HEIMDALL: HINDSIGHT DISTILLATION FOR PROACTIVE WARNING IN STREAMING VIDEO

Anonymous authors
Paper under double-blind review

ABSTRACT

Streaming vision-language models can judge whether a scene is dangerous. Proactive warning asks them to say so before the accident, within a preset cap on false alarms, and on the device beside the camera. However, training such a warner requires supervision, and the annotated accident times that anticipation methods train on are expensive to obtain and absent for a newly deployed camera's footage. We observe that a frozen large verifier is uncertain at the moment a warning is due but confident a few seconds later, once the accident is in view. This later verdict can become a training target for the earlier moment without any annotation. We propose Heimdall, which queries a frozen 8B verifier a few seconds after each window of the training stream, converts those later verdicts into a multi-horizon risk target for the window itself, and distills on that target a 2B model that runs in real time on a Jetson. At deployment the two models run as one online policy. Each alarms when its risk score crosses its threshold, and the two thresholds are chosen together so that the policy as a whole stays within the cap. On the held-out test split of PaSBench-Video, Heimdall warns before the accident on 1.5 times as many clips as a zero-shot model with the same verifier at half its false-alarm rate, and on a third more than a model distilled from the same verifier at the same moment. Published warners that find more accidents raise two to four times as many false alarms as the cap allows.

1 INTRODUCTION

Cameras already watch kitchens, care homes, factory floors, and vehicles (Zhao et al., 2025a; Schneider et al., 2026; Epstein et al., 2020; Zhao et al., 2026a), and vision-language models can now judge whether a live scene is dangerous (Qiu et al., 2026; Panpatil et al., 2026). A warning is useful only if it arrives a second or two before the accident, which turns recognition into a timing problem (Li et al., 2020; Eyzaguirre et al., 2024). A deployed warner also meets two requirements that no benchmark score captures. A small model has to score every window on the device beside the camera, since a large model on a server is too slow and costly to ask that often (Chen et al., 2024b; Gupta et al., 2024). The false alarms of the whole warner are also capped in advance, while benchmarks only measure them on normal footage (Zhao et al., 2026a; Panpatil et al., 2026).

Existing systems obtain the evidence for such a warning in two ways. Streaming memories and proactive assistants keep past frames, events, or caches and decide when to speak (Di et al., 2025; Chen et al., 2026; Zeng et al., 2025; He et al., 2024; Zhang et al., 2025; Liang et al., 2026; Zhang et al., 2026; Wang et al., 2025), but they answer questions whose answer lies in the past, and none is calibrated to a false-alarm cap. Accident anticipation instead scores how dangerous the seconds before an accident look (Bao et al., 2021; Karim et al., 2022; Liu et al., 2026), and its most effective recent methods learn from the future of the training video, so that they anticipate an accident instead of confirming it. Temporally shifted distillation (Patera et al., 2026) trains a lightweight student on a frozen teacher's features one step ahead, and RiskProp (Zou et al., 2026) propagates the model's own next-frame risk backward from the collision. However, both anchor their supervision on a hand-marked accident time, which a newly deployed camera's footage does not have.

Learning such a warner from the footage of a new camera raises two challenges. *The first challenge is to obtain supervision for the precursor without annotated accident times.* Anticipation methods take the label of an early window from the annotated onset or collision, and a new camera's footage

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

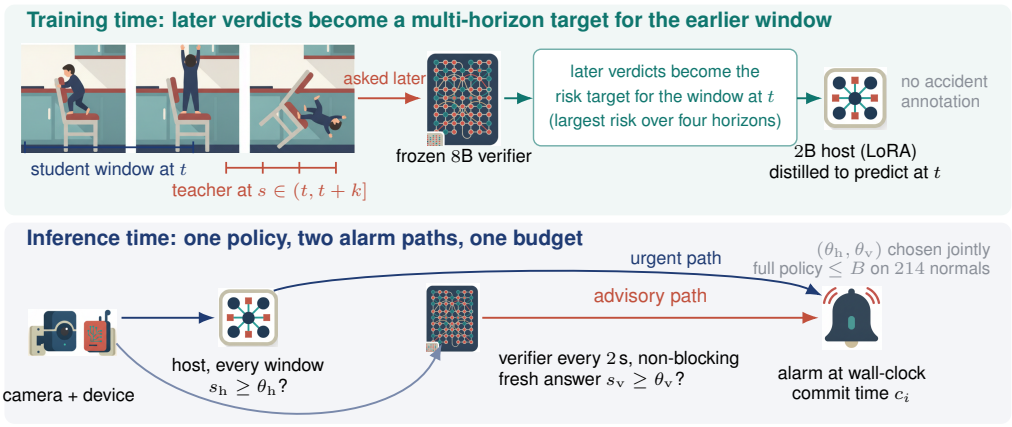


Figure 1: Overview of Heimdall. Training: the frozen verifier's verdicts a few seconds after each window form the risk target for the 2B host. Inference: host and verifier alarm on thresholds chosen jointly under one false-alarm cap.

has neither. Evidence that arrives later in the stream is a candidate label, but it is not the probability of an accident at the earlier moment. What such a label estimates and how the small model learns a precursor from it must be settled before it can serve as supervision. *The second challenge is to bound the false alarms of an online policy in which two models can each raise the alarm.* Question-answering accuracy (Fu et al., 2025) says nothing about false alarms on a stream, and selective prediction and conformal calibration bound the error of one model on independent samples (Geifman & El-Yaniv, 2017; Angelopoulos & Bates, 2023; Angelopoulos et al., 2024a). A warner counts false alarms per clip over a horizon, its two models see the same footage at different times, and a threshold that is safe for each of them alone is not safe for their union. The cap therefore has to be set for the whole policy as it runs, with both models and their timing.

Our observation is that the supervision is already present in the stream. A large frozen vision-language model (Bai et al., 2025) that hesitates a few seconds before an accident answers with confidence once the accident is in view, so queried a few seconds after a window it knows what followed. Heimdall turns this into hindsight distillation. A few seconds after each training window, the frozen 8B verifier that will run beside the 2B host is asked how soon an accident will happen, and the largest risk it reports over the following seconds becomes the target for the earlier window (Figure 1). The host distilled on this target learns to see at t what the verifier confirms only later. At deployment the two models run as one policy whose thresholds are calibrated together on a pool of normal clips, so one cap bounds both alarm paths. A miss-rate identity under this budget shows why supervision works where a streaming memory does not, since at the threshold the budget fixes a memory can only shift scores. The main contributions are summarized as follows.

- **A label-free supervision source for proactive warning.** We identify the deployed verifier's own later verdicts as supervision for the earlier window, turn them into a multi-horizon risk target, and characterize what this target estimates relative to the annotated event.
- **Hindsight distillation of the on-device host.** We distill a 2B host on this target, and same-time and fixed-offset teachers isolate what the future adds to distillation.
- **A jointly calibrated two-model policy and its analysis.** Host and verifier run as one online policy whose two thresholds are set together under one false-alarm cap, and a miss-rate identity explains why a streaming memory cannot move the operating point at a low false-alarm rate while supervising the host can.
- **A real-time system and its evaluation.** The loop runs in real time on Jetson boards. On a held-out test split Heimdall warns before the accident on half again as many clips as the same two models without distillation at half their false alarms, and the gain holds under two further caps and on a multi-source confirmation corpus.

2 Related Work

2.1 Streaming Memory and Proactive Warning

Online video assistants keep past evidence so that a later question can be answered from it. Key-value caches retain tokens across the stream (Di et al., 2025; Chen et al., 2026), event and scene memories summarize what has happened (Zeng et al., 2025; He et al., 2024; Song et al., 2024; Zhao et al., 2026b), and token compression keeps the budget bounded over long footage (Zhang et al., 2025; Xie et al., 2026; Islam et al., 2025). Their benchmarks are question answering and captioning, where the answer mostly lies in the past (Fu et al., 2025; Huang et al., 2025). A second line of work decides when a streaming assistant should speak (Chen et al., 2024a; Qian et al., 2025; Yang et al., 2025; Wang et al., 2025; Niu et al., 2025), and event and hierarchical memories such as OASIS and HERMES (Liang et al., 2026; Zhang et al., 2026) retrieve past evidence on demand for a current judgment. A memory changes what the model sees, not what it has learned to expect, so it can move a score only as far as the retrieved evidence moves it. None of these systems is evaluated under a fixed false-alarm cap on a running loop, and we evaluate OASIS and HERMES under one.

