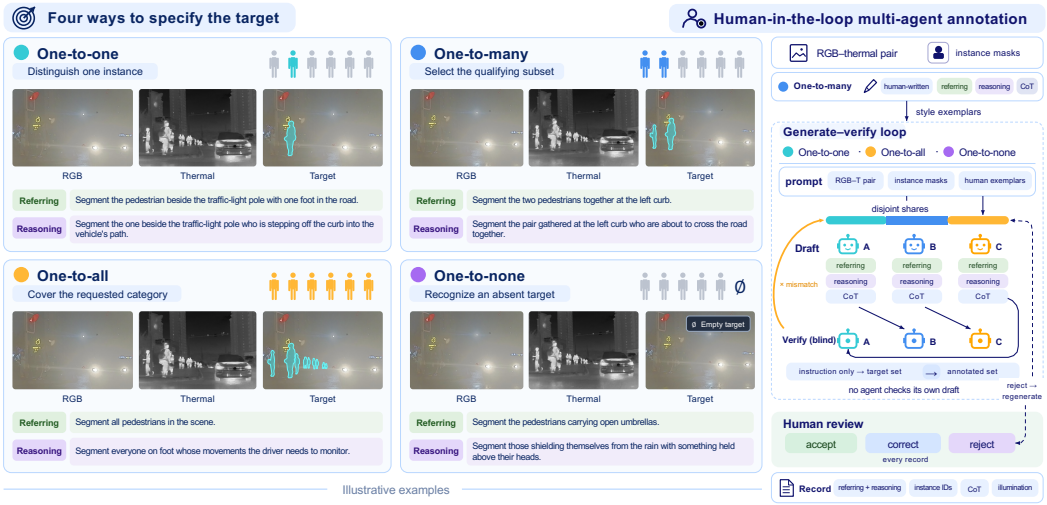

Figure 2: **Left:** one scene under the four selection scopes, each with a referring and a reasoning instruction for the same targets. **Right:** the human-in-the-loop multi-agent annotation pipeline.

3.2 MINT Benchmark

Our benchmark advances multispectral segmentation in two connected ways. First, selecting a particular object among others of the same category requires instance-level masks, rather than the semantic ones. Second, specifying which objects matter requires instructions that describe their attributes, behavior, function, or relations to the scene. Such descriptions become especially useful under poor illumination, when appearance cues such as color and texture are difficult to discern. We therefore curate 5,000 high-quality registered RGB–thermal image pairs from MSRS, PST900, and SemanticRT, and annotate them with instance-level targets and natural-language instructions. Instance masks are derived using ***SAM-guided morphological processing***, followed by human inspection and correction, while instructions are annotated by our ***human-in-the-loop multi-agent annotation system***. Table 1 compares our benchmark with existing datasets. Each annotation pairs a referring instruction with a reasoning instruction for the same target set. Referring instructions identify targets through explicit attributes and spatial relations, while reasoning instructions require interpreting function, intent, or context. We include all four selection scopes defined in Section 3.1: one-to-one, one-to-many, one-to-all, and one-to-none. Across all settings, instructions describe the desired targets without mentioning thermal evidence, while supporting chains of thought may use both images to explain the selection.

One-to-many instructions and rationales are written by human annotators to capture precise distinctions among same-category instances, and serve as exemplars for the other settings. For these settings, multiple VLMs generate annotations to diversify phrasing, followed by blind cross-model verification of target selection and human review of every record. MINT uses image-disjoint splits of

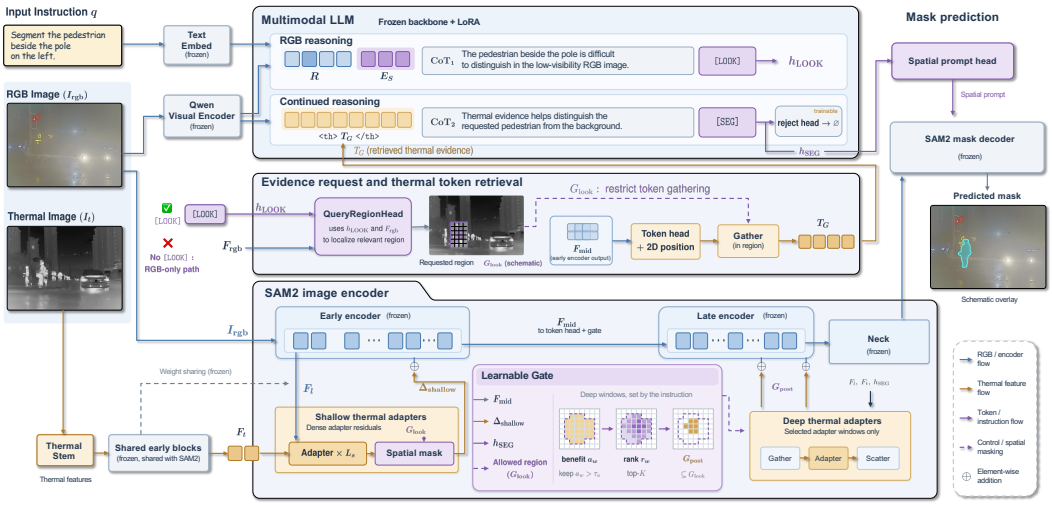

Figure 3: **Overview of our framework.** [LOOK] requests regional thermal evidence for continued reasoning. Shallow adapters supply evidence tokens, while an instruction-conditioned gate selects deep-processing windows; [SEG] guides mask prediction.

3,460, 612, and 928 image pairs for training, validation, and test, respectively. Construction details are provided in Appendix A.

4 METHOD

4.1 MODEL ARCHITECTURE

Overall architecture. Our framework makes thermal consultation an explicit action within instruction-driven reasoning. As illustrated in Figure 3, a vision-language model begins with RGB and can request regional thermal evidence through a learned [LOOK] token. The returned evidence informs continued reasoning, whose final [SEG] state guides both deeper thermal processing and segmentation.

We realize this interaction through shallow and deep adapters in a frozen SAM2 encoder (Ravi et al., 2025). Shallow adapters provide evidence for language reasoning, while deep adapters use the resulting segmentation state to refine selected windows. The visual backbones and SAM2 decoder remain frozen, and the language model is adapted through LoRA.

When and where to look: thermal evidence requests during reasoning. The language model initially receives RGB visual tokens R and instruction embeddings E_S. Emitting [LOOK] initiates consultation. Its hidden state h_{LOOK} queries RGB-only early SAM2 features F^0_{rgb} to predict a spatial request on a 16×16 window grid:

$$\begin{aligned} K_{\text{look}} &= \mathcal{T}_{\text{look}}\big(W_k\,\text{Pool}_{16}(F^0_{\text{rgb}}) + P\big)\,, \\ \ell_w &= \frac{(W_q h_{\text{LOOK}})^\top K_{\text{look},w}}{\sqrt{d}} + b, \\ G_{\text{look},w} &= \mathbb{1}[\sigma(\ell_w) > \tau_{\text{look}}]\,, \end{aligned} \tag{3}$$

where P encodes spatial position and $\mathcal{T}_{\text{look}}$ contextualizes the 16×16 window grid. This grounds the request in the RGB scene before thermal processing begins.

