Event-Centric Perception in Weak-Signal Physical Streams with Multimodal LLMs

Chi Xu[1,2] **Mengdi Jin**[1,2] **Jiaxing Li**[1,2] **William I. Atlas**[3] **Thor Veen**[4]
Edith C.H. Ngai[5] **Jiangchuan Liu**[1*]
[1]Simon Fraser University, Burnaby, Canada
[2]AXSOTA Intelligence Inc., Burnaby, Canada
[3]Wild Salmon Center, Portland, United States
[4]Lumax Ecological Analytics, Squamish, Canada
[5]The University of Hong Kong, Hong Kong SAR, China
{chix, mengdij, jla641}@sfu.ca, watlas@wildsalmoncenter.org,
thor@lumax.ai, chngai@eee.hku.hk, jcliu@sfu.ca

Abstract

Robotic and sensing systems operating in challenging environments often receive weak, noisy, and incomplete data streams, such as underwater imaging sonar or thermal video under low visibility. In these scenarios, useful evidence for downstream tasks is sparse over time. The objective extends beyond frame-level analysis, requiring perception to reconstruct physically meaningful events. We study this problem through ***event-centric perception***, which maps physical sensor streams into structured events with temporal support, spatial context, direction, count, and magnitude. Frontier multimodal LLMs (MLLMs), such as Qwen3.5-Plus and Gemini, show potential for interpreting weak physical signals, but direct prompting often leaves temporally distributed evidence underused. This problem is even worse with compact MLLMs. We address it through a shared event representation that connects agentic reasoning and lightweight event-aligned adaptation. For frontier models, the agentic workflow guides local inspection, refines event hypotheses, and consolidates evidence across windows. For compact models, event-aligned adaptation enables weak observations to contribute to event predictions. In our experiments on a sonar benchmark, the adapted approach improves positive event recall from 0.041 to 0.914 and reduces normalized event error from 0.973 to 0.335. Thermal experiments further show that the approach extends to wildlife event reconstruction and temporally grounded occupancy reasoning. These results support event-centric perception as a unified framework for physical-stream intelligence.

1 Introduction

Robotic and sensing systems operating underwater or in low-visibility environments often rely on physical sensing streams. For example, imaging sonar [1, 2] and thermal infrared video [3, 4] are increasingly used in ecological monitoring, industrial inspection, and infrastructure-scale surveillance. The broader markets for underwater sonar and thermal imaging alone are projected to reach US$7.1 billion and US$16.6 billion, respectively, by 2033 [5, 6]. In monitoring and inspection deployments, these systems often operate continuously, supporting decisions that depend on how physical activity unfolds over time.

*Corresponding author.

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

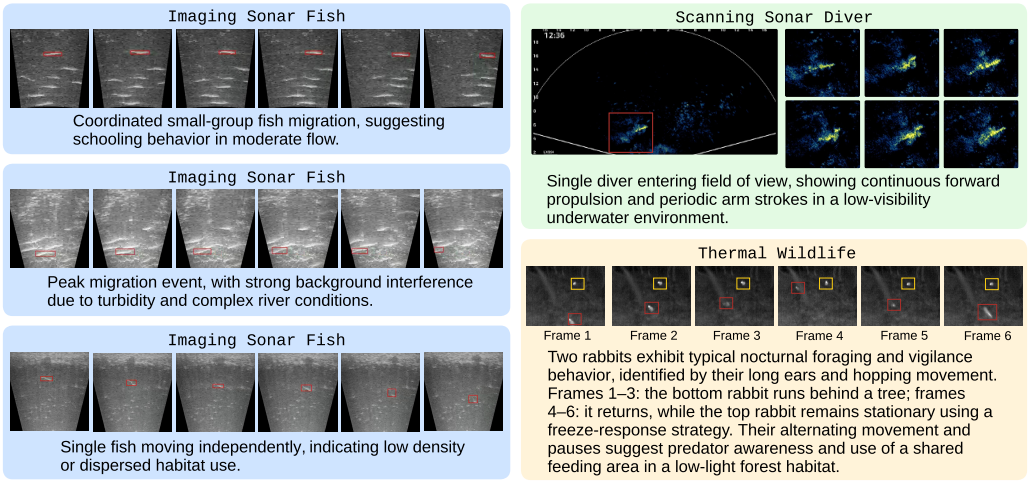

Figure 1: Examples of weak-signal physical sensing streams from different domains: imaging sonar capturing fish migration, scanning sonar spotting a human diver, and thermal infrared camera monitoring wildlife.

Their objectives require ***event-level understanding***: how many objects passed through a region, in which direction they moved, when activity occurred, and whether activity changed over time? Such quantities depend on observations accumulated across multiple frames and intervals.

As shown in Fig. 1, physical sensing streams behave very differently from conventional RGB video. Observations often contain weak, noisy signals shaped by sensing physics and environmental conditions [1, 7]. In imaging sonar, reverberation and scattering distort both structure and motion [8, 9]. In thermal sensing, contrast varies with ambient conditions and weather [10]. Targets may appear briefly, remain partially visible, or only be detectable through weak motion cues. Background and clutter dominate the stream, while meaningful events are sparse and unevenly distributed [11]. Evidence for a single event may therefore be fragmented across time. Reliable perception requires these partial observations to contribute to a coherent event judgment.

We study this problem through ***event-centric perception***, which maps physical streams into structured events with temporal support, spatial context, direction, count, and magnitude. This formulation connects model predictions to the quantities required in sensing tasks. It also makes weak evidence use observable: the model needs to determine whether a trace supports an event, identify its relevant attributes, and connect observations that may refer to the same event.

Frontier multimodal LLMs (MLLMs) [12], such as Qwen3.5-Plus [13] and Gemini [14], show potential for interpreting weak physical signals. However, direct prompting often leaves temporally distributed evidence underused. This problem is even worse in compact MLLMs, which can remain highly conservative and miss weak events despite receiving the relevant observations. We also observe that explicit output instructions alone provide limited recovery. These behaviors motivate methods that improve both how evidence is examined and how it contributes to event predictions.

We address this problem through a shared event representation that connects agentic reasoning and lightweight event-aligned adaptation. More specifically, local inspection constructs an event hypothesis, further observations refine its attributes, and stream-level consolidation updates and merges event records across overlapping windows. We use an event record to store the current event hypothesis, and the same representation also defines supervision targets for adapting MLLMs. Leveraging the event record, the model then distinguishes candidate coverage from weak evidence use: a window containing the event needs to be selected for local inspection, and the model then needs to use its observations in the event prediction.

For frontier models, the agentic workflow begins with a sampled stream overview, identifies likely event regions, and guides local inspection. Ambiguous regions can be revisited before the resulting event records are consolidated. For locally deployable MLLMs with fewer than ten billion parameters (e.g., Qwen3.5-9B), to facilitate the compact workflow, we introduce a fixed proposal scorer that selects candidate windows, which are short stream segments for local inspection. Event-aligned

adaptation trains the compact MLLM to use weak but consistent observations within these windows. These paths share the event representation while addressing limitations in event reconstruction.

We evaluate our solution on representative sonar and thermal sensing streams. On the sonar benchmark, event-aligned adaptation improves positive event recall from 0.041 to 0.914 and reduces normalized event error from 0.973 to 0.335. Comparisons under matched event-window supervision and transfer to a separate sonar site further assess the compact model's role in local event prediction. Thermal experiments extend event-centric inference to wildlife event reconstruction and temporally grounded occupancy reasoning. On thermal event reconstruction, event-centric inference improves animal event recall from 0.250 to 0.917 and mean temporal Intersection over Union (IoU) from 0.227 to 0.684. Component-level analyses are conducted to further examine context sensitivity, supervision, and candidate coverage.

Our contributions are summarized as follows:

- We formulate the event-centric perception for weak-signal physical streams through a shared event representation that connects local inspection, model supervision, and stream-level consolidation.
- We characterize weak evidence use in frontier and compact MLLMs, distinguishing candidate coverage, local event prediction, and the consistency of consolidated event records.
- We develop agentic event-centric inference and lightweight event-aligned adaptation, and evaluate them through sonar and thermal experiments, supervised comparisons, cross-site transfer, and component-level analyses.

2 Event-Centric Perception in Physical Streams

2.1 Motivating Examples

Consider two monitoring applications in Fig. 1. In river monitoring, an imaging sonar produces a stream of underwater images. The goal is to count how many fish pass through the monitored region and determine whether they move upstream or downstream. A fish may appear in many frames, disappear briefly, and reappear as it moves. These observations need to be linked to the same passage to avoid counting it repeatedly. An event therefore represents a fish passage, with its direction, time interval, and number of fish involved.