Proactive warning systems ask a model whether danger is imminent and act on its answer. The PaSBench-Video protocol (Zhao et al., 2026a) queries a model every few hundred milliseconds over a short look-back window with a carried state summary and scores warnings inside a lead-time window against a no-risk split. SPRINT (Qiu et al., 2026) probes a large model on an anchored window that grows from the clip start, and a recent driver-alerting policy holds an uncertain alert to observe longer before it speaks (Wang et al., 2026b). Each of them runs at the operating point its prompt or rule implies, and none learns from the footage it will run on. We run the first two as baselines.

2.2 Learning from the Future and Calibrating the Warner

Driving anticipation models are trained on annotated accident times and report average precision or AUC and time to accident within each corpus (Bao et al., 2021; Karim et al., 2022; Liu et al., 2026). Their supervision comes from frames the model will not see at test time. A teacher that sees the whole clip has distilled early action predictors and online action detectors (Wang et al., 2019; Tran et al., 2021; Zhao et al., 2022), temporally shifted distillation (Patera et al., 2026) trains a lightweight student on a frozen image teacher's features one step ahead, and RiskProp (Zou et al., 2026) propagates the model's own next-frame risk backward from the collision. V-JEPA 2 (Assran et al., 2025; Mur-Labadia et al., 2026) learns video representations by predicting masked latents, and FLaRA (Caselli et al., 2026) anticipates accidents from its predicted future latents. All of them anchor the loss on an annotated onset or collision, so the annotation and not the stream decides what counts as early. Heimdall keeps the idea of learning from the future and replaces the annotation with the verdict of the deployed verifier a few seconds later.

On the deployment side, two-model cascades and selective prediction decide when a small model defers to a large one or abstains (Gupta et al., 2024; Jitkrittum et al., 2023; Chen et al., 2024b; Ong et al., 2025; Geifman & El-Yaniv, 2017), and conformal methods give distribution-free risk guarantees under shift (Angelopoulos & Bates, 2023; Angelopoulos et al., 2024a;b; Farinhas et al., 2024). Heimdall instead sets the thresholds of its two models together on a pool of normal clips before any evaluation, so that one cap bounds the whole warner.[1]

3 Heimdall

The warner has two models. A 2B host runs on the device and scores every window, and a frozen 8B verifier runs on a server and answers a risk question on a fixed schedule. Either can raise the alarm, and the verifier that answers online is also the teacher that supervises the host offline. This section fixes the task and the cap, then shows how the later verdicts train the host, how the two models act as one policy, and what bounds the miss rate of such a policy under a false-alarm budget.

[1]The term hindsight distillation has been used for agents that learn from post-action feedback and for answer rationales (Li et al., 2026; Zhao et al., 2025b). We use it for the later verdicts of the same verifier on a video stream, given a few seconds after the window.

3.1 PRELIMINARIES: BUDGETED PROACTIVE WARNING

Stream and decisions. We consider a camera that watches one site and produces frames continuously, as in Figure 1. A warner processes them online window by window and may raise an alarm only at decision times $t_1 < t_2 < \dots$ on a fixed grid. Clip i carries a label $y_i \in \{\text{hazardous}, \text{normal}\}$, and a hazardous clip also carries the accident time a_i annotated by the corpus. Every clip is truncated at an observation horizon $H_i = \min(H_{\max}, \text{duration}_i)$, and evidence that arrives after H_i counts toward cost but cannot raise an alarm.

Hits and false alarms. Write c_i for the alarm time of clip i, the instant at which the running system commits the alarm. A hazardous clip is a hit at margin τ when $c_i \le a_i - \tau$, written $\mathrm{R}\tau$ and reported as R0, R3, and R5 for $\tau \in \{0, 3, 5\}$ s. R0 counts any alarm before the accident, R3 is the any-detection rule of PaSBench-Video (Zhao et al., 2026a) at a three-second lead, and R5 is the longest horizon the host is trained to predict. A normal clip is a false alarm when any $c_i \le H_i$ exists.

Budget and threshold. A deployment fixes how many false alarms are acceptable, not a probability, so the cap is a clip-level budget B on an independent pool of N normal clips from the deployment distribution that is never used for evaluation. For a single score stream with per-window scores s_{ij}, the threshold at budget B is

$$\theta(B) = \min\{\, \theta \in \Theta : \ \#\{i \in \text{pool} : S_i \ge \theta\} \le B \,\}, \tag{1}$$

where $\#$ counts clips, $S_i = \max_j s_{ij}$ is the running maximum within the horizon, and the candidates Θ are zero and a value just above each pool maximum S_i. The resulting false-alarm rate is a clip-level rate over the horizon.

System budget. The system alarms whenever the host or the verifier alarms, so bounding each of the two at B bounds the system only at $2B$, and we therefore calibrate the whole policy. Let π_λ be the policy with every path threshold set by Eq. (1) at a common per-path budget $\lambda \in \{1, \dots, B\}$, and let $\mathrm{FP}_{\text{pool}}(\pi_\lambda)$ be the number of pool clips that alarm when the full policy is replayed on the pool's recordings. The operating point is

$$\lambda^\star = \max\{\lambda : \ \mathrm{FP}_{\text{pool}}(\pi_\lambda) \le B\}, \tag{2}$$

and every two-model configuration is reported at its own $\lambda^\star$, calibrated on its own pool.

3.2 HINDSIGHT DISTILLATION

Heimdall supervises the on-device host with what the frozen verifier sees a few seconds later, so that no accident time has to be annotated.

Target. For a window ending at t, a warner wants the probability of the event Y_t^k that an accident happens in $(t, t+k]$. The four horizons $k_1 < k_2 < k_3 < k_4$ lie between one and five seconds and bound the buckets of the five-way time-to-accident question the host answers. The teacher is the frozen verifier itself. It is asked the same question in the host's window format at each whole second $s \in (t, t+k]$, its answer at s gives the probability that an accident follows within three seconds of s, and the largest of these answers over $(t, t+k]$ is the risk score of the window at t,

$$R_k(t) = \max_{s \in (t,\, t+k]} q_{\text{imm}}(x_{\le s}), \qquad \hat{p}(t) = \big(1 - R_{k_4},\ R_{k_4} - R_{k_3},\ R_{k_3} - R_{k_2},\ R_{k_2} - R_{k_1},\ R_{k_1}\big), \tag{3}$$

where q_{imm} is the teacher's probability that an accident occurs within three seconds of s, $x_{\le s}$ is the causal window ending at s, and the five entries of $\hat{p}(t)$ are the probabilities of the five buckets. R_k leads the event by up to one teacher horizon, so it is a risk score whose threshold is set on the pool and not a probability of Y_t^k.

Training. The student minimizes the cross-entropy between $\hat{p}(t)$ and its five-bucket output on $x_{\le t}$,

$$\mathcal{L}(\phi) = \sum_t \mathrm{CE}\big(\hat{p}(t),\ f_{\text{h}}^{\phi}(x_{\le t})\big), \tag{4}$$

where ϕ denotes the LoRA parameters of the 2B vision-language host. Teacher and student answer the same question, and only the moment of asking changes, which is why the host is a language model and not an encoder with a classification head. Training windows lie on a one-second grid

Algorithm 1 Online warner for one clip

Require: host f_{h}, verifier f_{v}, thresholds $(\theta_{\mathrm{h}}, \theta_{\mathrm{v}})$ from Eq. (2), period P, freshness T_a, horizon H_i

alarm $\leftarrow$ none
procedure ONHOST(window ending at t, available at wall-clock $u \le H_i$)
  $s_{\mathrm{h}} \leftarrow \Pr_{f_{\mathrm{h}}}[\text{accident} \le 3\,\mathrm{s} \mid \text{window}]$
  if alarm is none and $s_{\mathrm{h}} \ge \theta_{\mathrm{h}}$ **then alarm** $\leftarrow u$ ▷ urgent path
  end if
  if t is on the P-second grid **then** issue f_{v} on the causal window ending at t ▷ non-blocking
  end if
end procedure
procedure ONVERDICT(answer j with $s_{\mathrm{v}}(j)$, arriving at wall-clock $a \le H_i$)
  if $a - \mathrm{issue}(j) \le T_a$ and **alarm** is none and $s_{\mathrm{v}}(j) \ge \theta_{\mathrm{v}}$ **then alarm** $\leftarrow a$ ▷ advisory path
  end if
end procedure
all windows and answers are logged to H_i whether or not an alarm has been committed

wherever five seconds of stream follow, and we keep every window whose longest-horizon target exceeds a small threshold and twice as many others at random, so no annotation enters the selection.