Upon a request, a thermal stem and shared frozen early encoder blocks extract thermal features F_{t}. Shallow adapters compute bottleneck residuals Δ_{shallow} from the RGB and thermal features; these are masked by an upsampled, feathered G_{look} and added to the RGB stream, yielding the early fused representation F_{mid}. A token head projects this representation into the language embedding space, so that the gathered tokens carry both RGB context and the thermal contribution at the requested cells:

$$T_G = \text{Gather}(H_{\text{tok}}(F_{\text{mid}}) + P_{\text{tok}},\, G_{\text{look}})\,. \tag{4}$$

Positional encoding is added before gathering to preserve each token's image location. The evidence tokens are inserted after `[LOOK]`, producing the sequence C_1; `[LOOK]`; `<th>`; T_G; `</th>`; C_2; `[SEG]`, where C_1 and C_2 are the reasoning before and after retrieval. We allow one request per instruction. Without `[LOOK]`, the model proceeds using RGB alone and the thermal pathway is not executed.

Where to deep process: instruction-conditioned deep thermal allocation. Within a requested region, the gate must distinguish beneficial thermal contributions from harmful ones and prioritize the useful windows for deeper processing. We therefore predict a benefit score for filtering and a ranking score for allocation, both conditioned on the segmentation state (h_{SEG}). After evidence retrieval and continued reasoning, an instruction-conditioned gate allocates deeper processing within the requested region. It reads contextualized window representations of F_{mid} and the shallow residual $\Delta_{\mathrm{shallow}}$, using h_{SEG} as the query:

$$\begin{aligned}(r, a) &= g_\phi(F_{\mathrm{mid}}, \Delta_{\mathrm{shallow}}, h_{\mathrm{SEG}}),\\ \mathcal{A} &= \{\, w : G_{\mathrm{look},w} = 1,\ a_w > \tau_a \,\},\\ \mathcal{W} &= \mathrm{TopK}_{w\in\mathcal{A}}\ r_w, \qquad G_{\mathrm{post},w} = \mathbb{1}[w \in \mathcal{W}].\end{aligned} \tag{5}$$

Both r_w (priority) and a_w (benefit) are scaled dot products between an h_{SEG} query and the window keys; r is centred within each image, whereas a keeps its absolute scale so that its sign separates helpful from harmful windows. At most K eligible windows are kept, $G_{\mathrm{post}} \subseteq G_{\mathrm{look}}$.

Deep adapters gather only the selected windows, combine their visual and thermal features through cross-attention and FiLM conditioning on h_{SEG}, and scatter residuals back into the encoder stream; all adapters use zero-initialized output projections. The late encoder and neck produce F_{sam}, and, following CoPRS (Lu et al., 2026), a spatial prompting head combines F_{sam} and h_{SEG} into a dense prompt for the frozen SAM2 decoder. A rejection head on h_{SEG} handles instructions with no satisfying target.

4.2 Supervision and Learning

Intervention-derived utility. We supervise evidence use by its contribution to a trained model's predictions (Figure 4). A full-evidence reference model is trained and then frozen. Removing its adapter outputs defines two interventions:

$$u_i^\star = \mathcal{L}_{\mathrm{joint}}\big(\mathrm{KO}_i^{\mathrm{sh+deep}}\big) - \mathcal{L}_{\mathrm{joint}}(\mathrm{full}), \qquad u_w^{\mathrm{post},\star} = \mathcal{L}_{\mathrm{seg}}\big(\mathrm{KO}_w^{\mathrm{deep}}\big) - \mathcal{L}_{\mathrm{seg}}(\mathrm{full}). \tag{6}$$

The first removes shallow and deep residuals within instance i; the second removes only deep residuals in window w. Positive utility means that removal worsens the prediction, negative that it improves it. Since exact removal needs one pass per instance or window, we train on integrated-gradient surrogates u_i and u_w^{post} (Sundararajan et al., 2017) over adapter-output activations, aggregated by instance and by window, and validate them against exact removal (Appendix B). Deep-window utilities are recomputed on the request-trained model inside its own predicted region, so that the gate is supervised in the evidence context in which it operates.

Request supervision. Whether to request and where to request are supervised separately. For a target-bearing example, let

$$\begin{aligned}d_S &= \mathrm{IoU}\big(\widehat{M}_S^{\mathrm{full}}, M_S\big) - \mathrm{IoU}\big(\widehat{M}_S^{\mathrm{none}}, M_S\big), \qquad G_{\mathrm{look}}^\star = \mathcal{D}\Big(\Pi_\kappa\Big(\textstyle\bigcup_{i:\,u_i>\tau_u} m_i\Big)\Big),\\ y_{\mathrm{look}} &= \mathbb{1}[d_S > \delta]\ \mathbb{1}\big[\|G_{\mathrm{look}}^\star\|_1 > 0\big],\end{aligned} \tag{7}$$

where both predictions come from the frozen reference model, with and without thermal evidence; Π_κ projects the instance union onto the request grid with a cell-coverage threshold κ, and $\mathcal{D}$ dilates it by one cell. The gain d_S decides whether to request; the instance utilities decide where. For positive examples, we insert `[LOOK]` after the first sentence of the rationale and place the evidence tokens right after it; zero-request trajectories, used also for instructions without a target, omit both, so request and reasoning are supervised in one causal sequence.

Learning objectives. Request learning combines action generation, spatial localization, and target grounding:

$$\mathcal{L}_{\mathrm{request}} = \mathcal{L}_{\mathrm{trace}} + \lambda_{\mathrm{region}}\mathcal{L}_{\mathrm{region}} + \lambda_{\mathrm{seg}}\mathcal{L}_{\mathrm{seg}} + \lambda_{\mathrm{reject}}\mathcal{L}_{\mathrm{reject}}. \tag{8}$$

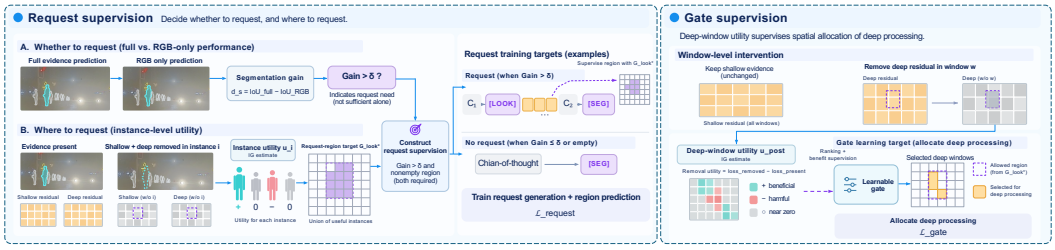

Figure 4: **Intervention-derived supervision. Left:** request supervision. The gain of full evidence over RGB decides whether to request; the instances whose thermal evidence carries positive utility, dilated by one cell, decide where. **Right:** gate supervision. Removing deep residuals window by window gives signed utilities that train the ranking and benefit heads.