In thermal wildlife monitoring, the input is a stream of thermal infrared images, and the goal is to determine when animals appear and which specific regions they occupy. For example, an animal entering a region, remaining there, and leaving forms an event with a location and a time interval. Reconstructing this event requires connecting observations over time and distinguishing animal activity from false triggers. These two cases motivate the event representation and inference workflows introduced below.

2.2 Formulation

Let $X_{1:T} = (x_1, \ldots, x_T)$ be a physical sensing stream, e.g., sonar or thermal, where x_t is the observation at time t. We introduce the following terms: a window is a short segment of this stream; a source clip is the longer video segment from which windows are drawn. A candidate window is a window selected for local inspection by an MLLM. The target is a set of structured events

$$\mathcal{E} = \{e_i\}_{i=1}^{N}, \qquad e_i = (\tau_i, \ell_i, z_i, d_i, c_i, s_i).$$

Here $\tau_i = [t_i^s, t_i^e]$ denotes temporal support, ℓ_i event type, z_i spatial context expressed through spatial support or zone membership, d_i direction, c_i magnitude, and s_i confidence. Magnitude c_i describes the task-specific quantity of interest. For example, in sonar-based fish monitoring, c_i represents passage count, the number of passages associated with an event. In thermal wildlife monitoring, the emphasis is on animal presence, temporal support, and occupancy, meaning whether and when an animal is present in a designated zone.

We write event-centric perception as $\hat{\mathcal{E}} = f_\theta(X_{1:T})$. In this formulation, each predicted event is stored in an event record that links a task-specific estimate to its temporal and spatial support. A physical event may appear across several observations, each revealing part of its activity. Temporal

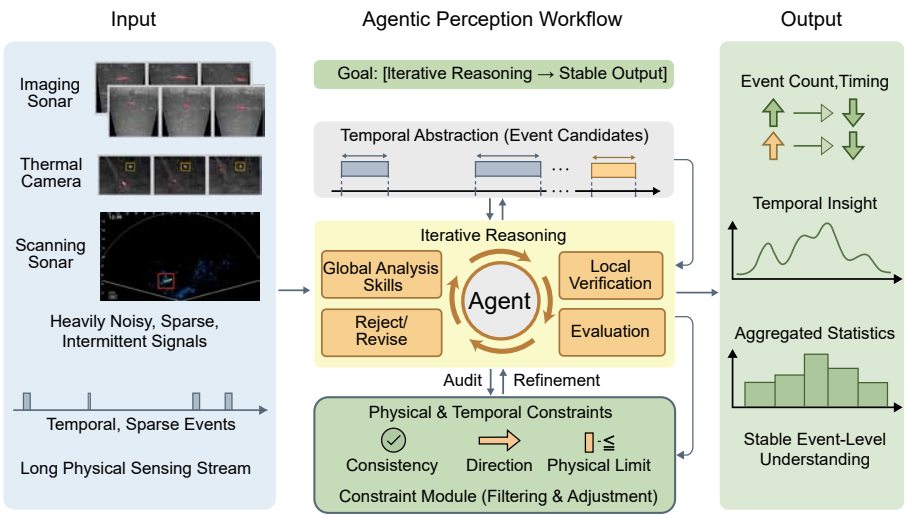

Figure 2: The frontier agentic workflow uses a global overview to localize candidate regions, inspect local evidence, and revisit ambiguous event hypotheses before stream-level consolidation. Structured event records connect these inspection stages; the same representation supplies supervision targets for event-aligned adaptation.

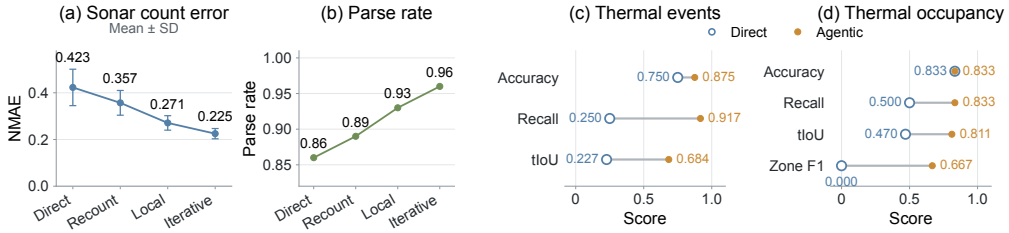

Figure 3: Frontier inference with Qwen3.5-Plus: (a,b) sonar count error and parse rate across inference stages; (c,d) direct and agentic inference on thermal event reconstruction and occupancy reasoning. Error bars in (a) show standard deviation (SD).

and spatial support provide a basis for connecting these observations. Event reconstruction therefore requires deciding which observations support a common event and which describe separate events, while preserving their attributes and magnitude.

To construct these event records from a physical sensing stream, the model examines selected candidate windows through local inspection. Each inspection forms an event hypothesis: a current interpretation of the observed activity, expressed through the fields of an event record. Further observations can refine this hypothesis and update its record. Because overlapping windows may capture the same physical event, stream-level consolidation then connects and merges their records to produce the reconstructed event set.

This process separates three questions about event reconstruction. First, candidate coverage asks whether an event reaches local inspection through the selected windows. Then, weak evidence use asks whether the model uses the observations within those windows to form an event prediction. Stream-level consolidation asks whether predictions across windows preserve event identity and count. Together, these questions connect the available event evidence to local prediction and the consistency of the reconstructed event set.

2.3 Frontier MLLMs under Event-Centric Inference

We use frontier MLLMs to refer to large general-purpose models with broad multimodal capabilities. We evaluate Qwen3.5-Plus [13], Gemini 2.5 Pro [14], and GPT-5.2 [15] without domain-specific adaptation. With direct prompting, the models produce a structured event prediction in one pass. Agentic inference uses the models to guide candidate selection and local inspection, with repeated

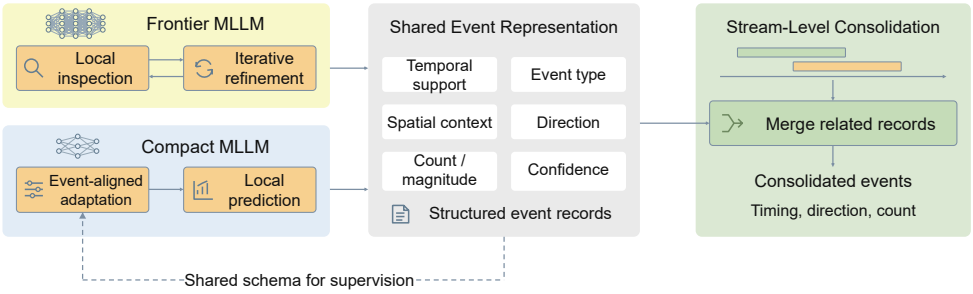

Figure 4: A shared event representation connects frontier inference, compact-model adaptation, and stream-level consolidation.

inspection when needed, as illustrated in Fig. 2. Iterative inspection denotes this repeated refinement within the agentic workflow. The recount condition asks the model to reconsider its initial count.

We first evaluate these inference settings on sonar-based fish monitoring, where the task is to estimate the number of fish passages. We measure count error using normalized mean absolute error (NMAE), with lower values indicating more accurate counts. For Qwen3.5-Plus, NMAE decreases from 0.423 under direct prompting to 0.357 with recounting, 0.271 with local inspection, and 0.225 with iterative inspection. Agentic Gemini and GPT reach 0.266 and 0.242, respectively. We also report parse rate, the fraction of outputs that satisfy the required event-record format. Figure 3(a,b) shows that count error decreases across the Qwen3.5-Plus inference stages while parse rate rises from 0.86 to 0.96. Error bars in panel (a) show standard deviation (SD). Full stage-wise values and additional model comparisons are included in Appendix A. These comparisons measure the cumulative gains across successive inference stages.

This setting probes whether general-purpose multimodal priors can support event reconstruction without additional domain training. Figure 3(a) shows that recounting improves on direct prompting, while local inspection and iteration yield further reductions in count error. The stages progressively organize how the model inspects evidence and updates event records. Their gains motivate an inference workflow in which an initial interpretation can be revised through further inspection, with a structured event hypothesis carrying the result forward. To examine whether these gains extend beyond sonar counting, we apply the same agentic workflow to thermal wildlife streams (Fig. 3(c,d)). Here, event reconstruction requires connecting animal observations over time. Agentic inference improves event recovery and temporal localization. In the separate occupancy task, it also improves event timing and zone-entry detection, despite unchanged binary accuracy.