Controls. Two controls keep these windows and change only when the teacher is asked. The same-time control asks it at t itself, which separates the value of the future from that of distillation, and the fixed-offset control asks it once at $t + 4\,\mathrm{s}$, one second past the three-second question horizon.

3.3 THE ONLINE POLICY

At deployment the host and the verifier run as one policy, given in Algorithm 1 for one clip. The host scores the one-second window ending at t, its five-bucket output is collapsed to the probability s_{h} of an accident within 3 s, and the clip alarms at once when $s_{\mathrm{h}} \ge \theta_{\mathrm{h}}$. This is the urgent path. A verifier request on the causal window is sent every P seconds without waiting for the answer, so the request schedule does not depend on the thresholds and calibration cannot change what the verifier sees. A fresh answer s_{v} arrives within T_a of its request and alarms when $s_{\mathrm{v}} \ge \theta_{\mathrm{v}}$. This is the advisory path. The two thresholds are the only free parameters, and Eq. (2) sets them.

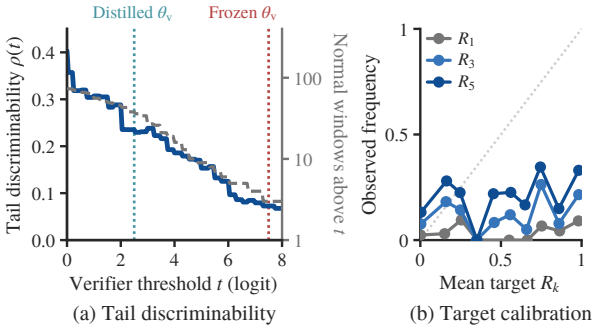


Figure 2: Tail discriminability $\rho(t)$ of the frozen verifier on the development windows (a), and calibration of the target R_k on the test split (b).

3.4 ANALYSIS: MISS RATE UNDER A FALSE-ALARM BUDGET

Setup and assumptions. This subsection uses an idealized window model to read the experiments. Write $F_0(\theta) = \Pr_{\text{normal}}[s < \theta]$ for the distribution of a window score on normal footage and $F_1(\theta) = \Pr_{\text{signal}}[s < \theta]$ for a window that contains a visible precursor, both on one common monotone scale, so that a threshold means the same for both.

Assumption 1 (Window model)**.** A normal clip contributes M_0 windows drawn from F_0. A hazardous clip evaluated at margin τ contributes M_τ windows that are available before $a_i - \tau$, of which $k_\tau \ge 1$ are signal windows drawn from F_1 and the rest from F_0. Windows are independent within a clip.

Proposition 1 (Miss identity)**.** Under Assumption 1, let θ_α satisfy $F_0(\theta_\alpha)^{M_0} = 1 - \alpha$, so that a normal clip alarms with probability α. Then

$$\Pr[\mathrm{miss}_\tau] = F_0(\theta_\alpha)^{M_\tau - k_\tau}\, F_1(\theta_\alpha)^{k_\tau} = (1-\alpha)^{M_\tau/M_0}\, e^{-k_\tau \rho(\theta_\alpha)}, \qquad \rho(\theta) = \log \frac{F_0(\theta)}{F_1(\theta)}, \tag{5}$$

with $\rho(\theta) \ge 0$ whenever $F_1(\theta) \le F_0(\theta)$.

On our development set a normal clip carries about 17 windows and the pre-accident part of a hazardous clip about 4, so the miss rate is governed by the budget term $(1-\alpha)^{M_\tau/M_0}$ and by $k_\tau\rho$, the amount of pre-accident evidence times the tail discriminability ρ, which measures how well each piece separates hazard from normal at the threshold.

Corollary 1 (Score-shift ceiling)**.** Let a mechanism map every signal-window score s to $s+\delta_j$ with $0 \le \delta_j \le \delta$ and leave the normal distribution unchanged, so that θ_α is unchanged too. A hazardous clip newly detected after the shift must have its pre-shift clip maximum in $[\theta_\alpha - \delta, \theta_\alpha)$, so the recall gain is at most $\Pr_{\text{hazard}}[\max_j s_j \in [\theta_\alpha - \delta, \theta_\alpha)]$.

Figure 2(a) shows that at the frozen verifier's operating threshold ρ is a quarter of its value at two logits, so one more piece of evidence there barely moves the miss rate off the budget term. The gain has to come from raising ρ where the threshold sits, which is what supervising the host does on the urgent path, and a memory that only shifts scores is bounded by Corollary 1.

4 Experiments

4.1 Setup

Corpora. PaSBench-Video (Zhao et al., 2026a) is released as an evaluation set with no training split, so we split it ourselves at the video level into training, validation, and test videos. The test split is held out from training and from every threshold choice. The development set is a subset of the validation split and was used to choose the method. Thresholds come from a pool of normal clips drawn from two external sources (Zhao et al., 2025a; Cao et al., 2023) and never used in training or evaluation, so the pool needs footage known to be normal and nothing else. Two corpora confirm the results after every threshold is frozen. The confirmation corpus draws hazardous and normal clips from several sources, including OOPS (Epstein et al., 2020), SmartHome-Bench (Zhao et al., 2025a), and four fall-detection sets (Zhang et al., 2014; Schneider et al., 2026; Eraso Guerrero et al., 2022). The second is a self-collected set from one site that has its own calibration normals.

Models. The host is Qwen3-VL-2B trained with LoRA and the verifier is Qwen3-VL-8B, frozen throughout (Bai et al., 2025).

Baselines. OASIS (Liang et al., 2026) is a training-free hierarchical event memory with on-demand retrieval, run with its official code on the same verifier and question. **HERMES** (Zhang et al., 2026) keeps a hierarchical key-value memory over the stream, run with its official code on a Qwen2.5-VL-3B backbone. **StreamSense** (Wang et al., 2026a) routes between a small and a large model with invoke, skip, and defer decisions. Its released router runs unchanged over the per-window scores of our host and verifier and fires at a fixed probability of one half, so we set its operating point on the pool. **SPRINT** (Qiu et al., 2026) probes the large model on an anchored window that grows from the clip start, and we step it online through the same 8B every half second. The **PaSBench-Video** protocol (Zhao et al., 2026a) queries the model every 0.3 s with a carried state summary, and we reimplement it from the paper because no code was released. Every calibrated baseline gets its own pool and frozen threshold at the same budget.

Metrics. Recall at margin τ is the share of hazardous clips whose alarm commits at least τ seconds before the accident, and we report it as R0, R3, and R5. The false-alarm rate FAR is the share of normal clips with any alarm inside the horizon. Per-clip calls and latency are the cost.

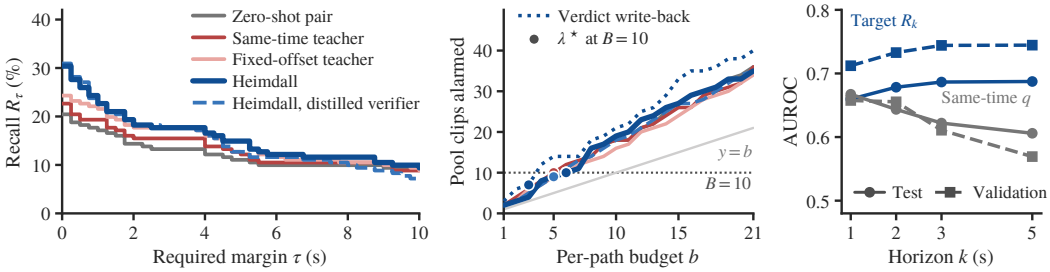
Implementation. Batches run in real time on a DGX Spark, and edge measurements use two Jetson Orin boards. The budgets $B = 4$, 10, and 21 are 2, 5, and 10 percent of the pool, $B = 10$ is the main operating point, and $H_{\max} = 40$ s. Thresholds are chosen jointly for the full policy by Eq. (2) on each configuration's own recorded pool, and every alarm is replayed from recorded real-time logs whose schedules do not depend on the thresholds.