The trace loss is the negative log-likelihood of text and action tokens, excluding injected evidence; region supervision combines binary cross-entropy and Dice. The gate learns both the ordering and the sign of deep-window utility:

$$\begin{aligned} \mathcal{L}_{\text{gate}} &= \lambda_r \mathcal{L}_{\text{rank}} + \lambda_a \mathcal{L}_{\text{sign}} + \lambda_m \mathcal{L}_{\text{seg}}(\mathcal{W}) + \lambda_n \mathcal{L}_{\text{nec}}, \\ \mathcal{L}_{\text{nec}} &= \text{softplus}\big(\mathcal{L}_{\text{seg}}(\mathcal{W}) - \mathcal{L}_{\text{seg}}(\varnothing)\big), \end{aligned} \tag{9}$$

where $\mathcal{L}_{\text{seg}}(\varnothing)$ is the loss without deep processing. Pairwise ranking within the requested region aligns r_w with the ordering of u_w^{post}, and sign supervision trains a_w on clearly beneficial and harmful windows. The necessity term ties both scores to the segmentation they produce: processing the selected windows must cost less than processing none of them.

Training procedure. We train the full-evidence model (adapters, token head, LoRA) while dropping 30% of the thermal cells at random, freeze it, and derive utilities and request trajectories. Request learning then updates LoRA, special-token embeddings, and the region head, with deep processing covering the requested region. Finally, we recompute deep-window utilities inside the predicted region and train the gate alone with hard selection and a straight-through gradient. Training is teacher-forced; inference runs the actual request, retrieval, and continuation loop.

5 Experiments

Thermal evidence serves an instruction best when selected rather than applied everywhere. Existing systems fall short even with the thermal image in view (§5.2); two levels of selection explain the difference: the scene decides where to look, the instruction where to process (§5.3).

5.1 Experimental Setup

Data and evaluation. We evaluate on MINT using image-disjoint splits of 3,460, 612, and 928 image pairs for training, validation, and test, respectively. We report mIoU over queries with a nonempty target set, gIoU from accumulated intersections and unions, and no-target accuracy (N-acc) on one-to-none queries, all under free-running inference.

Compared methods. LISA (Lai et al., 2024) and VisionReasoner (Liu et al., 2026) represent instruction-driven segmentation from RGB; VisionReasoner shares our Qwen2.5-VL-7B and SAM2 and may return several instances or none, and we also give it the thermal image as a second input. RIS-FUSION (Ma et al., 2025) represents RGB–thermal referring segmentation. No RGB–thermal model follows free-form instructions, so we pair the RGB–thermal semantic segmenter Sigma (Wan et al., 2025) with Qwen2.5-VL-7B, which maps each instruction to a category name. All are fine-tuned on the MINT training split (Appendix D).

Implementation. We use Qwen2.5-VL-7B-Instruct (Bai et al., 2025) and SAM2.1 Hiera-B+ (Ravi et al., 2025) split at block 13, a 16×16 window grid, $\tau_{\text{look}} = 0.1$, and $K = 32$ (Eq. 5); details are in Appendix C.

Table 2: **Comparison with existing methods.** §Instruction mapped to a category name.

Method	Input	LLM	mIoU	gIoU	Instruction		Selection scope			Night	N-acc
					Refer.	Reason.	1-to-1	1-to-many	1-to-all		
LISA-7B (Lai et al., 2024)	RGB	LLaVA-7B	35.4	38.1	39.3	31.5	32.9	28.1	39.2	33.8	–
VisionReasoner-7B (Liu et al., 2026)	RGB	Qwen2.5-VL-7B	38.9	41.7	41.3	36.5	36.9	34.2	41.6	37.6	71.9
VisionReasoner-7B (Liu et al., 2026)	RGB+T	Qwen2.5-VL-7B	42.6	45.0	45.2	40.0	40.9	36.8	45.3	42.9	73.6
RIS-FUSION (Ma et al., 2025)	RGB-T	–	45.2	47.0	51.6	38.8	44.6	36.2	47.4	41.9	–
Qwen2.5-VL → Sigma§ (Wan et al., 2025)	RGB-T	Qwen2.5-VL-7B	57.1	60.6	60.2	54.0	52.0	41.0	64.9	53.9	38.5
Ours	RGB-T	Qwen2.5-VL-7B	65.6	68.4	66.0	65.2	64.8	69.3	65.7	61.1	82.6

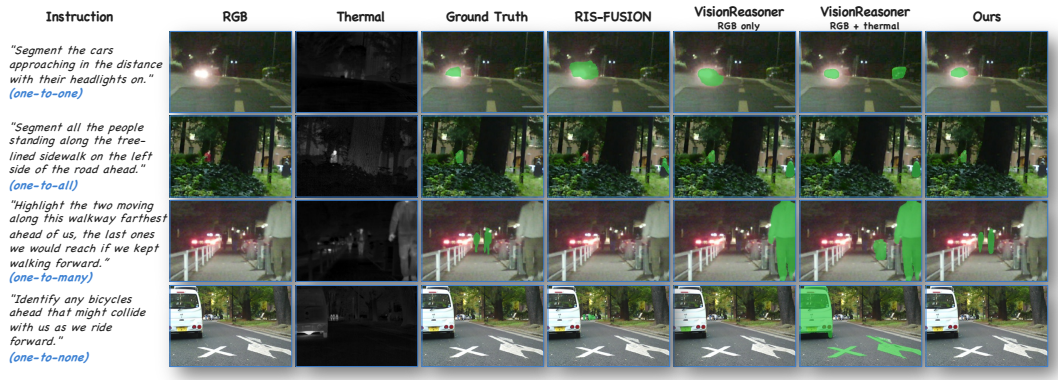

Figure 5: **Qualitative comparison.** One instruction and its selection scope per row.

5.2 Comparison with Existing Methods

Quantitative comparison. Our model leads all compared methods, by 8.5 mIoU over the strongest (Table 2). That method, a thermal-aware semantic segmenter driven by a category name, shows what category labels can and cannot do: it nears ours on one-to-all queries (64.9 versus 65.7) but trails by 12.8 on one-to-one and 28.3 on one-to-many, where instances must be singled out. Instruction-driven RGB models stay near our RGB-only variant (38.9 for VisionReasoner, 38.8 in Table 4). With the thermal image as a second input, VisionReasoner gains only 3.7 (5.3 at night): seeing thermal evidence is not knowing which of it serves the instruction. RIS-FUSION, built for short referring expressions, drops from 51.6 on referring to 38.8 on reasoning instructions.