2.4 Event Inference Behavior of Compact MLLMs

We next examine how Qwen3.5-9B, a representative compact MLLM suitable for local deployment, interprets weak signals before event-aligned adaptation. We refer to this unadapted model as frozen. We compare two direct-prompting settings: generic prompting asks the model to identify and count fish passages, while strict prompting also specifies the event fields and output format. Here, we use positive event recall to measure the fraction of windows containing annotated events that the model correctly identifies as positive. On the sonar benchmark, the two settings achieve positive event recall of 0.012 and 0.041, respectively. Both settings miss most positive windows, indicating that clearer output instructions alone are insufficient for recovering weak events.

These missed events motivate event-aligned adaptation, which trains the compact model to predict structured event records from observations within each window. The training targets use the same event fields that guide frontier models during agentic inference. The following section describes this shared representation and its role in both approaches.

3 Methods

As illustrated in Fig. 4, our method is to use a shared event representation for inference, supervision, and stream-level consolidation. For frontier models, agentic reasoning is implemented through model-guided candidate selection and local inspection. For compact models, a fixed proposal scorer selects candidate windows for inspection by an adapted compact MLLM.

In both paths, event records connect local observations to the final stream output. Local inspection uses the observed evidence to predict event fields. Adaptation trains the compact model to predict the same fields, while stream-level consolidation updates and merges records that describe the same event. This shared representation also helps identify where errors occur—an event may be missed during candidate selection, missed or incorrectly described during local prediction, or counted more than once during consolidation.

3.1 Event-Centric Inference

A multimodal LLM encodes observations as $V = v_\eta(X)$ and generates a structured response conditioned on a task prompt P:

$$p_\theta(Y \mid X, P) = \prod_t p_\theta(y_t \mid y_{<t}, V, P).$$

Here $Y = (y_1, \ldots, y_{|Y|})$ is the output token sequence, v_η is the visual encoder, and θ denotes the model parameters used for generation. Local inspection operates on candidate windows $w_k = X_{t_k^s:t_k^e}$ with task prompts P_k. After output parsing, it produces an event record $\hat{e}_k$.

Frontier agentic workflow. The MLLM first examines a sampled overview to locate likely event regions within the stream. This overview provides coarse temporal context, and lightweight motion cues help identify regions that warrant closer inspection. The model uses these cues to estimate candidate start and end times, defining windows for local inspection.

Within each candidate window, the model examines the observations in greater detail and records its current event hypothesis. If the passage count, direction, or temporal interval remains unclear, it can revisit the relevant region and update the record using the extra observations. Repeated inspection therefore refines a specific event hypothesis, with the record carrying the current prediction across model calls. Stream-level consolidation then connects and merges records across overlapping windows to form the final event output.

Compact candidate selection and local inspection. For compact models, we train a fixed R3D-18 [16] proposal scorer that selects candidate windows for local inspection. The scorer is to predict event presence and scans the stream to assign each window a score $q_k = h_\phi(w_k)$. To reduce repeated inspection of the same activity, temporal non-maximum suppression (NMS) is used to remove strongly overlapping candidates. The highest-scoring windows are then passed to the adapted compact MLLM, with top-k selection retaining k windows per recording.

This selection determines which observations reach the compact model and limits the cost of local inspection. Increasing k allows more windows to be inspected, improving the opportunity to cover weak events while requiring more model calls. Event recovery therefore depends on both candidate coverage and the model's use of the selected observations, motivating their separate evaluation.

3.2 Event Records and Stream-Level Consolidation

During local inspection, the MLLM generates a response $\hat{y}_k = g_\theta(w_k, P_k)$. We then check the required fields and output format and use a parser to convert a valid response into an event record $\hat{e}_k = \psi(\hat{y}_k)$. This shared format allows us to compare predictions across windows.

These records need to be considered together because windows may overlap or capture only part of an event. For example, two windows may describe the same passage, leading to duplicate counts. Partial observations may also split one event into several records, while merging distinct passages can lead to undercounting. We therefore consolidate the records into a stream-level event set:

$$\hat{\mathcal{E}} = A(\hat{e}_1, \ldots, \hat{e}_K).$$

Specifically, we use temporal support and direction to connect records describing the same activity, update their attributes, and merge duplicates. Consolidation also uses the magnitude field to keep passage count information during these updates. Together, these steps connect observations across windows and support consistent event counts.

3.3 Lightweight Event-Aligned Adaptation

We use low-rank adaptation (LoRA) [17] to learn updates to the compact model: $W' = W + BA$, where W is a frozen base weight matrix and BA is a learned update of rank r. Event-aligned supervision supplies targets in the shared event representation, and event-aligned adaptation learns to predict these targets. We train through supervised fine-tuning (SFT) with the structured generation objective

$$\mathcal{L}_{\mathrm{SFT}} = -\sum_{(X,P,Y)\in\mathcal{D}} \sum_{t} \log p_{\theta+\Delta\theta}(y_t \mid y_{<t}, X, P).$$

Here $\mathcal{D}$ contains input windows X, prompts P, and target token sequences Y; $\Delta\theta$ denotes the learned adapter parameters. We use two target formats drawn from the shared event representation: counts-only targets specify leftward and rightward passage counts; full-event targets also include event timing and crossing information. These formats allow us to train the model to estimate passage magnitude and predict more detailed event attributes. Through token-level supervision, we train the compact model to connect observations with the corresponding event fields. Positive windows show how weak but consistent traces support an event prediction, while negative windows teach the model to reject background activity and noise. Through this supervision, the model learns to use weak signals when predicting events while avoiding unsupported detections.

4 Evaluations

4.1 Experimental Setup and Metrics

We evaluate sonar perception on the Caltech Fish Counting (CFC) dataset [7, 18] and thermal perception on wildlife data [4]. The aggregate sonar benchmark contains 620 windows, with a separate evaluation on 130 source clips. Controlled experiments use the Kenai sonar site for training and model selection, Kenai-59 for testing on disjoint source recordings, and Elwha for cross-site transfer. Our compact experiments compare Qwen3.5-9B before adaptation (frozen) and with learned low-rank updates (adapted). Local-prediction comparisons provide labeled windows directly to each model. Candidate-coverage and consolidation tests separately examine window selection and reconstruction across observations.

For sonar, we report positive event recall, specificity (correct rejection of negative windows), event F1, and normalized mean absolute error (NMAE) on passage counts. Thermal experiments evaluate pretrained frontier models on separate event-reconstruction and occupancy subsets using binary accuracy, event recall, mean temporal IoU, and zone-entry F1. Detailed data partitions, evaluation protocols, and metric definitions appear in Appendix B.

4.2 Compact Event Reconstruction under Weak Signals

In Table 1, we show compact-model results on the aggregate sonar benchmark. We find that strict prompting identifies few positive windows, while event-aligned adaptation substantially improves recall and continues to reject most negative windows. The full results and holdout diagnostics are presented in Appendix C.

The adapted variants show the tradeoff between recovering events and avoiding false detections. The high-recall configuration (Row 3, Table 1) achieves recall of 0.931 and specificity of 0.842, compared with 0.914 and 0.903 for the balanced configuration (Row 4). Both identify far more positive windows than strict prompting, while the balanced configuration rejects more negative windows. We use NMAE to further measure the accuracy of the predicted passage counts.

4.3 Matched Supervision and Cross-Site Transfer

We compare the adapted compact MLLM with two non-LLM video models, R3D-18 [16] and Swin3D-T [19], using the same Kenai training and selection windows and the Kenai-59 test set. All three models receive the same event-window supervision. The non-LLM models use frozen Kinetics-400 encoders with trained prediction heads. Each model receives the test windows directly. The R3D-18 predictor is separate from the proposal scorer; training details appear in Appendix D.

Table 1: Compact sonar results on the 620-window aggregate evaluation.

Model	Recall	Specificity	NMAE
Frozen (generic prompt)	0.012	0.995	1.021
Frozen (strict prompt)	0.041	0.991	0.973
Adapted (high recall)	0.931	0.842	0.331
Adapted (balanced)	0.914	0.903	0.335

Table 2: Local event prediction under matched supervision and cross-site specialist comparisons. These panels use separate evaluation protocols.

Model	Event F1 ↑	NMAE ↓
Kenai-59: matched event-window supervision		
Compact MLLM	0.943	0.231
R3D-18	0.912	0.308
Swin3D-T	0.887	0.325
Elwha: cross-site evaluation and specialist references		
Frozen compact MLLM	0.403	0.969
Kenai-adapted compact MLLM	0.874	0.417
YOLO26s detection and tracking	0.697	0.567
CFC Baseline++	0.873	0.299

The compact MLLM has the strongest point estimates (Table 2). Relative to R3D-18, its F1 difference is $+0.031$ (95% confidence interval (CI) $[-0.018, 0.109]$) and its NMAE difference is -0.077 ($[-0.231, 0.038]$). Both intervals include zero, supporting comparable performance. The corresponding differences relative to Swin3D-T favor the compact model (Appendix D).