4.2 Main Results

Held-out test split. The zero-shot pair is the same host and verifier without distillation. Heimdall warns before the accident on 30.4% of the hazardous clips against 20.4% for this pair at a false-alarm rate of 5.1% against 10.3% (Table 1). Its lead holds at every required margin up to nine seconds (Figure 3(a)). The zero-shot host sees the same frames but has never been told what they precede, so

Table 1: Held-out test split, in percent, $B = 10$. StreamSense is at its pool operating point, SPRINT and PaSBench at their native operating points, and all other rows at the joint operating point of Eq. (2). Gained and Lost count the clips each row gains and loses against the zero-shot pair at $\tau = 0$, and p is the exact sign-test p-value.

System	R0	R3	R5	FAR	Gained	Lost	p
StreamSense router, zero-shot pair	26.0	18.8	15.5	16.7	23	13	0.13
StreamSense router, distilled host	34.3	26.0	18.2	10.3	42	17	0.002
SPRINT probe, 8B every 0.5 s	**51.9**	**42.0**	**33.7**	21.8	62	5	<0.001
PaSBench-Video protocol, 8B every 0.3 s	7.2	5.5	5.0	11.5	5	29	<0.001
V-JEPA 2 probe + verifier, annotation-supervised	16.6	11.6	9.9	6.4	4	11	0.12
Zero-shot pair	20.4	13.3	11.0	10.3	–	–	–
Same-time teacher	22.7	15.5	12.2	6.4	9	5	0.42
Fixed-offset teacher	24.3	17.7	14.9	**5.1**	12	5	0.14
Heimdall	30.4	17.7	14.9	**5.1**	22	4	0.0005
Heimdall, distilled verifier	30.9	17.7	12.7	10.3	26	7	0.001
Annotation-supervised host (reference)	30.4	14.9	12.2	6.4	23	5	0.0009

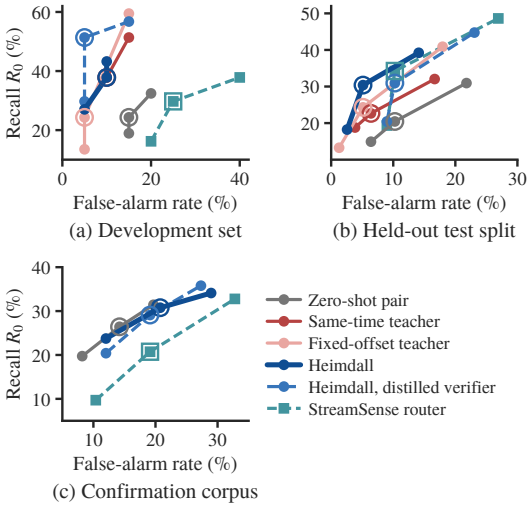


(a) Held-out test split (b) Joint calibration on the pool (c) Target quality

Figure 3: (a) Held-out test split at $B = 10$. (b) Pool clips alarmed against the per-path budget b, $\lambda^\star$ marked. (c) AUROC of the target and the same-time score by horizon.

it fires late or on the wrong cue. The distilled host has learned from the verifier's later verdicts which earlier cues precede an accident, and the joint calibration spends the same cap on fewer normal clips.

Published systems. With the distilled host, StreamSense's router reaches an R0 of 34.3% at a false-alarm rate of 10.3%, an eighth more hits than our policy for twice the false alarms. Its rules move a fixed decision point and every window can fire, so the cap binds it only loosely, and its operating point lies below the line joining our operating points at the same and the next budget (Figure 4(b)). SPRINT asks the 8B every half second and has no score to calibrate, so it reaches the highest R0 of 51.9% at a false-alarm rate of 21.8% and twenty times the verifier time of our loop. The PaSBench-Video prompt asks the same 8B for a concrete, actionable risk on a contact sheet, and the model rarely commits, so that protocol warns on at most 7.2% of the hazardous clips on any corpus.



(a) Development set (b) Held-out test split (c) Confirmation corpus

Figure 4: Recall at $\tau = 0$ against realized false-alarm rate at $B \in \{4, 10, 21\}$, ringed at $B = 10$, with the development thresholds kept unchanged on the confirmation corpus.

Development set and streaming memory. On the development set OASIS and HERMES both warn on 32.4% of the hazardous clips at zero margin. Heimdall reaches 37.8% there at no more false alarms, and both lead it at three seconds (Figure 5(a)). The zero-shot pair and Heim-

dall differ only in the host's weights, and distillation alone moves the operating point from an R0 of 24.3% at 15.0% false alarms to 37.8% at 10.0%. At the same budget none of ten streaming memories over three host configurations warns on more clips than its control. At the control's threshold no memory gains more than 10.8% of the hazardous clips, and every gain lies inside the ceiling of Corollary 1 (Figure 5(b)). A memory only shifts scores, whereas distillation changes what the host expects and so raises ρ on the urgent path. Score smoothing, clip statistics, per-domain thresholds, and logit fusion do not move the operating point either.

4.3 Ablations

Teacher target. The four hosts share training windows, LoRA budget, verifier, schedule, and calibration rule, and differ only in what the teacher was asked and when. Against the same-time teacher, Heimdall lifts R0 from 22.7% to 30.4%, R3 by a seventh, and R5 by a quarter, at one fewer false alarm. The same-time teacher can only copy the verifier's present judgment, which is uncertain before the accident, whereas a teacher asked later carries the certainty of the accident back to the seconds before it. The fixed-offset teacher is asked once four seconds ahead. It already carries the whole lead-time gain and trails Heimdall only at zero margin, so a single later verdict brings the earlier warnings, and the maximum over horizons adds the timely ones on precursors shorter than the offset. The host trained on human accident times reaches the same hits at zero margin and about a fifth fewer at three and five seconds, so the label-free target matches the annotation and leads on lead time.

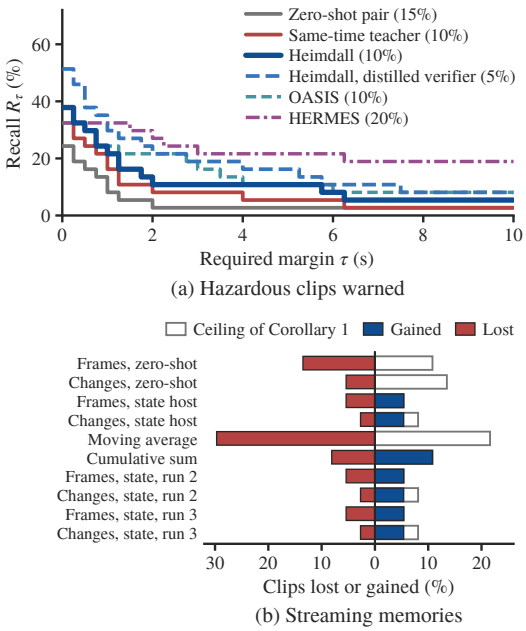


Figure 5: Development set. (a) Hazardous clips warned against the required lead, false-alarm rate in parentheses. (b) Ten streaming memories against the score-shift ceiling of Corollary 1. Frames and changes are frame histories fed to the verifier, the state host answers a scene-state question, and the moving average and cumulative sum transform the host score.

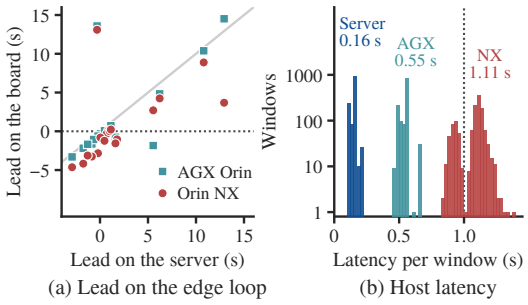
Target quality. The target ranks the annotated event better than the same-time score at every horizon beyond one second on the test and validation splits (Figure 3(c)). Its curves in Figure 2(b) sit below the diagonal because the maximum over later verdicts leads the event, which is why its thresholds are set on the pool instead of being read as probabilities.