Qualitative comparison. Without knowing which evidence the instruction needs, the compared models follow salient appearance (Figure 5): headlight glare, the nearest walker, or, for VisionReasoner with RGB+T, a van when no instance qualifies. Ours does not.

5.3 Ablation and Analysis

From thermal availability to effective use. Access to thermal evidence is not enough, because not all of it serves the current instruction. Thermal input itself is essential: always-fused processing, which runs thermal adapters over the entire image, exceeds RGB-only by 26.4 mIoU (Table 4), more at night than by day (20.9 versus 18.3) and most in unlit PST900 scenes (48.7). Yet applying it everywhere is not its best use: our model surpasses always-fused processing by 0.4 mIoU while admitting thermal evidence from 41.6% of the image and processing 12.5% of its windows in depth. Fed only to the decoder (naive dual fusion), thermal features trail always-fused processing by 2.9: the evidence must shape the visual representation.

Table 3: **Ablations.** Top: Δ mIoU w.r.t. the complete model (*given the same regions). Bottom: choice of deep windows.

Variant	Δ mIoU
Where to look	
no request training*	−13.2
region from h_{LOOK} alone	−4.6
one-shot region	−2.3
no zero-request trajectories	−1.4
Where to process	
no utility supervision	−1.9
one score for rank and sign	−0.7
deep adapters tuned jointly	+0.2

Deep windows	#	mIoU
none	0	55.9
random	32	63.2
whole region	106	64.5
selected by the gate	32	65.6

Table 4: **Ablation of thermal evidence use.** Coverage: share of the grid whose thermal evidence reaches the model (Thermal) or is processed in depth (Deep), averaged over requesting queries. The last two rows replace the gate by random windows or an instruction-blind gate.

	Coverage (%)				Instruction		Selection scope				
Variant	Thermal	Deep	mIoU	gIoU	Refer.	Reason.	1-to-1	1-to-many	1-to-all	Night	N-acc
RGB-only	0	0	38.8	41.6	38.9	38.7	38.6	43.2	38.2	39.2	71.3
Naive dual fusion	100	–	62.3	64.6	62.8	61.8	61.0	65.9	62.9	58.0	76.2
Always-fused	100	100	65.2	68.0	65.4	65.0	64.7	69.2	64.9	60.1	79.5
Ours, whole region deep	41.6	41.6	64.5	66.8	65.0	64.0	63.8	67.4	64.6	59.6	80.8
Ours	41.6	12.5	65.6	68.4	66.0	65.2	64.8	69.3	65.7	61.1	82.6
random windows	41.6	12.5	63.2	65.9	63.5	62.9	61.9	67.1	63.7	58.4	80.1
instruction-blind gate	41.6	12.5	64.1	66.2	64.7	63.5	63.3	67.9	64.2	59.4	81.3

When and where to look? The request places thermal evidence where the scene makes it informative. Against random regions of equal size, the learned region leads at every coverage, by 5.8 mIoU at the 41.6% it requests, where it recovers 97% of the always-fused gain. The model also learns when no evidence is needed: it skips the request on 9.3% of queries, mostly by day (17.6% versus 3.9% at night), at a cost of 0.3 mIoU on those queries. Regions requested for different targets in one image overlap by 0.95, close to 0.98 under rewording (Figure 6). The region covers the target and the candidates needed to distinguish it; the gate then allocates deep processing by instruction.

Where to deep process? The gate makes deep processing specific to the instruction. Its selected windows have a Jaccard overlap of 0.83 under rewording but only 0.46 when the target changes within the same image (0.19 for random selection; Figure 6). An instruction-blind gate retains an overlap of 0.88 across targets and falls 1.5 mIoU behind ours at the same budget (Table 4). The advantage over random selection grows as the budget decreases, reaching 3.6 mIoU at $K = 16$. Selecting also beats processing everything: deep processing of the whole region instead of the gate's windows costs 1.1 mIoU (Tables 4 and 3), so extra thermal processing can hurt the instruction.



Figure 6: **Where to look vs. where to process.** Two instructions on one image: the `[LOOK]` region barely changes; the gate's deep windows follow the target. Night RGB brightened for display.

Component ablations. Both decisions depend on how they are learned (Table 3, top). The request needs its own training (13.2 mIoU), a region grounded in RGB windows rather than regressed from the `[LOOK]` state (4.6), and the reasoning that precedes it (2.3 for a region fixed beforehand); zero-request trajectories keep it from requesting everywhere (1.4). The gate needs utility supervision (1.9) and a separate benefit head (0.7): how much a window matters does not tell whether it helps. Tuning the deep adapters jointly brings no meaningful gain (+0.2), so they stay frozen.

6 Conclusion

We introduced instruction-driven multispectral instance segmentation, its benchmark, and a model that decides which thermal evidence each instruction needs. Selecting this spatial, signed, target-dependent evidence by scene and by instruction outperforms applying it everywhere. Any model that consults a modality on demand faces the same question: not only whether to look, but which evidence serves the request.

AI USE STATEMENT

Large language models were used to construct the benchmark. Annotation agents based on `Claude Opus 4.7`, `ChatGPT 5.5`, and `Gemini3 Pro` draft the instructions and rationales of the one-to-one, one-to-all, and one-to-none settings from human-written exemplars, and a different model checks each draft by selecting its targets independently (Appendix A.2); instance masks are derived with SAM (Appendix A.1). One-to-many instructions are written by human annotators, and human reviewers verify every record and its masks before acceptance. The authors take full responsibility for the benchmark and for all content of this paper.

ETHICS STATEMENT

The benchmark is built on the public RGB–thermal datasets MSRS, PST900, and SemanticRT; we collect no new images and use them under their original terms. The annotations refer to people only through their position, appearance, and behaviour in the scene and contain no identity information. Instruction-driven thermal perception serves night-time driving, search and rescue, and industrial inspection, but thermal sensing of people can also serve surveillance; we will release the benchmark and models for research use, under terms consistent with the licenses of the source datasets.

REPRODUCIBILITY STATEMENT

Appendix A describes how instance masks, instructions, and splits are constructed and reports the benchmark statistics. Appendix B defines the utility estimates and validates them against exact removal. Appendix C specifies every module, training stage, loss weight, and threshold, together with trajectory construction and inference, and Appendix D states how each compared method is run and how each metric is computed. All compared methods use released checkpoints. We will release the benchmark with its splits, the annotation and verification prompts, the code for mask derivation, training, and evaluation, and the trained models.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Siqi Fan, Zhe Wang, Yan Wang, and Jingjing Liu. Spidermesh: Spatial-aware demand-guided recursive meshing for rgb-t semantic segmentation. *arXiv preprint arXiv:2303.08692*, 2023.