In cross-site transfer experiments, adaptation using only Kenai data improves the compact model's recall on Elwha from 0.257 to 0.931 and reduces NMAE from 0.969 to 0.417. We also compare against two specialist fish-counting pipelines. The first uses a YOLO26s [20] detector and a tracker, trained with dense frame-level annotations from the same Kenai training recordings. No Elwha data is used for training or model selection. The second uses released tracks from CFC Baseline++ [7] under its published protocol. CFC Baseline++ and the compact MLLM achieve similar event F1, but CFC Baseline++ has lower count error. Because these specialist pipelines use different supervision, we report their results separately from the matched comparison.

4.4 Component-Level Analysis

Context and weak evidence use. We test whether adaptation gains remain when extra background content is added to each labeled window. We compare three inputs: the window alone, with static background content, or with changing background content containing no annotated target events. On Kenai-59, adapted F1 is 0.943, 0.943, and 0.930, respectively (Fig. 5(a)). With dynamic background content, adaptation improves F1 by 0.930 (95% CI $[0.837, 1.000]$). Both frozen and adapted models achieve parse rate 1.000 on Kenai-59 and Elwha, meaning their outputs follow the required format. The gains remain with added background and cannot be explained by formatting alone, supporting improved weak evidence use. These context definitions are shown in Appendix E.

Supervision and event structure. In our experiments, increasing the number of training windows from 32 to 157 keeps recall on Kenai-59 at 0.971 while reducing NMAE from 0.744 to 0.231 (Fig. 5(b)). More supervision therefore improves passage counts even when event recall remains unchanged. We also evaluate passage timing using full-event outputs, which achieve a crossing F1 of 0.853. Training and timing-evaluation settings are also described in Appendix E.

Consolidation. To test whether consolidation prevents repeated counting, our extended Overlap-36 experiment introduces repeated observations into a set of 36 local event records. As repetition increases from 25% to 100%, directly summing the counts raises NMAE from 0.5 to 2.0. Stream-level consolidation removes duplicate records and keeps NMAE at zero (Fig. 5(d)). This test isolates the effect of repeated observations, but we also note that naturally occurring overlap in long streams presents additional challenges (Appendix E).

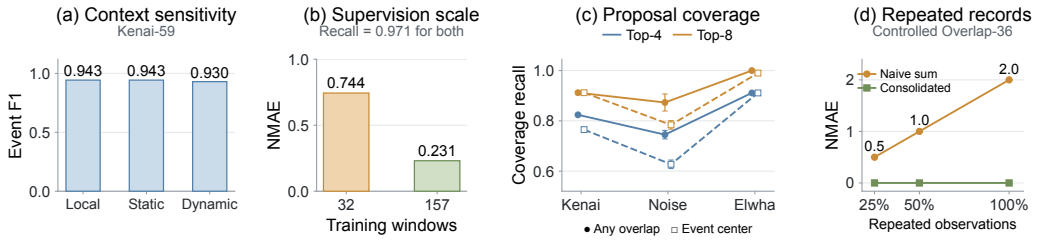

Figure 5: Component analyses: (a) context sensitivity, (b) supervision scale, (c) candidate coverage, and (d) duplicate suppression under controlled repetition. Noise error bars in (c) show SD over three draws. Panels use separate evaluation protocols.

Table 3: Thermal results with Qwen3.5-Plus without domain-specific adaptation. Event reconstruction and occupancy reasoning use separate task subsets.

Task	Method	Bin. Acc.	Event Recall	Mean tIoU	Zone F1
Event reconstruction	Direct prompting	0.750	0.250	0.227	–
Event reconstruction	Agentic inference	0.875	0.917	0.684	–
Occupancy reasoning	Direct prompting	0.833	0.500	0.470	0.000
Occupancy reasoning	Agentic inference	0.833	0.833	0.811	0.667

Candidate coverage. In Fig. 5(c), we compare candidate coverage when retaining four or eight windows per recording (top-4 and top-8). We use any-overlap recall to measure the fraction of events that overlap at least one selected window, while event-center recall requires a selected window to contain the event midpoint. On held-out Kenai, top-4 and top-8 achieve any-overlap recall of 0.824 and 0.912, respectively. Under severe noise, these values fall to 0.745 and 0.873 on average. Selecting more windows also improves event-center recall. On Elwha, top-8 achieves any-overlap recall of 1.000 and event-center recall of 0.990. These results show that keeping more candidates improves coverage, but local inspection cannot recover events outside all selected windows. Scorer settings, noise conditions, and full results are described in Appendix F.

4.5 Event Consolidation within Source Recordings

We evaluate event consolidation on 130 source clips to test how local predictions are brought together across windows. The adapted compact model reduces NMAE from 1.018 to 0.781 (Appendix C). Count errors remain large, especially when passages overlap or move in different directions. This evaluation examines reconstruction within source recordings; consistency across days or weeks of monitoring remains to be tested. The gap between local event recovery and source-clip count accuracy shows that recognizing events in individual windows is only part of the task. When records from several windows are brought together, the same event may be counted more than once or split into separate events, while different events may be incorrectly merged. The source-clip evaluation tests these problems together, while Overlap-36 in Section 4.4 focuses on repeated records. Both tests point to the need for better event matching and more accurate counts during consolidation.

4.6 Cross-Domain Thermal Evidence

We use thermal wildlife streams to test agentic inference with frontier models on a different sensing modality and different event types. Table 3 reports results on two separate subsets for event reconstruction and occupancy reasoning. For event reconstruction, agentic inference improves Qwen3.5-Plus recall from 0.250 to 0.917 and mean temporal IoU from 0.227 to 0.684, showing better event recovery and more accurate event timing.

Occupancy reasoning tests whether and when animals occupy or enter a designated zone. Binary accuracy remains at 0.833 under both direct prompting and agentic inference, but mean temporal IoU increases from 0.470 to 0.811 and zone-entry F1 from 0.000 to 0.667. These gains show why event-level metrics matter: a correct yes-or-no judgment alone does not show whether the model identifies when an animal is present or enters the zone. Together, these results extend event-centric perception beyond sonar counting to animal event timing and zone occupancy.

5 Discussion

The shared event representation proposed in this paper connects local inspection, supervision, and consolidation, while the two inference paths (frontier and compact) offer different deployment choices. Matched comparisons support compact MLLMs as local event predictors with structured outputs. The specialist counting advantage on Elwha also indicates that event F1 and count accuracy can diverge. These results show that richer outputs alone do not establish better counting.

In terms of computational efficiency, the compact workflow processes streams at $31.4\times$ real time for directional counts and $8.28\times$ for full event records on one RTX PRO 6000 under the measured top-8 workload. Across eight source clips, the Qwen3.5-Plus agentic workflow averages 4.625 model calls and 26.3 seconds of serial model time per clip. These measurements characterize throughput and processing cost. More timing details are shown in Appendix H. Our current evaluation covers representative sonar and thermal streams. Event recovery remains bounded by candidate coverage, and consolidating overlapping observations in long streams remains an open challenge.

6 Related Works

Video recognition and temporal modeling. Video recognition methods learn spatial and temporal features through convolutional architectures such as SlowFast [21] and Transformers such as TimeSformer [22] and Video Swin [19]. AdaFrame [23] further improves efficiency through adaptive frame selection. These methods advance temporal feature learning and efficient video processing. Our focus is event reconstruction in weak-signal physical streams, where sparse and fragmented observations need to support predictions of timing, direction, and count.

Long-context modeling and memory mechanisms. MeMViT [24], Selective S4 [25], MA-LMM [26], AdaCM^2 [27], ∞-Video [28], Sophia [29], and LongVILA [30] extend long-video understanding through temporal modeling, memory mechanisms, and context management. Access to longer context does not by itself determine how weak observations contribute to event predictions. Event-centric perception examines this evidence use through local inspection and structured event records that support consolidation across windows.

Temporal action localization. Temporal action proposal methods such as BSN [31], BMN [32], and BCNet [33] identify candidate intervals through boundary modeling and proposal scoring. These methods provide relevant foundations for candidate selection and temporal localization. Our event representation includes temporal support alongside task-specific fields such as spatial context, direction, and magnitude, connecting local predictions to stream-level consolidation.

Agentic perception and iterative reasoning. ReAct [34], Toolformer [35], and LATS [36] explore reasoning and tool use in language-model workflows. Multimodal frameworks including LensWalk [37], Active Video Perception [38], EGAgent [39], and VideoAgent [40] further explore adaptive interaction with video evidence. Our frontier workflow uses local inspection and iterative refinement to reconstruct events from weak physical signals. The same event representation also defines supervision targets for compact-model adaptation.