Policy variants. Verdict write-back keeps a verdict in force and lowers the host threshold meanwhile. It spends the false-alarm budget twice and lowers hits at the same cap (Figure 3(b)), so it stays out of the method. Halving the verifier period instead raises hits by a quarter at the same false alarms, and every added clip is one the slower schedule warned on after the accident, so a denser schedule buys warnings that arrive in time. Applying the hindsight recipe to the verifier as well raises the pair's hits on the development set by a third at one fewer false alarm. On the test split it matches Heimdall's hits at twice the false alarms. Its thresholds are set on a pool that holds no driving footage, so it fires on the driving normals the pool never priced. On the confirmation corpus it falls slightly below Heimdall at the same false alarms. The verifier's gain therefore depends on the pool representing the site, and the host's gain does not.

Student architecture. A frozen V-JEPA 2 encoder with a small attention probe replaces the host in this ablation. It reaches little more than half of the host's hits for the hindsight and the annotated target, and the three targets tie for it. The probe cannot answer the teacher's question and can only fit the teacher's score, so a later verdict brings it no more than a present one, and neither a larger encoder nor the predictor's future tokens close the gap. The hindsight gain therefore needs a student that answers the same question as the teacher.

4.4 Confirmation Corpora and Edge Deployment

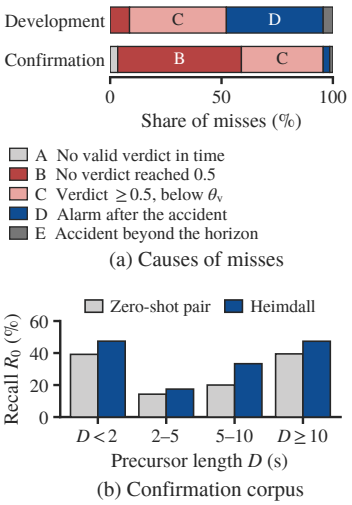
Edge and server closed loop. The loop runs the development clips in real time with the host on a Jetson and the verifier on a server at the joint thresholds of the main configuration. On the 64 GB AGX Orin the 16-bit host takes half a second per one-second window, so it keeps the grid, issues almost every verifier request, and keeps most of the server's hits at fewer false alarms. On the 16 GB Orin NX the four-bit host needs more than a second per window, so the schedule drifts and the zero-margin hits are lost to timing while the three-second hits remain (Figure 6). On the AGX the schedule matches the server, so the gap that remains comes from the inference stack, and on the clips both catch the board commits about a third of a second later. The 16-bit host draws about 18 W on the AGX during the loop, and the four-bit host on the NX spends about 15 J per window.



(a) Lead on the edge loop (b) Host latency

Figure 6: (a) Lead on the server and on two Jetson boards for the same clips and thresholds; points below the dotted line alarm after the accident. (b) Host latency per window, one-second grid dotted.

Transferred thresholds. On the confirmation corpus the development-pool thresholds are left unchanged. Heimdall keeps its lead over the zero-shot pair. It lifts R0 from 26.4% to 30.8% and R3 from 7.0% to 12.4%, and at three seconds it loses no clip that the zero-shot pair catches. The lead holds without the clips shared with the host's training data or with the test split. The zero-margin lead comes with a false-alarm rate of 20.8% against 14.2%. The pool holds footage from only one of these sources, so its thresholds sit too low for the others and the realized false-alarm rates run several times the cap. Recalibrating on the corpus's own normals in two folds restores the nominal rate, and the ordering holds at every budget (Figure 4). Most of the remaining misses are clips on which the verifier never scores above one half (Figure 7(a)). A lower verifier threshold recovers them only where every normal clip alarms, so only a more discriminating verifier can reach them. Most of the gain and of the extra false alarms come from the OOPS footage, and on the fall-detection footage neither policy warns at three seconds. Precursors are short on this corpus and on the development set. Their median is about two seconds and most are under five, so the footage itself bounds the lead that any warning can have (Figure 7(b)). On the self-collected site the development thresholds raise no alarm because the verifier scores this camera lower, and calibrating on the site's own normals moves the thresholds without changing how the clips are ranked.



(a) Causes of misses (b) Confirmation corpus

Figure 7: (a) Causes of the hazardous clips Heimdall misses at $\tau = 0$ on each corpus. (b) Confirmation clips warned at $\tau = 0$ by precursor length.

5 Conclusion

We have introduced Heimdall, a label-free method for proactive warning in streaming video. A frozen verifier that is uncertain when a warning is due is confident a few seconds later, so its later verdicts supervise the on-device host, and the two models run as one policy under one false-alarm cap. On a held-out test split, this host warns before more accidents at fewer false alarms than one distilled from the same verifier at the same moment, so the gain comes from asking the teacher later, and no inference-time mechanism adds to it. It matches a host trained on human accident times and exceeds an annotated V-JEPA 2 probe. The miss-rate identity explains this ordering, since a memory only shifts scores below a threshold the budget fixes, while supervision raises the host's discriminability where that threshold sits. The loop runs in real time on a Jetson at the same thresholds. Future work will calibrate per site and distill the verifier itself on the risk question it answers online.

USE OF LARGE LANGUAGE MODELS

Generative AI tools (Anthropic Claude, OpenAI Codex, and Moonshot Kimi) assisted with orchestrating experiment batches under protocols fixed by the authors, with writing and reviewing the analysis, calibration, and evaluation scripts, with an independent recomputation of the tables from the raw per-window logs, and with drafting and revising the manuscript. The icons in Figure 1 were generated with Google Gemini, and the text and routing of the figure are drawn by the authors. The authors reviewed all code and text and are responsible for the correctness and completeness of the paper.

ETHICS STATEMENT

This work concerns physical-risk warning and uses public human-activity footage together with a small self-collected set of controlled recordings. Video monitoring can expose private behavior, and missed hazards and false alarms fall unevenly on different people, so false alarms are reported alongside pre-accident hits, and the reported performance does not establish that a deployment is suitable for unattended safety-critical use. Dataset access and redistribution follow the conditions of the original providers, and no footage or identifying information is released to demonstrate an alarm. The self-collected recordings are scripted enactments performed by volunteers who gave informed consent to being recorded and to the use of the footage for this research. They are reported as a scene-specific evaluation and are not redistributed to anyone.

REPRODUCIBILITY STATEMENT

Section 3.1 fixes the task, the budget, and the joint calibration rule, Sections 3.2 and 3.3 specify the method and the online policy, and Appendix E documents the data split, the configurations, the real-time protocol, and the failed runs. Code and frozen thresholds will be released upon acceptance.

REFERENCES

Anastasios Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal Risk Control. In *International Conference on Learning Representations (ICLR)*, pp. 55198–55218, 2024a.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. doi: 10.1561/2200000101.

Anastasios Nikolas Angelopoulos, Rina Barber, and Stephen Bates. Online conformal prediction with decaying step sizes. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 1616–1630. PMLR, 2024b.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL technical report. arXiv preprint arXiv:2511.21631, 2025.

Wentao Bao, Qi Yu, and Yu Kong. DRIVE: Deep Reinforced Accident Anticipation with Visual Explanation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7599–7608, 2021.

Congqi Cao, Yue Lu, Peng Wang, and Yanning Zhang. A new comprehensive benchmark for semi-supervised video anomaly detection and anticipation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20392–20401, 2023.

Lorenzo Caselli, Tomaso Trinci, Tommaso Bianconcini, Simone Magistri, Leonardo Taccari, Francesco Sambo, and Andrew D. Bagdanov. FLaRA: Predicting future latent representations for accident anticipation. *arXiv preprint arXiv:2606.14380*, 2026.

Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu, Ziteng Gao, Dongxing Mao, and Mike Zheng Shou. VideoLLM-online: Online video large language model for streaming video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18407–18418, 2024a.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024b.