Yuchen Guo, Junli Gong, Hongmin Cai, Yiu-ming Cheung, and Weifeng Su. Can segmentation models understand the world? towards proactive affordance reasoning via visual chain-of-thought. *arXiv preprint arXiv:2605.27764*, 2026.

Qishen Ha, Kohei Watanabe, Takumi Karasawa, Yoshitaka Ushiku, and Tatsuya Harada. Mfnet: Towards real-time semantic segmentation for autonomous vehicles with multi-spectral scenes. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5108–5115. IEEE, 2017.

Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. Segmentation from natural language expressions. In *European conference on computer vision*, pp. 108–124. Springer, 2016.

Wei Ji, Jingjing Li, Cheng Bian, Zhicheng Zhang, and Li Cheng. Semanticrt: A large-scale dataset and method for robust semantic segmentation in multispectral images. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 3307–3316, 2023.

Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9579–9589. IEEE, 2024.

Jingjing Li, Yue Feng, Yuchen Guo, Jincai Huang, Wei Ji, Qi Bi, Yongri Piao, Miao Zhang, Xiaoqi Zhao, Qiang Chen, et al. Sam3-i: Segment anything with instructions. *arXiv preprint arXiv:2512.04585*, 2025.

Chang Liu, Henghui Ding, and Xudong Jiang. Gres: Generalized referring expression segmentation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23592–23601. IEEE, 2023a.

Jinyuan Liu, Zhu Liu, Guanyao Wu, Long Ma, Risheng Liu, Wei Zhong, Zhongxuan Luo, and Xin Fan. Multi-interactive feature learning and a full-time multi-modality benchmark for image fusion and segmentation. In *2023 IEEE/CVF international conference on computer vision (ICCV)*, pp. 8081–8090. IEEE, 2023b.

Yuqi Liu, Bohao Peng, Zhisheng Zhong, Zihao Yue, Fanbin Lu, Bei Yu, and Jiaya Jia. Seg-zero: Reasoning-chain guided segmentation via cognitive reinforcement. *arXiv preprint arXiv:2503.06520*, 2025.

Yuqi Liu, Tianyuan Qu, Zhisheng Zhong, Bohao Peng, Shu Liu, Bei Yu, and Jiaya Jia. Vision-Reasoner: Unified visual perception and reasoning via reinforcement learning. In *International Conference on Learning Representations*, 2026.

Zhenyu Lu et al. CoPRS: Learning positional prior from chain-of-thought for reasoning segmentation. In *International Conference on Learning Representations*, 2026.

Siju Ma, Changsiyu Gong, Xiaofeng Fan, Yong Ma, and Chengjie Jiang. RIS-FUSION: Rethinking text-driven infrared and visible image fusion from the perspective of referring image segmentation. *arXiv preprint arXiv:2509.12710*, 2025.

Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13009–13018. IEEE, 2024.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. In *International Conference on Learning Representations*, volume 2025, pp. 28085–28128, 2025.

Zhongwei Ren, Zhicheng Huang, Yunchao Wei, Yao Zhao, Dongmei Fu, Jiashi Feng, and Xiaojie Jin. Pixellm: Pixel reasoning with large multimodal model. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26364–26373. IEEE, 2024.

Shreyas S Shivakumar, Neil Rodrigues, Alex Zhou, Ian D Miller, Vijay Kumar, and Camillo J Taylor. Pst900: Rgb-thermal calibration, dataset and segmentation network. In *2020 IEEE international conference on robotics and automation (ICRA)*, pp. 9441–9447. IEEE, 2020.

Yuxiang Sun, Weixun Zuo, and Ming Liu. Rtfnet: Rgb-thermal fusion network for semantic segmentation of urban scenes. *IEEE Robotics and Automation Letters*, 4(3):2576–2583, 2019.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pp. 3319–3328. PMLR, 2017.

Linfeng Tang, Jiteng Yuan, and Jiayi Ma. Image fusion in the loop of high-level vision tasks: A semantic-aware real-time infrared and visible image fusion network. *Information Fusion*, 82: 28–42, 2022.

Zifu Wan, Pingping Zhang, Yuhao Wang, Silong Yong, Simon Stepputtis, Katia Sycara, and Yaqi Xie. Sigma: Siamese mamba network for multi-modal semantic segmentation. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, 2025.

Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. Cris: Clip-driven referring image segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11676–11685. IEEE, 2022.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13084–13094. IEEE, 2024.

Zhuofan Xia, Dongchen Han, Yizeng Han, Xuran Pan, Shiji Song, and Gao Huang. Gsva: Generalized segmentation via multimodal large language models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3858–3869. IEEE, 2024.

Senqiao Yang, Tianyuan Qu, Xin Lai, Zhuotao Tian, Bohao Peng, Shu Liu, and Jiaya Jia. Lisa++: An improved baseline for reasoning segmentation with large language model. *arXiv preprint arXiv:2312.17240*, 2023.

Zhao Yang, Jiaqi Wang, Yansong Tang, Kai Chen, Hengshuang Zhao, and Philip HS Torr. Lavt: Language-aware vision transformer for referring image segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18134–18144. IEEE, 2022.

Meng Yu, Yufeng Yue, Luojie Yang, Xunjie He, Yi Yang, and Mengyin Fu. Open-rgbt: Open-vocabulary rgb-t zero-shot semantic segmentation in open-world environments. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–7. IEEE, 2025.

Jiaming Zhang, Huayao Liu, Kailun Yang, Xinxin Hu, Ruiping Liu, and Rainer Stiefelhagen. Cmx: Cross-modal fusion for rgb-x semantic segmentation with transformers. *IEEE Transactions on intelligent transportation systems*, 24(12):14679–14694, 2023a.

Jiaming Zhang, Ruiping Liu, Hao Shi, Kailun Yang, Simon Reiß, Kunyu Peng, Haodong Fu, Kaiwei Wang, and Rainer Stiefelhagen. Delivering arbitrary-modal semantic segmentation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1136–1147. IEEE, 2023b.

Ziwei Zheng, Minghao Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, and Chao Shen. Deepeyes: Incentivizing" thinking with images" via reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 126775–126798, 2026.

Kailai Zhou, Fuqiang Yang, Shixian Wang, Bihan Wen, Chongde Zi, Linsen Chen, Qiu Shen, and Xun Cao. M-specgene: Generalized foundation model for rgbt multispectral vision. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7872–7872. IEEE, 2025.

A Data Construction Details

This section describes how the instance masks, instructions, and splits of the benchmark are obtained, so that every target set of Section 3.2 can be traced to its pixels and its instructions.

A.1 Instance: SAM-guided Morphological Processing

The source datasets annotate categories, not instances. For MSRS, PST900, and SemanticRT alike, we split each category mask into instances with a marker-controlled watershed whose markers come from SAM: SAM proposes object masks over the image, and a proposal becomes a marker for a category when more than 50 of its pixels, and at least half of its area, fall inside that category's mask. The watershed then runs on the distance transform of the category mask, so that touching objects separate along their narrowest junction, and parts of the mask that no marker reaches become instances of their own. In dark frames, where the mean RGB brightness falls below 40, the thermal image replaces RGB as the input of SAM. Components smaller than 10 pixels are discarded.