7 Conclusion

We presented event-centric perception for weak-signal physical streams using a shared event representation that connects agentic reasoning, event-aligned adaptation, and stream-level consolidation. Frontier MLLMs refined event predictions through repeated inspection, while compact MLLMs learned to use weak observations through lightweight adaptation. Our sonar and thermal experiments showed improved event recovery, count estimation, and event timing.

These findings support event-centric perception as a unified framework for physical-stream intelligence, connecting the reasoning abilities of frontier models with compact models suitable for local deployment. By turning scattered observations into structured events, the framework helps sensing systems determine what happened, when it happened, and how activity changed over time. Our approach can support ecological monitoring, robotic sensing, and infrastructure inspection, where detecting weak events and tracking changes can inform timely decisions.

Acknowledgments and Disclosure of Funding

We thank the anonymous NeurIPS reviewers for their constructive feedback. This research was supported by an NSERC Discovery Grant. Part of this work was also supported by a Mitacs Accelerate Grant and Mitacs Globalink Research Award.

References

[1] Collin D. Smith, Tyson W. Hatton, and Noah S. Adams. Monitoring fish abundance and behavior using multi-beam acoustic imaging sonar at a selective water withdrawal structure in lake billy chinook, deschutes river, oregon, 2020. Technical Report Open-File Report 2022–1038, U.S. Geological Survey, 2022. U.S. Geological Survey Open-File Report.

[2] Deepak Singh and Matias Valdenegro-Toro. The marine debris dataset for forward-looking sonar semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 3734–3742, 2021.

[3] Elizabeth Bondi, Raghav Jain, Palash Aggrawal, Saket Anand, Robert Hannaford, Ashish Kapoor, Jim Piavis, Shital Shah, Lucas Joppa, Bistra Dilkina, and Milind Tambe. Birdsai: A dataset for detection and tracking in aerial thermal infrared videos. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1747–1756, 2020.

[4] The Cacophony Project. New zealand wildlife thermal imaging. `https://lila.science/datasets/new-zealand-wildlife-thermal-imaging/`, 2023.

[5] Grand View Research. Underwater sonar market size, share & trends analysis report, 2026–2033. `https://www.grandviewresearch.com/industry-analysis/underwater-sonar-market-report`, 2026. Accessed: November 6, 2026.

[6] Grand View Research. Thermal imaging market size, share & trends analysis report, 2026–2033. `https://www.grandviewresearch.com/industry-analysis/thermal-imaging-market`, 2026. Accessed: November 6, 2026.

[7] Justin Kay, Peter Kulits, Suzanne Stathatos, Siqi Deng, Erik Young, Sara Beery, Grant Van Horn, and Pietro Perona. The caltech fish counting dataset: A benchmark for multiple-object tracking and counting. In *European Conference on Computer Vision (ECCV)*, pages 290–311, 2022.

[8] Rongsheng Qian, Chi Xu, Xiaoqiang Ma, Hao Fang, Yili Jin, William I. Atlas, and Jiangchuan Liu. Self-supervised compression and artifact correction for streaming underwater imaging sonar. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, Tucson, Arizona, USA, March 2026.

[9] Suzanne Stathatos, Michael Hobley, Markus Marks, and Pietro Perona. Saved: Learning to denoise low-snr video for improved downstream performance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2026.

[10] Axis Communications. Thermal cameras: White paper. Technical report, 2021. URL `https://whitepapers.axis.com/en-us/thermal-cameras`.

[11] Chi Xu, Rongsheng Qian, Hao Fang, Xiaoqiang Ma, William I Atlas, Jiangchuan Liu, and Mark A Spoljaric. Salina: Towards sustainable live sonar analytics in wild ecosystems. In *Proceedings of the 22nd ACM Conference on Embedded Networked Sensor Systems*, pages 68–81, 2024.

[12] Duzhen Zhang, Yahan Yu, Jiahua Dong, Chenxing Li, Dan Su, Chenhui Chu, and Dong Yu. Mm-llms: Recent advances in multimodal large language models. *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12401–12430, 2024.

[13] Qwen Team. Qwen3.5: Towards native multimodal agents. `https://qwen.ai/blog?id=qwen3.5`, 2026. Official Qwen blog, accessed 2026-05-06.

[14] Gheorghe Comanici et al. Gemini 2.5: Pushing the frontier with advanced reasoning and multimodal understanding. *arXiv preprint arXiv:2507.06261*, 2025.

[15] OpenAI. Introducing gpt-5.2. `https://openai.com/index/introducing-gpt-5-2/`, 2025. Accessed: 2026-05-06.

[16] PyTorch. r3d_18: Torchvision video classification models. `https://docs.pytorch.org/vision/stable/models/generated/torchvision.models.video.r3d_18.html`. Accessed: November 7, 2026.

[17] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

[18] Madison Van Horn, Suzanne Stathatos, Sevan Brodjian, Justin Kay, Kai Van Brunt, Erik Young, Sara Beery, Pietro Perona, and Michael Hobley. Cfc26: Building evaluations for deployment in sonar-based fish counting. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2026. URL `https://arxiv.org/abs/2610.22549`.

[19] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3202–3211, 2022.

[20] Glenn Jocher, Jing Qiu, Mengyu Liu, Shuai Lyu, Fatih Cagatay Akyon, and Muhammet Esat Kalfaoglu. Ultralytics YOLO26: Unified real-time end-to-end vision models, 2026. URL `https://arxiv.org/abs/2606.03748`.

[21] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6202–6211, 2019.

[22] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 2, page 4, 2021.

[23] Zuxuan Wu, Caiming Xiong, Chih-Yao Ma, Richard Socher, and Larry S. Davis. Adaframe: Adaptive frame selection for fast video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1278–1287, 2019.

[24] Chao-Yuan Wu, Yanghao Li, Karttikeya Mangalam, Haoqi Fan, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13587–13597, 2022.

[25] Jue Wang, Wentao Zhu, Pichao Wang, Xiang Yu, Linda Liu, Mohamed Omar, and Raffay Hamid. Selective structured state-spaces for long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6387–6397, 2023.

[26] Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. Ma-lmm: Memory-augmented large multimodal model for long-term video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13504–13516, 2024.

[27] Yuanbin Man, Ying Huang, Chengming Zhang, Bingzhe Li, Wei Niu, and Miao Yin. Adacm^2: On understanding extremely long-term video with adaptive cross-modality memory reduction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8534–8544, 2025.

[28] Saul Santos, António Farinhas, Daniel C McNamee, and Andre Martins. ∞-video: A training-free approach to long video understanding via continuous-time memory consolidation. In *International Conference on Machine Learning*, pages 52877–52893. PMLR, 2025.

[29] Daoze Zhang, Yuze Zhao, Jintao Huang, and Yingda Chen. Sharper and faster mean better: Towards more efficient vision-language model for hour-scale long video understanding. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4423–4439, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.222. URL https://aclanthology.org/2025.acl-long.222/.

[30] Yukang Chen, Fuzhao Xue, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang Yang, Zhijian Liu, Yihui He, Hongxu Yin, Pavlo Molchanov, Jan Kautz, Linxi Fan, Yuke Zhu, Yao Lu, and Song Han. LongVILA: Scaling long-context visual language models for long videos. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=wCXAlfvCy6.

[31] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018.

[32] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3889–3898, 2019.

[33] Haosen Yang, Wenhao Wu, Lining Wang, Sheng Jin, Boyang Xia, Hongxun Yao, and Hujie Huang. Temporal action proposal generation with background constraint. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3054–3062, 2022.

[34] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

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

[36] Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, and Yu-Xiong Wang. Language agent tree search unifies reasoning, acting and planning in language models. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

[37] Keliang Li, Yansong Li, Hongze Shen, Mengdi Liu, Hong Chang, and Shiguang Shan. Lenswalk: Agentic video understanding by planning how you see in videos. *arXiv preprint arXiv:2603.24558*, 2026.

[38] Ziyang Wang, Honglu Zhou, Shijie Wang, Junnan Li, Caiming Xiong, Silvio Savarese, Mohit Bansal, Michael S. Ryoo, and Juan Carlos Niebles. Active video perception: Iterative evidence seeking for agentic long video understanding. *arXiv preprint arXiv:2512.05774*, 2025.

[39] Aniket Rege, Arka Sadhu, Yuliang Li, Kejie Li, Ramya Korlakai Vinayak, Yuning Chai, Yong Jae Lee, and Hyo Jin Kim. Agentic very long video understanding. *arXiv preprint arXiv:2601.18157*, 2026.