Yilong Chen, Xiang Bai, Zhibin Wang, Chengyu Bai, Yuhan Dai, and Ming Lu. StreamKV: Streaming Video Question-Answering with Segment-based KV Cache Retrieval and Compression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 3120–3128, 2026. doi: 10.1609/aaai.v40i4.37305.

Shangzhe Di, Zhelun Yu, Guanghao Zhang, Haoyuan Li, Tao Zhong, Hao Cheng, Bolin Li, Wanggui He, Fangxun Shu, and Hao Jiang. Streaming Video Question-Answering with In-context Video KV-Cache Retrieval. In *International Conference on Learning Representations (ICLR)*, pp. 42115–42127, 2025.

Dave Epstein, Boyuan Chen, and Carl Vondrick. Oops! predicting unintentional action in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 916–926, 2020.

José Camilo Eraso Guerrero, Elena Muñoz España, Mariela Muñoz Añasco, and Jesús Emilio Pinto Lopera. Dataset for human fall recognition in an uncontrolled environment. *Data in Brief*, 45:108610, 2022.

Cristóbal Eyzaguirre, Eric Tang, Shyamal Buch, Adrien Gaidon, Jiajun Wu, and Juan Carlos Niebles. Streaming Detection of Queried Event Start. In *Advances in Neural Information Processing Systems*, volume 37, pp. 100698–100733, 2024. doi: 10.52202/079017-3194.

António Farinhas, Chrysoula Zerva, Dennis Ulmer, and Andre Martins. Non-Exchangeable Conformal Risk Control. In *International Conference on Learning Representations (ICLR)*, pp. 50952–50966, 2024.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. OPTQ: Accurate Quantization for Generative Pre-trained Transformers. In *International Conference on Learning Representations (ICLR)*, 2023.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. Video-MME: The First-Ever Comprehensive Evaluation Benchmark of Multi-modal LLMs in Video Analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24108–24118, 2025.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Neha Gupta, Harikrishna Narasimhan, Wittawat Jitkrittum, Ankit Singh Rawat, Aditya Krishna Menon, and Sanjiv Kumar. Language model cascades: Token-level uncertainty and beyond. In *International Conference on Learning Representations (ICLR)*, 2024.

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)*, pp. 13504–13514, 2024.

Zhenpeng Huang, Xinhao Li, Jiaqi Li, Jing Wang, Xiangyu Zeng, Cheng Liang, Tao Wu, Xi Chen, Liang Li, and Limin Wang. Online Video Understanding: OVBench and VideoChat-Online. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3328–3338, 2025.

Md Mohaiminul Islam, Tushar Nagarajan, Huiyu Wang, Gedas Bertasius, and Lorenzo Torresani. BIMBA: Selective-Scan Compression for Long-Range Video Question Answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29096–29107, 2025.

Wittawat Jitkrittum, Neha Gupta, Aditya K. Menon, Harikrishna Narasimhan, Ankit Rawat, and Sanjiv Kumar. When does confidence-based cascade deferral suffice? In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Muhammad Monjurul Karim, Yu Li, Ruwen Qin, and Zhaozheng Yin. A Dynamic Spatial-Temporal Attention Network for Early Anticipation of Traffic Accidents. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):9590–9600, 2022.

Mengtian Li, Yu-Xiong Wang, and Deva Ramanan. Towards streaming perception. In *European Conference on Computer Vision (ECCV)*, 2020. arXiv:2005.10420.

Xiaozhe Li, Tianyi Lyu, Yang Li, Yichuan Ma, Peiji Li, Linyang Li, Qipeng Guo, Dahua Lin, and Kai Chen. What and when to distill: Selective hindsight distillation for multi-turn agents. *arXiv preprint arXiv:2605.19447*, 2026.

Zhijia Liang, Jiaming Li, Weikai Chen, Yanhao Zhang, Haonan Lu, and Guanbin Li. OASIS: On-Demand Hierarchical Event Memory for Streaming Video Reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2821–2831, 2026.

Xingcheng Liu, Bin Rao, Yanchen Guan, Chengyue Wang, Haicheng Liao, Jiaxun Zhang, Chengyu Lin, Meixin Zhu, and Zhenning Li. Predict and Resist: Long-Term Accident Anticipation Under Sensor Noise. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 782–790, 2026. doi: 10.1609/aaai.v40i1.37045.

Lorenzo Mur-Labadia, Matthew Muckley, Amir Bar, Mido Assran, Koustuv Sinha, Mike Rabbat, Yann LeCun, Nicolas Ballas, and Adrien Bardes. V-JEPA 2.1: Unlocking dense features in video self-supervised learning. *arXiv preprint arXiv:2603.14482*, 2026.

Junbo Niu, Yifei Li, Ziyang Miao, Chunjiang Ge, Yuanhang Zhou, Qihao He, Xiaoyi Dong, Haodong Duan, Shuangrui Ding, Rui Qian, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, and Jiaqi Wang. OVO-Bench: How Far is Your Video-LLMs from Real-World Online Video Understanding? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18902–18913, 2025.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to Route LLMs from Preference Data. In *International Conference on Learning Representations (ICLR)*, pp. 34433–34448, 2025.

Siddhant Panpatil, Arth Singh, Mijin Koo, Chaeyun Kim, Haon Park, and Dasol Choi. EgoSafety-Bench: A diagnostic egocentric video benchmark for evaluating embodied VLMs as runtime safety guards. *arXiv preprint arXiv:2607.00218*, 2026.

Patrik Patera, Yie-Tarng Chen, and Wen-Hsien Fang. Lightweight spatio-temporal modeling via temporally shifted distillation for real-time accident anticipation. In *International Conference on Learning Representations (ICLR)*, 2026.

Rui Qian, Shuangrui Ding, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. Dispider: Enabling video LLMs with active real-time interaction via disentangled perception, decision, and reaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24045–24055, 2025.

Jiawei Qiu, Yichen Xu, Jianzhe Ma, Mingyang Yu, Wenbin Zhu, Yang Han, Pinzheng Lv, and Wenxuan Wang. From sports to safety: Benchmarking proactive risk inference in MLLMs. *arXiv preprint arXiv:2608.05560*, 2026.

David Schneider, Zdravko Marinov, Moritz Mistol, Zeyun Zhong, Alexander Jaus, Rodi Düger, Rafael Baur, M. Saquib Sarfraz, and Rainer Stiefelhagen. OmniFall: From staged through synthetic to wild, a unified multi-domain dataset for robust fall detection. In *Computer Vision – ECCV 2026*, volume 17069 of *Lecture Notes in Computer Science*, pp. 112–129. Springer, 2026.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, Yan Lu, Jenq-Neng Hwang, and Gaoang Wang. MovieChat: From Dense Token to Sparse Memory for Long Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18221–18232, 2024.

Vinh Tran, Yang Wang, Zekun Zhang, and Minh Hoai. Knowledge distillation for human action anticipation. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 2518–2522, 2021.

Haibo Wang, Bo Feng, Zhengfeng Lai, Mingze Xu, Shiyu Li, Weifeng Ge, Afshin Dehghan, Meng Cao, and Ping Huang. StreamBridge: Turning your offline video large language model into a proactive streaming assistant. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 132332–132359, 2025.

Han Wang, Deyi Ji, Lanyun Zhu, Jiebo Luo, and Roy Ka-Wei Lee. StreamSense: Streaming social task detection with selective vision–language model routing. In *Proceedings of the ACM Web Conference 2026 (WWW)*, 2026a.

Xionghui Wang, Jian-Fang Hu, Jian-Huang Lai, Jianguo Zhang, and Wei-Shi Zheng. Progressive teacher-student learning for early action prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3551–3560, 2019.

Yuhang Wang, Lingyao Li, and Hao Zhou. Observe before you alert: Adaptive driver alerting with vision-language models. *arXiv preprint arXiv:2609.08130*, 2026b.

Yiweng Xie, Bo He, Junke Wang, Xiangyu Zheng, Ziyi Ye, and Zuxuan Wu. FluxMem: Adaptive Hierarchical Memory for Streaming Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31272–31282, 2026.