Only categories of objects are split: persons, cars, tricycles, bicycles and motorcycles together with their riders, and the portable objects of PST900. Other categories, such as curbs, poles, and traffic lights, keep one instance per image. Instances are numbered $1000\,c + j$ for the j-th instance of category c, and every record stores the numbers of its targets, so that an instruction always resolves to the same pixels. Remaining merges and fragmentations are corrected by the human reviewers who inspect every record and its masks (Appendix A.2).

A.2 Instruction: Human-in-the-loop Multi-agent Annotation System

We organize instruction annotation into two tracks:

- For one-to-many, annotators write both instruction forms and the supporting chain of thought, while checking and correcting the instance masks. Selecting a subset requires a description that includes every intended instance and excludes the remaining same-category candidates. Writing the instructions and inspecting the masks together allows annotators to check this correspondence directly. These human-written examples also serve as language exemplars for the second track.
- For one-to-one, one-to-all, and one-to-none, we use a human-in-the-loop multi-agent workflow. Annotation agents based on `Claude Opus 4.7`, `ChatGPT 5.5`, and `Gemini3 Pro` each generate instruction pairs and supporting rationales for a portion of the data, using the human-written examples as few-shot guidance. Distributing generation across models reduces reliance on a single model's phrasing habits. Each draft is then checked by a different model, which independently identifies the target set from the images and instruction without seeing the proposed targets or supporting rationale. Its selection is compared with the annotation, and discrepancies trigger revision. Human reviewers verify every record and its associated masks, correcting errors before acceptance. This workflow combines human guidance, cross-model verification, and final human review to maintain agreement between each instruction and the instances it selects.

Annotation and verification instructions. Figure 7 gives the instructions of the annotation and verification agents in abridged form. The annotation agent sees the RGB image, the thermal image, and the RGB image with the target instances outlined. The verification agent sees the two images with every candidate instance numbered, together with the instruction alone, and returns the numbers of the instances the instruction designates. A record is accepted only when this selection equals the annotated set, empty sets included; otherwise the draft returns to its annotation agent together with the verifier's selection.

Annotation agent. *Input:* RGB image; thermal image; RGB image with the target instance(s) outlined; the selection scope; human-written exemplars. *Output (JSON):* `referring_cot`, `referring_instruction`, `reasoning_cot`, `reasoning_instruction`.

Rules. (1) Each instruction designates exactly the outlined instances: all of them and no other instance of their category. (2) The referring instruction uses visible attributes and spatial relations. (3) The reasoning instruction uses function, intent, or context and does not name the category. (4) Instructions never mention thermal evidence (e.g., *thermal, infrared, heat, warm, temperature, glow*); the chains of thought use both images. (5) For one-to-none, write a request of the assigned type that no instance satisfies.

Verification agent. *Input:* RGB image; thermal image; RGB image with all candidate instances numbered; one instruction. *Output:* the numbers of the designated instances, or `none`.

Rules. Select every instance the instruction designates and no other; answer `none` only if no instance qualifies.

Figure 7: Instructions to the annotation and verification agents (abridged).

One-to-none requests. A one-to-none request can be answered only by inspecting the scene. We construct two types. In the first, the requested category is absent from the image (612 of the 1,497 one-to-none records); in the second, the category is present but none of its instances has the requested attribute or relation (the other 885), so that rejection requires checking every candidate. Both types follow the rules of target-bearing instructions, and a request is accepted only if the verification agent answers `none`.

A.3 Benchmark Statistics

Table 5 gives the benchmark by split and selection scope. Every target set carries one referring and one reasoning instruction. The scopes share images, so the number of images is not the sum over scopes.

Table 5: **Benchmark statistics.** Top: target sets by split and selection scope; each target set has one referring and one reasoning instruction.

		Target sets					
Split	Images	1-to-1	1-to-many	1-to-all	1-to-none	Total	Instructions
Train	3,460	5,987	1,175	6,342	1,029	14,533	29,066
Val	612	1,071	208	1,109	190	2,578	5,156
Test	928	1,612	315	1,694	278	3,899	7,798
All	5,000	8,670	1,698	9,145	1,497	21,010	42,020

B Utility Estimation and Validation

Requests and the gate are supervised by the utilities of Equation 6. This section defines how they are estimated and verifies the two properties that the training losses use: their order and their sign.

Losses. The instance utility uses $\mathcal{L}_{\text{joint}} = 0.3\,\mathcal{L}_{\text{seg}} + \mathcal{L}_{\text{trace}}$, the training loss of a target-bearing example without its rejection term, so that thermal evidence is credited both for the mask and for the reasoning that selects the target. The window utility uses $\mathcal{L}_{\text{seg}}$ alone, since deep windows act only on the mask. $\mathcal{L}_{\text{seg}}$ and $\mathcal{L}_{\text{trace}}$ are defined in Appendix C.

Integrated gradients. Each utility integrates gradients along a straight path from zero to the adapter outputs that its intervention in Equation 6 removes. For the instance utility, δ collects the outputs of all shallow and deep adapters; for the window utility, δ collects only the outputs of the deep adapter, and the shallow outputs keep their actual values. With $\mathcal{L}(\alpha\delta)$ the loss when the outputs in δ are scaled by α, the attribution of position p is

$$a_p = -\Big\langle \delta_p,\ \frac{1}{M}\sum_{m=1}^{M} \nabla_{\delta_p}\mathcal{L}\Big(\tfrac{m-1/2}{M}\,\delta\Big)\Big\rangle, \tag{10}$$

where $\langle\cdot,\cdot\rangle$ sums over channels, evaluated with the midpoint rule. Positive attribution means that removing the evidence raises the loss, as in Equation 6, and the attributions sum approximately to $\mathcal{L}(0) - \mathcal{L}(\delta)$. Training labels use $M = 4$ and the validation below $M = 8$. The instance utility u_i sums the attributions of the shallow-and-deep path over all adapters and positions, weighting each position by the fraction of its cell covered by instance i; it is computed for every instance of the image, target or not. The window utility u_w^{post} sums the attributions of the deep-only path over window w.

Cost. Exact removal needs one forward pass per instance for u_i and one per window for u_w^{post}, that is, 257 passes per example over the full grid and $|\mathcal{R}| + 1$ inside a requested region of $|\mathcal{R}|$ windows. Integrated gradients need M forward–backward passes per example, whatever the number of instances and windows, with gradients taken only with respect to adapter outputs. On one B200 GPU, window utilities over the full grid cost 1.2 s per example with $M = 4$, against about 69 s for exact removal (0.27 s per pass). All utilities are computed once, on frozen reference models, before the stage they supervise; the gate then trains on the stored labels and never removes windows one by one.