[40] Hengji Zhou, Lingxuan Huang, KunpengTan, Si Wu, Lianghao Xia, and Chao Huang. Videoagent: All-in-one agentic framework for video understanding and editing, 2026. URL https://openreview.net/forum?id=cTqGsLYkRl.

A Frontier Inference Diagnostics

We compare direct prompting, recounting, local inspection, and iterative inspection. Table 4 provides the values shown in Fig. 3(a,b), while Table 5 reports additional frontier-model results.

Table 4: Qwen3.5-Plus behavior under direct prompting and successive stages of agentic inference.

Inference	NMAE ↓	NMAE SD ↓	Parse Rate ↑	Behavior
Direct prompting	0.423	0.078	0.86	Weak signals frequently ignored
Prompt + recount	0.357	0.053	0.89	Partial reconstruction under refinement
Local inspection	0.271	0.031	0.93	Local event reconstruction improves
Iterative inspection	**0.225**	**0.022**	**0.96**	Stable event-level reconstruction

Table 5: Representative frontier results under direct and agentic event-centric inference. Qwen, Gemini, and GPT denote Qwen3.5-Plus, Gemini 2.5 Pro, and GPT-5.2, respectively.

Sonar			Thermal			
Model	Workflow	NMAE	Model	Workflow	Recall	tIoU / F1
Qwen	Direct	0.423	Qwen	Direct	0.250	0.227
Qwen	Agentic	0.225	Qwen	Agentic	0.917	0.684
Gemini	Agentic	0.266	Gemini	Agentic	0.920	0.633
GPT	Agentic	0.242	GPT	Agentic	0.956	0.665

B Experimental Setup and Metric Definitions

Data and partitions. The sonar experiments use the Caltech Fish Counting (CFC) dataset [7, 18]. Thermal experiments use wildlife data [4]. The aggregate sonar evaluation contains 620 windows, including 402 positives, 218 negatives, and 517 count-target units. A positive window contains an annotated target event, while a negative window contains none. A separate source-clip evaluation contains 130 source clips and 483 count-target units. These aggregate evaluations are separate from the controlled splits below.

The controlled experiments use Kenai as the sonar training site, with 157 training windows from 32 recordings and 37 windows from five recordings for model selection. The original holdout has 25 windows from eight other recordings. Kenai-59 contains 59 windows from 16 further recordings, disjoint from training, selection, and the original holdout. These partitions use different source recordings so that windows from the same recording stay within one partition. Elwha provides cross-site evaluation with 193 windows from 71 recordings, including 101 positives and 92 negatives. Training, model selection, and proposal-scorer calibration use Kenai data; Elwha serves as the transfer test site.

Evaluation protocols. To evaluate the compact model's use of the available observations, local-prediction comparisons provide predefined labeled windows directly to each model. Candidate-coverage tests separately apply the proposal scorer to source recordings. We assess consolidation through source-clip reconstruction and the controlled Overlap-36 repetition test, described in Appendix E. This separation allows us to examine candidate coverage, weak evidence use, and consolidation at their respective stages.

Metrics and thermal tasks. We report positive event recall, specificity, event F1, and NMAE. For window-level event prediction, recall is the fraction of positive windows correctly identified, and specificity is the fraction of negative windows correctly rejected. Precision is the fraction of predicted positive windows that are positive, and F1 is the harmonic mean of precision and recall. We quantify normalized event error using normalized mean absolute error (NMAE) on passage counts. We also report parse rate, the fraction of responses that satisfy the required event-record format, alongside event-prediction metrics. Appendix G reports unnormalized count errors using MAE, MSE, and RMSE.

Thermal experiments evaluate pretrained frontier models on separate event-reconstruction and occupancy subsets drawn from a source inventory of approximately 121,190 clips. Binary accuracy measures correct event-presence or zone-occupancy decisions. Temporal IoU is the overlap duration divided by the union duration of two intervals, both for proposal suppression and for comparing predicted and annotated events. We report mean temporal IoU (tIoU) for thermal event timing.

Table 6: Compact sonar event-window results on the 620-window benchmark.

Model	Presence	Recall	Specificity	NMAE	Abs. Error
Frozen (generic prompt)	0.338	0.012	0.995	1.021	528/517
Frozen (strict prompt)	0.366	0.041	0.991	0.973	503/517
Adapted (high recall)	0.918	0.931	0.842	0.331	171/517
Adapted (balanced)	0.937	0.914	0.903	0.335	173/517

Table 7: Event-aligned adaptation stability diagnostics. Holdout metrics compare strict prompting with the adapted model; bootstrap deltas are paired estimates over repeated evaluations.

Diagnostic	Recall	Specificity / Acc.	NMAE / Delta
Holdout: frozen, strict prompt	0.083	0.991	0.988
Holdout: adapted	0.887	0.982	0.192
Bootstrap delta	$+0.846$	$+0.522$	-0.694

Zone-entry F1 measures detection of entries into a designated zone. Table 3 reports the two thermal tasks separately.

C Aggregate Results and Holdout Diagnostics

The full aggregate table, source-clip table, and holdout diagnostics are shown here. The aggregate evaluation, original holdout, and Kenai-59 are separate protocols. The adaptation sweep reports additional configurations; its rows should not be interpreted as a controlled single-parameter ablation.

D Supervised Comparisons and Transfer Protocols

All matched local predictors use the same 157 training and 37 selection windows, followed by the Kenai-59 test with disjoint source recordings. R3D-18 and Swin3D-T use frozen encoders pretrained on the Kinetics-400 video dataset, with three independently trained prediction heads per architecture. The compact-model differences relative to Swin3D-T are $+0.056$ in event F1 (95% CI $[0.010, 0.135]$) and -0.094 in NMAE ($[-0.208, -0.024]$). The R3D-18 intervals are given in the main text.

The YOLO26s reference uses 15,817 annotated frames and 10,566 boxes from the same 32 Kenai training recordings. Detector and tracker operating points are selected on the five Kenai selection recordings. No Elwha data is used for training or selection. CFC Baseline++ is evaluated using released tracks under its published protocol and has different supervision. Its results are a specialist reference, not a matched-supervision comparison.

For Kenai-to-Elwha transfer, paired bootstrap over source recordings gives a 95% CI of $[0.571, 0.778]$ for the compact recall improvement and $[-0.779, -0.369]$ for the NMAE change. These local-window results do not include missed stream-level proposals.

E Additional Component Analyses

The context experiments use local windows alone or with static or dynamic background content. Static background is non-changing visual content; dynamic background changes over time and contains no annotated target events. Their adapted F1 values are 0.943, 0.943, and 0.930, respectively. Frozen and adapted models each have parse rate 1.000 on Kenai-59 and Elwha. Together, the context and parseability diagnostics support an event-prediction change beyond format validity.

We examine supervision scale and target content in separate evaluations. The supervision-scale experiment compares a 32-window subset with the full 157-window Kenai training partition, evaluating both on Kenai-59. Both sizes yield recall 0.971 on Kenai-59, while NMAE changes from 0.744 to 0.231. Counts-only and full-event targets evaluate different output requirements. A separate full-event evaluation measures the timing information added to directional passage counts. Crossing F1

Table 8: Compact event-centric inference on the source-clip benchmark.

Model	Presence / Recall	Specificity	NMAE	Abs. Error
Frozen compact MLLM	0.173	–	1.018	492/483
Adapted compact MLLM	**0.741**	–	**0.781**	**377/483**

Table 9: Additional compact sonar split diagnostics.

Split	Condition	Recall	Specificity	Presence	NMAE
Selection	Frozen (generic prompt)	0.011	0.996	0.301	1.024
Selection	Frozen (strict prompt)	0.038	0.991	0.344	0.981
Selection	Adapted	0.942	0.841	0.914	0.338
Holdout	Frozen (generic prompt)	0.017	0.995	0.447	1.033
Holdout	Frozen (strict prompt)	0.083	0.991	0.503	0.988
Holdout	Adapted	0.887	0.982	0.928	0.192

compares predicted passage crossings with annotated crossings using a 1.5-second timing tolerance; it reaches 0.853.

Overlap-36 introduces controlled repetition into 36 structured local predictions. At 25%, 50%, and 100% repetition, naive summation yields NMAE 0.5, 1.0, and 2.0, while consolidation yields 0.0 in each case. This diagnostic evaluates duplicate suppression for the introduced repetitions. It does not establish error-free reconstruction under naturally fragmented, overlapping, or mixed-direction events.