Zhenyu Yang, Kairui Zhang, Yuhang Hu, Bing Wang, Shengsheng Qian, Bin Wen, Fan Yang, Tingting Gao, Weiming Dong, and Changsheng Xu. LiveStar: Live streaming assistant for real-world online video understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 31266–31304, 2025.

Xiangyu Zeng, Kefan Qiu, Qingyu Zhang, Xinhao Li, Jing Wang, Jiaxin Li, Ziang Yan, Kun Tian, Meng Tian, Xinhai Zhao, Yi Wang, and Limin Wang. StreamForest: Efficient Online Video Understanding with Persistent Event Memory. In *Advances in Neural Information Processing Systems*, volume 38, pp. 75804–75835, 2025. doi: 10.52202/085713-2547.

Haoji Zhang, Yiqin Wang, Yansong Tang, Yong Liu, Jiashi Feng, and Xiaojie Jin. Flash-VStream: Efficient Real-Time Understanding for Long Video Streams. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21059–21069, 2025.

Haowei Zhang, Shudong Yang, Jinlan Fu, See-Kiong Ng, and Xipeng Qiu. HERMES: KV Cache as Hierarchical Memory for Efficient Streaming Video Understanding. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 8411–8430, 2026.

Zhong Zhang, Christopher Conly, and Vassilis Athitsos. Evaluating depth-based computer vision methods for fall detection under occlusions. In *Advances in Visual Computing (ISVC), Lecture Notes in Computer Science 8888*, pp. 196–207, 2014.

Peisen Zhao, Lingxi Xie, Jiajie Wang, Ya Zhang, and Qi Tian. Progressive privileged knowledge distillation for online action detection. *Pattern Recognition*, 129:108741, 2022.

Xinyi Zhao, Congjing Zhang, Pei Guo, Wei Li, Lin Chen, Chaoyue Zhao, and Shuai Huang. SmartHome-Bench: A comprehensive benchmark for video anomaly detection in smart homes using multi-modal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 3975–3985, 2025a.

Yu Zhao, Ying Zhang, Xuhui Sui, Baohang Zhou, Xinying Qian, Li Shen, and Dacheng Tao. From hindsight to foresight: Self-encouraged hindsight distillation for knowledge-based visual question answering. *arXiv preprint arXiv:2511.11132*, 2025b.

Yusong Zhao, Yuejin Xie, Youliang Yuan, Junjie Hu, Jitian Guo, Yujiu Yang, and Pinjia He. PaSBench-Video: A streaming video benchmark for proactive safety warning. *arXiv preprint arXiv:2606.02443*, 2026a.

Zicheng Zhao, Kangyu Wang, Shijie Li, Rui Qian, Weiyao Lin, and Huabin Liu. CogStream: Context-guided Streaming Video Question Answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 13332–13341, 2026b. doi: 10.1609/aaai.v40i16.38336.

Yiyang Zou, Tianhao Zhao, Peilun Xiao, Hongyu Jin, Longyu Qi, Yuxuan Li, Liyin Liang, Yifeng Qian, Chunbo Lai, Yutian Lin, Zhihui Li, and Yu Wu. RiskProp: Collision-anchored self-supervised risk propagation for early accident anticipation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2768–2777, 2026.

A PROOFS

Proposition 1. Under Assumption 1 the M_τ window scores of a hazardous clip available before $a_i - \tau$ are independent, k_τ of them drawn from F_1 and $M_\tau - k_\tau$ from F_0. A miss at margin τ is the event that all of them fall below θ, so $\Pr[\text{miss}_\tau] = F_0(\theta)^{M_\tau - k_\tau} F_1(\theta)^{k_\tau}$. Writing $F_0(\theta)^{M_\tau - k_\tau} = \big(F_0(\theta)^{M_0}\big)^{M_\tau/M_0} F_0(\theta)^{-k_\tau}$ and substituting $F_0(\theta_\alpha)^{M_0} = 1 - \alpha$ gives $\Pr[\text{miss}_\tau] = (1-\alpha)^{M_\tau/M_0}\big(F_1(\theta_\alpha)/F_0(\theta_\alpha)\big)^{k_\tau}$, which is Eq. (5). Non-negativity of ρ requires $F_1(\theta_\alpha) \le F_0(\theta_\alpha)$, that is, signal windows stochastically larger at that threshold. When M_τ, k_τ, or the window count are random, the identity holds conditionally and the unconditional miss rate is its expectation. Scores are quantized, so a threshold attaining α exactly need not exist and the pool rule of Eq. (1) attains the largest atom below it. □

Corollary 1. The mechanism leaves F_0 unchanged, hence leaves θ_α unchanged. A clip alarms after the shift and not before only if $\max_j(s_j + \delta_j) \ge \theta_\alpha > \max_j s_j$, which requires $\max_j s_j \ge \theta_\alpha - \delta$ since $\delta_j \le \delta$. The condition is necessary and not sufficient, since the realized increment may fall short of the gap. No clip loses an alarm because the shift is non-negative. Therefore the change in recall is at most the probability mass of $\max_j s_j$ in $[\theta_\alpha - \delta, \theta_\alpha)$. □

Running maximum under one latent risk. Suppose every window score of a clip is an increasing function $s_j = g_j(z)$ of one latent risk z whose likelihood ratio between hazardous and normal clips is monotone. Then the running maximum attains the receiver operating characteristic of the optimal test, and no other statistic of the scores improves on it. To see this, note that the vector $(s_1, \dots, s_M)$ is a measurable bijection of z on the range of the g_j, so every statistic $f(s_1, \dots, s_M)$ is a function of z. By the Neyman-Pearson lemma the optimal test for z thresholds the likelihood ratio of z, which is monotone in z by the monotone likelihood ratio condition, and $\max_j s_j = \max_j g_j(z)$ is itself an increasing function of z. Hence the maximum attains the same receiver operating characteristic as the optimal test and no statistic improves on it. □

Proposition 2. Conditional independence gives $p(u, v \mid c) = p(u \mid c)p(v \mid c)$ for $c \in \{\text{signal}, \text{normal}\}$, so $\lambda_{uv} = \log \frac{p(u,v|\text{signal})}{p(u,v|\text{normal})} = \lambda_u + \lambda_v$. Thresholding λ_{uv} is optimal by Neyman-Pearson. If each score is a calibrated logit of the same binary question under a logistic model with common scale σ, then $\lambda_u = (u - u_0)/\sigma$ and $\lambda_v = (v - v_0)/\sigma$, so $\lambda_{uv} = (u + v - u_0 - v_0)/\sigma$ is an increasing affine function of $u + v$ and the equal-weight sum induces the same test. Strictness fails only when one score is uninformative given the other. □

Cadence. Write $r = M_\tau/M_0$ and $Q = k_\tau \rho(\theta_\alpha)$, so $\log \Pr[\text{miss}_\tau] = r \log(1-\alpha) - Q$. A denser grid changes r, Q, and θ_α together, and recall at a fixed margin increases exactly when $\Delta Q > \Delta r \, \log(1-\alpha)$. Only when r is unchanged does this reduce to $\Delta Q > 0$. □

B STREAMING MEMORY AND CLIP STATISTICS UNDER THE BOUND

The two analyses below test Corollary 1 and the running-maximum result of Appendix A.

Discriminability at the operating point. Figure 2(a) plots ρ over the development windows. At the frozen verifier's operating threshold, 0.9994 or 7.5 logits, $F_0 = 0.990$ and $F_1 = 0.920$, so $\rho = 0.073$ from 3 normal and 17 signal windows in the tail, against 0.29 at 2 logits; with $k_\tau = 1$ one piece of evidence at this threshold barely moves the miss rate off the budget term, which is what the verifier-only row of Table 12 shows.

Shift ceiling. Under the verifier statistic at its frozen threshold, 27 of 37 hazardous clips are missed with a median gap of 3.0 logits to the threshold, two streaming memories move nine signal windows in ten by less than two logits, and their measured effect is no clip gained and two and five lost, as for ten memory arms in all (Table 2).