Reference models and evidence scope. Request labels come from the frozen full-evidence model with thermal evidence admitted everywhere, which values each instance's evidence before any request is made. Gate labels come from the request-trained model with thermal evidence admitted only inside its own predicted region, the context in which the gate acts; window utilities outside the region are zero by construction.

Validation. We compare the estimates of the reference model with exact removal on 190 held-out images drawn by stratified sampling over annotation track, source dataset, and lighting (Table 6). Order and sign are checked separately, because the losses use them for different purposes. Order supervises the request region and the priority r_w: we compute, per image with at least three instances, the Spearman correlation between u_i and the loss change from removing each instance's evidence, and report the median over images and the correlation pooled over all instances. At the window level, we remove the eight windows of highest and of lowest estimated utility and check that the first

removal hurts more. Sign supervises the benefit score a_w: we compare the sign of each window's estimated utility with the sign of the loss change from removing that window alone.

Table 6: **Fidelity of the utility estimates** of the reference model ($M = 8$) against exact removal on 190 stratified held-out images.

Property	Test	Result
Instance order	Spearman correlation, median over 186 images	0.50
	Spearman correlation, pooled over 1,533 instances	0.51
Window order	top-8 removal hurts more than bottom-8 removal	94.5% of 200 records
Window sign	sign agreement with single-window removal	82.4%

The estimates certify order and sign rather than magnitude, and every loss that uses them relies on order (the request region and ranking) or sign (the benefit score) only.

C Architecture and Training Details

This section lists what is needed to rebuild the reported model: its modules (Table 7), its training (Table 8 and the losses below), and the construction of reasoning trajectories and inference.

Table 7: **Module specification.** Widths refer to SAM2.1 Hiera-B+ (stages of width 112/224/448/896) and Qwen2.5-VL-7B (hidden size 3584). All new output projections into the encoder stream are zero-initialized.

Module	Specification
Encoder split	Blocks 0–12 form the early stage and 13–23 the late stage; trunk, neck, and decoder are frozen.
Thermal stem	Standardized intensity, Sobel magnitude, and 15×15 local contrast, mixed by a 1×1 convolution (identity initialization), then the shared frozen patch embedding and blocks 0–2.
Shallow adapters	After blocks 1, 4, and 12; RGB and thermal 1×1 projections to 64 channels, concatenation fusion, GELU, 1×1 up-projection; output multiplied by G_{look}, upsampled and feathered by a 7×7 Gaussian (σ=1.5).
Token head	Block-12 features pooled to 16×16, MLP $448\to1792\to3584$, 2-D sine–cosine position code added before gathering; at most 256 evidence tokens.
Region head	Query $W_q h_{\mathrm{LOOK}} \in \mathbb{R}^{128}$; keys from RGB-only block-12 features pooled to 16×16 (1×1 convolution to 128, position code, one Transformer layer with 4 heads and FFN 512); $\tau_{\mathrm{look}} = 0.1$.
Gate	Pooled F_{mid} and $\Delta_{\mathrm{shallow}}$ (1×1 convolution to 128 each), fused to 128 with position code, two Transformer layers (4 heads, FFN 512); r_w and a_w are scaled dot products with separate projections of h_{SEG}.
Deep adapter	After block 20; one 4×4-token window of the 64×64 map per grid cell; projections to 64 channels, cross-attention between RGB and thermal tokens within each window (4 heads), FiLM from h_{SEG}; selected windows are physically gathered and scattered back.
Prompt head	Following CoPRS: h_{SEG} attends to the 64×64 neck features (4 heads of 64 channels), convolutional fusion, dense mask prompt for the frozen SAM2 decoder.
Rejection head	Linear layer on h_{SEG}; rejection when its probability exceeds 0.5.

Losses. The full-evidence stage minimizes $\mathcal{L}_{\mathrm{trace}} + 0.3\,\mathcal{L}_{\mathrm{seg}} + \mathcal{L}_{\mathrm{reject}}$; the request stage uses Equation 8 with $\lambda_{\mathrm{region}} = 1$, $\lambda_{\mathrm{seg}} = 0.3$, and $\lambda_{\mathrm{reject}} = 1$; the gate stage uses Equation 9 with $\lambda_r = \lambda_a = \lambda_n = 1$ and $\lambda_m = 0.3$. The trace loss is the token-level negative log-likelihood of the response. The segmentation loss acts on the prompt heatmap and the decoded mask, $\mathcal{L}_{\mathrm{seg}} = \mathrm{BCE}_{\mathrm{heat}} + 2\,\mathrm{Dice}_{\mathrm{mask}} + 5\,\mathrm{Focal}_{\mathrm{mask}} + 2\,\mathrm{Dice}_{\mathrm{heat}}$. The rejection loss $\mathcal{L}_{\mathrm{reject}}$ is a binary cross-entropy on the rejection head over all examples, with the examples without a target as positives; these examples carry the trace and rejection losses but no segmentation loss. The region loss is the sum of binary cross-entropy and soft Dice on requesting examples. $\mathcal{L}_{\mathrm{rank}}$ averages $\mathrm{softplus}(r_j - r_i)$ over all pairs of windows (i, j) in the requested region with $u_i^{\mathrm{post}} > u_j^{\mathrm{post}}$. $\mathcal{L}_{\mathrm{sign}}$ is a binary cross-entropy on a_w that takes windows with utility above the median positive utility of

Table 8: **Training configuration.** All stages use AdamW (weight decay 0.01), linear warm-up followed by cosine decay, gradient clipping at 1.0, bf16 computation with fp32 master weights, and one NVIDIA B200 GPU. LoRA: rank 16, $\alpha = 32$, dropout 0.05, on the query, key, value, and output projections of every language layer.

Stage	Trained parameters	Steps	Batch	LR	Warm-up
Full evidence	thermal stem, adapters, token head, prompt and rejection heads, LoRA, new token embeddings	16k	2	10^{-4}	400
Request	region head, LoRA, new token embeddings	4k	2	10^{-4}	100
Gate	gate	4k	4	10^{-3}	100

the training set as positives and windows with negative utility as negatives, and ignores the rest. The necessity term compares the segmentation loss with the selected windows and with no deep window processed. The selection thresholds are $\tau_u = \tau_a = 0$, $\kappa = 0.15$, and $\delta = 0.02$. During gate training, the selection keeps at most 64 eligible windows. The forward pass uses the hard selection; the backward pass replaces it by the derivative of a per-window sigmoid of the score (straight-through).

Trajectory construction. A requesting trajectory places `[LOOK]` after the first sentence of the rationale, which states from RGB what the instruction asks for; the evidence tokens follow `[LOOK]`, and the rationale continues after them. The instructions never mention thermal evidence (Appendix A.2). Examples whose gain d_S does not exceed δ, whose instances carry no positive utility, whose target covers no grid cell, or that have no target keep the full rationale without a request; these zero-request trajectories make up 10.4% of the training set. Under teacher forcing, every response token carries the trace loss except the injected evidence tokens and their delimiters.