F Proposal Coverage under Noise and Site Transfer

For candidate selection, we train the R3D-18 proposal scorer on the 157-window Kenai training partition and keep it fixed during evaluation. It scans six-second windows at a three-second stride and applies temporal NMS at an IoU threshold of 0.3. We evaluate top-4 and top-8 selection, retaining the four or eight highest-scoring candidates per recording without a probability threshold. The scorer selects these windows using predicted event-presence scores. Any-overlap recall counts an event as covered when a retained window intersects it; event-center recall requires a retained window to contain its midpoint. Severe noise reduces contrast to 0.25 and adds multiplicative speckle noise with standard deviation 0.12. A motion baseline based on frame-to-frame image changes reaches clean Kenai any-overlap recall 0.412 at top-4 and 0.882 at top-8. The learned scorer has its largest coverage advantage at the tighter review budget.

A separate joint diagnostic counts an event as recovered if a proposal covers it and the compact MLLM recognizes it on the corresponding labeled six-second window. On one fixed noise draw, this joint recall is 0.735 at top-4 and 0.882 at top-8; on Elwha it is 0.891 and 0.931. The noise diagnostic uses a single draw, while the coverage table averages three draws. This joint criterion is not a full end-to-end reconstruction evaluation on the selected proposal windows.

G Unnormalized Count Errors

For each six-second window, total count is the sum of leftward and rightward passages. Within each evaluation set of n windows, let c_i and $\hat{c}_i$ denote ground-truth and predicted totals. We report

$$\mathrm{MAE} = \frac{1}{n}\sum_i |\hat{c}_i - c_i|, \qquad \mathrm{MSE} = \frac{1}{n}\sum_i (\hat{c}_i - c_i)^2, \qquad \mathrm{RMSE} = \sqrt{\mathrm{MSE}}.$$

MAE and RMSE are measured in fish per window; MSE is in squared fish-count units.

H Computational Efficiency

Compact timing is measured on one RTX PRO 6000. Candidate scoring requires 0.907 seconds of GPU time per minute of source stream. Each inspected six-second window requires 0.874 seconds

Table 10: Bootstrap deltas for event-aligned adaptation.

Metric	Mean Delta	95% CI
NMAE	-0.694	$[-0.771, -0.612]$
Presence accuracy	$+0.522$	$[0.401, 0.648]$
Positive recall	$+0.846$	$[0.771, 0.914]$

Table 11: Representative adaptation behaviors under different configurations.

Configuration	Selection Recall	Holdout Recall	Holdout NMAE
High-recall adaptation	0.903	0.887	0.381
Balanced adaptation	0.851	0.832	0.247
Lower-rank adaptation	0.693	0.604	0.451

for directional counts or 5.537 seconds for the full event record. Under top-8 selection, the measured workload averages 1.145 inspected windows per minute of source stream. Candidate scoring and local inspection together yield $31.4\times$ real-time throughput for counts and $8.28\times$ for full records.

The frontier timing analysis uses existing traces from eight source clips. Iterative Qwen3.5-Plus inference averages 4.625 model calls and 26.3 seconds of serial model time per clip. Inspections of independent candidate windows can run in parallel, but the reported measurements assume no parallel speedup. Sequential refinement within an event hypothesis retains its data dependencies. These timings describe measured processing costs and workload-dependent throughput, not a guarantee of online control latency.

I Supplementary Reproducibility Details

I.1 Benchmark Protocols

Appendix B describes the sonar and thermal datasets, data partitions, and metric definitions. The aggregate sonar evaluation and source-clip evaluation are separate from the controlled Kenai partitions. Local-prediction comparisons provide labeled windows directly to each model, while candidate-coverage and stream-level consolidation experiments evaluate the corresponding stages on source recordings. This distinction allows us to assess candidate selection, event prediction, and consolidation separately.

Thermal experiments use separate subsets for event reconstruction and zone-occupancy reasoning. These subsets are drawn from a source inventory of approximately 121,190 clips, which includes false triggers and animal activity. The inventory size describes the source collection; the reported results concern the task-specific evaluation subsets.

I.2 Model Families and Inference Settings

The frontier experiments use GPT-5.2, Gemini 2.5 Pro, and Qwen3.5-Plus through hosted APIs, without domain-specific adaptation. The compact experiments use Qwen3.5-9B. Direct prompting produces a structured event prediction in one pass. For frontier models, agentic inference uses a global overview to select candidate regions, inspects local windows, and revisits ambiguous predictions before stream-level consolidation. Thermal inference follows the same workflow with task-specific prompts and event fields.

For the compact model, local evaluation parses responses into event records and compares their predictions with event-presence, direction, and count targets. The adaptation sweep trains LoRA adapters and evaluates them on the model-selection split. Selected configurations are then evaluated on the held-out split.

Table 12: Proposal coverage. Each pair reports any-overlap / event-center recall. Severe-noise values are mean ± SD over three draws.

Evaluation	Recordings	Top-4	Top-8
Held-out Kenai	16	$0.824/0.765$	$0.912/0.912$
Severe noise	16	$0.745 \pm .017/0.627 \pm .017$	$0.873 \pm .034/0.784 \pm .017$
Elwha	71	$0.911/0.911$	$1.000/0.990$

Table 13: Raw count errors on separate evaluations with disjoint source recordings.

Evaluation	Compact model	MAE	MSE	RMSE
Original holdout (25 windows)	Frozen	0.520	0.600	0.775
	Adapted	0.080	0.080	0.283
Kenai-59	Frozen	0.644	0.847	0.921
	Adapted	0.153	0.186	0.432

I.3 Adaptation Configuration and Selection Procedure

Compact adaptation learns low-rank updates to selected transformer projections. Counts-only targets specify leftward and rightward passage counts, while full-event targets also include event timing and crossing information. Both use fields from the shared event representation.

The default training configuration uses a per-device batch size of 1, gradient accumulation of 1, a learning rate of 2×10^{-4}, a warmup ratio of 0.03, a cosine learning-rate schedule, weight decay of 0.0, gradient clipping of 1.0, and gradient checkpointing. The optimizer is paged AdamW 8-bit. The default LoRA configuration uses rank 16, alpha 32, and dropout 0.05 for causal language modeling, with a sequence budget of 4096 tokens. Individual sweep configurations override the corresponding defaults.

The adaptation sweep considers ranks $\{16, 32\}$, learning rates $\{10^{-4}, 2 \times 10^{-4}, 3 \times 10^{-4}\}$, update counts $\{16, 24, 32\}$, and dropout values $\{0.0, 0.05\}$. The core sweep emphasizes learning rates of 10^{-4} and 2×10^{-4}. Supervision variants include positive-event upweighting, positive-plus-negative weighting, and count-emphasized targets. The high-recall and balanced operating points are selected on the model-selection split before held-out evaluation.

I.4 Code and Reproduction Entry Points

We provide accompanying code for event-centric inference, structured output parsing, event-level evaluation, and compact adaptation. Benchmark configurations specify input clips, data splits, event targets, prompt templates, and output schemas, connecting the implementation to the experimental protocols described in Appendix B.

The compact training and evaluation scripts use the following entry points:

```
bash scripts/train_compact_qlora.sh
bash scripts/run_compact_lora_sweep.sh
bash scripts/eval_compact_base.sh
bash scripts/eval_compact_adapted.sh
```

For frontier inference, the code provides the modules `physagent.experiments.prompt_batch` and `physagent.experiments.event_agent_batch` for direct and agentic inference. Thermal evaluation uses `physagent.experiments.thermal_prompt_batch` and `physagent.experiments.thermal_event_agent_batch`. Paths are relative to the code root. Hosted-model experiments also require access to the corresponding provider APIs.

I.5 Uncertainty and Statistical Analysis

The local compact adaptation analysis uses paired bootstrap comparisons against the frozen model under strict prompting. It measures changes in normalized mean absolute error (NMAE), event-presence accuracy, and positive event recall, with paired resampling at the evaluation-example level.

Table 14: Compute resources and schedules. Sweep budgets refer to complete searches; update counts refer to individual adaptation runs.

Experiment	Hardware or service	Schedule or cost measure
Compact local evaluation	RTX PRO 6000 (96 GB)	Per-window inference time
Compact adaptation	RTX PRO 6000 (96 GB)	Typically 24–32 updates, batch size 1
Adaptation sweep	RTX PRO 6000 (96 GB)	4, 6, or 8 hours per sweep
Frontier inference	Hosted model APIs	Model calls and serial model time

Statistical results for the additional controlled comparisons follow the protocols specified in their corresponding appendix sections.

I.6 Compute Resources

Compact adaptation and local evaluation use one NVIDIA RTX PRO 6000 Blackwell Workstation Edition GPU with 96 GB of memory, running Ubuntu with PyTorch and Transformers. Training runs typically use 24–32 update steps at batch size 1. The focused adaptation sweeps use single-GPU budgets of four, six, or eight hours. These budgets describe the complete sweeps, while the update counts describe individual training runs.

Frontier inference uses hosted model APIs. Its processing cost depends on the number of model calls and their response times. The timing analysis reports measured serial model time and distinguishes independent candidate inspections from sequential refinement of the same event hypothesis.

J Supplementary Responsible-Use Details

J.1 Broader Impacts

Event-centric perception can support ecological monitoring by reducing manual review and producing passage counts, event timing, and occupancy estimates that can be checked against the sensing stream. Structured event records also help researchers inspect predictions and trace errors to candidate selection, local inspection, or stream-level consolidation.

False event predictions and missed activity can affect monitoring decisions. Overcounting and undercounting therefore remain relevant when interpreting ecological trends or planning interventions. Related sensing methods could also support persistent surveillance in settings where the subjects of monitoring have limited control over data collection. The intended use is decision support with expert review, especially when predictions inform consequential decisions.

The shared event representation supports review by retaining temporal support, direction, and magnitude. Separating local prediction from stream-level consolidation further helps identify where an error occurs and which stage needs correction.

J.2 Accompanying Code and Experimental Materials

The code package includes the inference pipeline, event-record parser, evaluation routines, compact adaptation trainer, and parameter-sweep scripts. Supporting configurations document the data partitions, supervision targets, and task-specific prompts used by these components. The documentation distinguishes source sensing data, derived event labels, and model-generated predictions. Access to third-party data and hosted models follows the terms of the original providers.

J.3 Existing Datasets, Models, and Software

The experiments use the CFC sonar dataset [7, 18] and thermal wildlife data [4]. The frontier model families are documented in the cited provider publications [13–15]. Table 15 summarizes the roles of the datasets, models, and software used in the experiments.

Table 15: Existing datasets, models, and software used in the experiments.

Resource	Role in the experiments
CFC sonar dataset	Sonar event prediction and passage counting
Thermal wildlife dataset	Event reconstruction and zone-occupancy evaluation
Qwen3.5-9B	Compact inference and event-aligned adaptation
Qwen3.5-Plus	Frontier direct and agentic inference
Gemini 2.5 Pro	Frontier direct and agentic inference
GPT-5.2	Frontier direct and agentic inference
PyTorch, Transformers, and PEFT	Model execution and compact adaptation

J.4 Human Subjects and Ethics Scope

The experiments analyze existing sonar and thermal sensing streams. They do not involve crowdsourcing, human-subject data collection, participant studies, or interventions on human subjects. The ethical considerations concern the use of existing data and models and the consequences of downstream deployment.

J.5 Declaration of LLM Usage

Multimodal LLMs are a core component of the method. Frontier models perform direct and agentic inference, while the compact model is evaluated before and after event-aligned adaptation. Prompts, event fields, candidate windows, and supervision targets define how these models contribute to the reported predictions.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: We state the scope as event-centric perception for weak-signal physical streams, evaluated on imaging sonar and thermal sensing. The main claims cover the event-centric formulation, the weak-signal failure regime, and the proposed event-aligned reasoning and adaptation approach. These claims are supported by the reported experiments and are scoped to the evaluated sensing domains.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss limitations in evaluation scope, event-window dependence, and long-horizon consolidation. The empirical claims are scoped to imaging sonar and thermal sensing. Local event reconstruction does not automatically solve stream-level aggregation, particularly for high-throughput, overlapping, or mixed-direction cases. The approach also depends on the quality of candidate windows and structured event supervision, which can affect recall, specificity, and count calibration.

   Guidelines:

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

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

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

   Answer: [N/A]

   Justification: We do not make formal theoretical claims or present theorem-level results. The mathematical notation is used only to specify the event-centric task formulation, matching objective, inference procedure, and adaptation loss. The main claims are empirical and are supported through experiments on the evaluated sensing domains.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We provide the experimental setup needed to reproduce the main claims, including benchmark sizes, split construction, event-window targets, model families, inference regimes, adaptation setup, evaluation metrics, and reproduction entry points. The supplementary material specifies compact adaptation scripts, compact baseline and adapted evaluation scripts, frontier inference entry points, and thermal event-centric inference commands. For hosted frontier models, reproducibility depends on access to the corresponding provider endpoints, but the paper reports the prompt, schema, and evaluation protocol needed to support independent verification.

   Guidelines:

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

(a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [No]

   Justification: We provide code and fixed entry points for the event-centric inference pipeline, structured output parsing, event-level evaluation logic, compact adaptation trainer, and sweep scripts. The supplementary material describes benchmark configuration files, split construction, event-window targets, prompt templates, output schemas, and relative artifact paths. However, some raw sensing data, derived annotations, and closed-source frontier-model outputs may be subject to third-party access restrictions or API availability. Fully open reproduction of every experiment is therefore not guaranteed at submission time, although the released code and documentation support reproduction where data access permits.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: The paper and supplementary material specify the evaluated domains, benchmark sizes, local event-window and parent-clip settings, model-selection and held-out splits, model families, inference workflows, adaptation procedure, and evaluation metrics. The

supplementary material further reports compact adaptation details, including LoRA rank, alpha, dropout, learning rate, optimizer, batch size, gradient accumulation, warmup ratio, schedule, weight decay, gradient clipping, checkpointing, sequence budget, sweep axes, and operating-point selection procedure.

Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: The main compact adaptation claim is supported with paired bootstrap analysis. We report bootstrap deltas and 95% confidence intervals for normalized event error, presence accuracy, and positive recall against the strict prompted compact baseline. The bootstrap is paired at the evaluation-example level under identical output constraints. Frontier-model and thermal results are reported as point estimates because they serve as capability references and cross-domain support rather than the main model-selection axis.

   Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: The supplementary material reports the compute setting for the compact local evaluation, compact adaptation, focused sweeps, and frontier inference. Compact adaptation and local evaluation use a single NVIDIA RTX PRO 6000 Blackwell Workstation Edition GPU with 96GB memory. The compact runs use short structured adaptation schedules, typically 24–32 update steps at batch size 1. The focused sweeps use four-hour, six-hour, and eight-hour single-GPU harness budgets. Frontier-model experiments are executed through hosted APIs and are described as API-latency bound, with reproducibility depending on access to the corresponding provider endpoints.

Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The work studies machine perception on existing sensing streams and hosted-model inference. It does not involve crowdsourcing, human-subject data collection, participant studies, or interventions on human subjects. The supplementary material discusses responsible-use considerations, including risks from weak-signal monitoring, false event claims, systematic misses, and possible misuse in persistent monitoring settings. The paper frames the method as decision-support infrastructure and describes safeguards based on structured event records, post-hoc review, auditing, and threshold calibration.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: The supplementary material discusses positive impacts in scientific and operational settings that depend on weak-signal sensing, including ecological monitoring, wildlife passage estimation, infrastructure inspection, and reduced manual review burden. It also discusses negative impacts and risks, including false event claims, systematic misses, possible use in surveillance or persistent monitoring contexts, and the need to treat the method as decision support rather than a fully autonomous replacement for expert review.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The release consists of research code and experimental configurations and does not include high-risk datasets or pretrained models. Potential deployment risks and the role of expert review are discussed in Appendix J.1.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [No]

    Justification: The supplementary material identifies the existing assets used in the paper, including sonar and thermal datasets, compact and hosted multimodal models, and the PyTorch/Transformers/PEFT software stack. It also includes an asset table listing the source to cite and the governing terms to verify at submission time. However, exact release identifiers and license names are not yet fully filled in for every asset. We therefore do not claim that all licenses and terms are completely specified in the current version.

    Guidelines:

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

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [Yes]

    Justification: The paper introduces derived assets beyond raw source data, including structured event-window JSONL files, benchmark split files, event-centric prompt templates, structured event schemas, and model-generated outputs. The supplementary material documents the construction procedure, intended use, and limitations of these derived assets, and distinguishes raw sensing data, derived event labels, and model-generated outputs. Release is subject to anonymous review and third-party data constraints where applicable.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: The reported experiments do not involve crowdsourcing, human-subject data collection, participant studies, compensation, or interventions on human subjects. The work studies machine perception on existing sensing streams and hosted-model inference.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: The reported experiments do not involve crowdsourcing, human-subject data collection, participant studies, or interventions on human subjects. The main ethics issues concern data/model use, release practice, and downstream deployment context rather than participant risk.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [Yes]

    Justification: LLMs are a core methodological component of this research. Frontier multimodal LLMs are used as direct and event-centric inference systems in the capability study, and a compact multimodal LLM is used as the local adaptation substrate in the weak-signal adaptation study. The prompts, structured event schemas, localized event windows, and adaptation objectives directly affect the reported scientific results. LLM usage is therefore central to the method and is declared in the supplementary material.

    Guidelines:

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