Inference. Decoding is greedy, with at most 160 new tokens and one request. If the model emits no `[LOOK]`, or its region is empty, the thermal branch is not executed and the mask is decoded from RGB alone. Otherwise the gate keeps the $K = 32$ highest-priority eligible windows of the region, or all of them if fewer are eligible. The prediction is empty when the model emits no `[SEG]` or the rejection probability exceeds 0.5; otherwise the mask logits are upsampled bilinearly to the original resolution and thresholded at zero.

D Baselines and Evaluation Protocol

This section states how each compared method is run and how every metric is computed, so that Tables 2 and 4 can be reproduced.

Table 9: **Compared methods.** Each is fine-tuned on the MINT training split from the listed initialization.

Method	Initialization	Input	Multiple and absent targets
LISA-7B	LISA-7B-v1	RGB	one mask per instruction; no rejection
VisionReasoner-7B	VisionReasoner-7B, SAM2-L	RGB (+T)	one box and point per instance, masks merged; an empty list is a rejection
RIS-FUSION	RIS-FUSION	RGB-T	one mask per instruction; no rejection
Qwen2.5-VL → Sigma	Qwen2.5-VL-7B-Instruct; Sigma (MFNet, PST900)	RGB-T	all pixels of the named category; `none` is a rejection

Training and inputs. Every compared method starts from its pretrained weights and is fine-tuned on the MINT training split with its original objective and schedule, supervised by the union mask of each target set. LISA receives the instruction followed by "Please output segmentation mask." VisionReasoner is trained with its original reward on the boxes of the target instances, with an empty list for instructions without a target, and uses 840×840 inputs. For RGB+T, the thermal image, replicated to three channels and resized identically, is the second image, and the prompt is prefixed with "Image 1 is an RGB photo and Image 2 is a thermal infrared image of the same scene, pixel-aligned with Image 1. Give all coordinates in Image 1."; the segmenter runs on RGB. RIS-FUSION is trained at 480×480 with its BERT text encoder on both instruction types. For Qwen2.5-VL → Sigma, Sigma is fine-tuned on the category masks of the training images, and Qwen2.5-VL-7B sees

the RGB image and the instruction and answers with one category name from the dataset's list, or `none`. All decoding is greedy, and baseline predictions are resized to the original resolution by nearest-neighbour interpolation.

Metrics. For query q with target mask M_q (Equation 2) and prediction $\widehat{M}_q$, both at the original resolution, let Q^+ and $Q^\varnothing$ be the queries with and without targets. Then

$$\mathrm{mIoU} = \frac{1}{|Q^+|} \sum_{q \in Q^+} \frac{|\widehat{M}_q \cap M_q|}{|\widehat{M}_q \cup M_q|}, \qquad \mathrm{gIoU} = \frac{\sum_{q \in Q^+} |\widehat{M}_q \cap M_q|}{\sum_{q \in Q^+} |\widehat{M}_q \cup M_q|}, \tag{11}$$

$$\text{N-acc} = \frac{1}{|Q^\varnothing|} \sum_{q \in Q^\varnothing} \mathbb{1}\big[\widehat{M}_q = \varnothing\big]. \tag{12}$$

A target set with several instances is scored through the union of their masks, so the metrics reward selecting the right instances; an empty prediction on a query with targets scores zero. A prediction is empty when a method rejects or returns no mask. N-acc is undefined for methods that cannot reject, and the Night column reports mIoU on the night images of MSRS and SemanticRT.

E Additional Results

This section gives the full measurements behind the analysis of Section 5.3.

Region coverage. Table 10 compares requested regions with random regions of the same size across coverages, for the query-based region head and for a head that regresses the region from h_{LOOK} alone. The learned region leads at every coverage, and most when the region is small, where its placement matters most.

Table 10: **Learned vs. random regions of equal size (mIoU).** Coverage: mean share of the grid per requesting query. Request decisions are teacher-forced, except in the free-running rows.

Region head	Coverage (%)	Learned	Random	Δ
Regressed from h_{LOOK}	11.2	56.2	46.3	+9.9
	12.9	57.8	47.5	+10.3
	14.1	58.5	48.1	+10.4
	16.2	59.7	49.1	+10.6
	18.4	60.6	50.0	+10.6
	21.6	61.6	51.1	+10.5
free-running	11.5	56.3	46.6	+9.7
Query head (ours)	27.2	62.5	54.9	+7.6
	32.6	63.4	56.3	+7.1
	36.0	63.8	57.2	+6.6
	41.3	64.2	58.2	+6.0
	52.6	64.6	60.2	+4.4
free-running	41.6	64.5	58.7	+5.8

Deep-window budget. Table 11 varies the number K of deep windows inside the same predicted regions. The gate's lead over random windows grows as the budget shrinks, and processing the gate's selection is better than processing the whole region.

Request behaviour. Table 12 relates the requests to the benefit of thermal evidence. Queries that lose no accuracy without thermal evidence occur under every lighting, and the model skips its request mostly by day, at a small cost on the skipped queries.

Paired comparison. We estimate the differences between our model and its variants in Table 4 with a paired bootstrap over images, which resamples all instructions of an image together (10,000 resamples, 95% intervals). Our model exceeds deep processing of the whole region by 1.1 mIoU ([0.6, 1.7]), random windows by 2.4 ([1.8, 3.1]), and the instruction-blind gate by 1.5 ([0.9, 2.0]).

Table 11: **Deep-window budget (mIoU).** Share: K relative to the mean region of 106 windows.

K	Share (%)	Gate	Random	Δ
0	0	55.9		–
16	15	64.0	60.4	+3.6
32	30	65.6	63.2	+2.4
64	60	65.3	64.6	+0.7
all	100	64.5		–

Table 12: **Request behaviour.** No gain: full-evidence IoU exceeds RGB-only IoU by at most δ. Cost: mIoU change on skipped queries relative to forcing a request. Unlabeled: mostly PST900.

Lighting	No gain (%)	Skipped (%)	Cost
Day	14.5	17.6	−0.2
Night	13.3	3.9	−0.6
Unlabeled	9.5	4.8	−0.5
All	13.0	9.3	−0.3

F LIMITATIONS

Instance masks are derived from semantic annotations; although every record is inspected and corrected, touching or occluded objects can still be split or merged imperfectly. The request and the gate act on a 16×16 grid, while 48.6% of the MSRS instances are smaller than one cell, so the evidence of small targets is admitted and processed together with their surroundings. The model makes one request per instruction and does not consult evidence outside its first region. Finally, the method assumes registered RGB–thermal pairs, and the benchmark covers road, urban, and subterranean scenes with English instructions.