Table 2 extends the test to ten arms. Frame history feeds the verifier eight uniformly spaced frames from the last sixteen seconds, change-triggered history feeds it eight frames chosen at visual changes, and the moving average and the cumulative sum transform the host's own score stream. The zero-shot host answers the risk question, the state-question host answers a scene-state question, and runs 2 and 3 repeat the latter's two memories in two further batches. In all ten the clips a memory arm gains lie inside the ceiling, and in six of ten the arm loses at least as many as it gains.

Table 2: Score-shift ceiling and measured effect of ten memory arms at the frozen threshold of the same host without the memory, in percent of the 37 hazardous clips.

Memory	Host	Ceiling	Gained	Lost
Frame history to the verifier	zero-shot host	10.8	0.0	13.5
Change-triggered history to the verifier	zero-shot host	13.5	0.0	5.4
Frame history to the verifier	state-question host	5.4	5.4	5.4
Change-triggered history to the verifier	state-question host	8.1	5.4	2.7
Moving average of the host score	state-question host, alone	21.6	0.0	29.7
Cumulative sum of the host score	state-question host, alone	10.8	10.8	8.1
Frame history to the verifier	state-question host, run 2	5.4	5.4	5.4
Change-triggered history to the verifier	state-question host, run 2	8.1	5.4	2.7
Frame history to the verifier	state-question host, run 3	5.4	5.4	5.4
Change-triggered history to the verifier	state-question host, run 3	8.1	5.4	2.7

Figure 5(b) plots the table at the control's frozen threshold, so Gained and Lost compare clip maxima under one threshold, while the budget-matched counts below use each arm's own frozen threshold and causal commit times. The two score-level arms are the cleanest case. They consume a host score stream identical to their control at all 1253 paired windows and change the clip statistic on every one of them. Budget-matched hits fall from 9 to 6 for both, the tail ratio falls from 4.04 to 2.06 and 1.77, and ρ falls from 0.056 to 0.041 and 0.038, so aggregating a fixed score stream lowered its discriminability. Sweeping the threshold of every arm over the same clips, each memory arm sits at or below its own control at every budget up to 4 false alarms of 20 and overtakes it only above a 35 percent clip-level false-alarm rate, seven to eleven times the operating budget. Streaming memory therefore helps only far above the false-alarm rates a deployment would accept.

Clip-level statistics. Table 3 compares clip-level statistics on the development set under the causal online rule, where the statistic is recomputed after every window and the clip alarms at the first window that crosses the threshold. Thresholds are set on the development normals at three budgets, and hazard counts are out of the 36 clips that have at least one pre-accident window. The e-process row multiplies per-window e-values $e = \kappa p^{\kappa-1}$ with $\kappa = 0.5$ from pool p-values. Dependent windows do not make the product a valid e-process, so the row is an aggregation heuristic. No statistic has a consistent advantage. The mean of the last two windows is above the maximum at $B = 1$ and $B = 3$ and below it at $B = 2$, and every other statistic is at or below the maximum at every budget. A non-causal mean would reward clip length instead of hazard, since normal clips carry more windows, so the causal form is used.

Table 3: Clip-level statistics at three false-alarm budgets, share of the 36 hazardous clips with a pre-accident window that are hit, in percent.

Clip statistic	$B = 1$	$B = 2$	$B = 3$
Running maximum (used by all systems)	11.1	25.0	27.8
Running mean	8.3	11.1	25.0
Exponential moving average, $\rho = 0.5$	11.1	16.7	25.0
Mean of the last two windows	16.7	16.7	36.1
Count of windows above a fixed level	2.8	13.9	27.8
E-process with a valid calibrator	2.8	11.1	25.0
Fisher combination	0.0	2.8	8.3

C Ablations and Run-to-Run Variation

Counts in this section are at the per-stream thresholds of Eq. (1) unless a joint operating point is named, and the joint numbers of every arm are in Table 14.

Same-time teacher. The control that isolates the value of the future keeps the training windows, the host, and the LoRA budget of Heimdall and asks the teacher at t instead of over $(t, t + k]$. The windows are the ones selected by the hindsight rule, so the control tests a change of target on a fixed sample. In an offline replay on the test split with thresholds set on a held-out normal fold at a 20 percent budget, the same-time student is below the hindsight student on every source, for instance 0.41 against 0.88 pre-accident recall on driving footage and 0.07 against 0.50 on the fall corpus at matched false-alarm rates. Under the live protocol of Section 4 the same-time student reaches 14, 3, and 2 hits at two false alarms at its joint operating point against 14, 4, and 4 for Heimdall, tied at zero margin with four clips each way and behind by two clips to one at three seconds. The test split of Table 1 separates the two more clearly.

What the target measures. On the 259-video test split and the validation split, where accident times are annotated, we compare the target $R_k(t)$ of Eq. (3) and the same-time score $q(t)$ against the event Y_t^k on 3379 hazardous-clip windows selected by the training rule (Table 4). R_k ranks Y_t^k better than the same-time score at every horizon beyond one second, and the margin grows with the horizon, which is the intended effect of looking ahead. R_k is not calibrated. In its top bucket the empirical frequency of Y_t^k is 9 to 16 percent, because the proxy fires on windows whose accident falls just beyond the horizon, 5 to 9 percent of windows have $R_k \geq 0.5$ with $Y_t^k = 0$, and 10 to 14 percent of those are the counterexample of Section 3.2 with the accident in $(t + k, t + k + 3]$. On normal clips $R_k \geq 0.5$ on 0.6 to 2.3 percent of the 3569 windows. On hazardous-clip windows that precede the annotated onset, $R_k \geq 0.5$ on 1.8 percent at $k = 1$ and 10.0 percent at $k = 5$ against 1.6 percent for q, so the longer horizons do label some pre-onset windows as risky, which is either an earlier cue than the annotation or a scene prior. Appendix G examines the alarms this produces.

Table 4: Target quality on the test split: AUROC of R_k and of the same-time score q against Y_t^k, and shares of windows with $R_k \geq 0.5$ but no event.

k (s)	positives	AUROC R_k	AUROC q	early (%)	normal ≥ 0.5 (%)
1	91	0.660	0.667	5.0	0.6
2	192	0.678	0.644	6.8	1.0
3	302	0.687	0.622	8.2	1.5
5	538	0.687	0.606	9.4	2.3

Student architecture. Table 5 replaces the host by a frozen V-JEPA 2 ViT-L/16 encoder (Assran et al., 2025) with an attention probe, one cross-attention block with a learned query and sixteen heads followed by an MLP and five logits, 6.3M parameters. The probe sees the host's window, the same decision time and span and only frames up to t, sampled uniformly to sixteen frames at 256^2, which is denser than the host's eight frames. The hindsight and same-time probes are trained on the host's 3,573 training windows with the cached teacher targets of Eq. (3) and of the same-time control. The annotated probe is the baseline form: it is trained on all 24,152 one-second-grid windows of the 412 training videos with the annotated accident time turned into a hard bucket label (15,605 windows with no accident within five seconds and 6,812 within one second), so that no teacher score enters its training set. Training uses AdamW with soft or hard cross-entropy, early stopping on the validation

split, and three seeds, with seed 0 as the audited arm. A fourth probe is a FLaRA-style variant (Caselli et al., 2026) that adds the frozen predictor's future tokens to the encoder tokens, and two upper-bound probes use the ViT-g/384 encoder (1B parameters, 318 ms per window on the server) with the hindsight and the annotated target, one seed each. Probe rows are scored offline on the same grid and their delivery time is the decision time plus the measured encoder and probe latency, per window for the soft-target probes and its serial median for the annotated probe, 47 to 54 ms, which is about 0.1 s earlier than the live host's 0.14 s and favors the probe at the margins; the verifier rows of the recorded captures are reused unchanged, so Eq. (2) runs on the probe's own pool exactly as for the host. On the 64 GB AGX Orin the encoder and probe take 0.24 s per window in 16-bit at batch one (0.21 s for the encoder), 771 MB of memory and 13 W on average, against 0.55 s for the 16-bit host on the same board, so the probe is the cheaper student by a factor of two on this hardware; its probabilities differ from the server's by up to 0.03 for the hindsight probe and 0.24 for the annotated one under 16-bit arithmetic. At $B = 10$ on the test split the seed means are 31, 31, 34, and 30 hits at 4 to 6 false alarms for the hindsight, same-time, annotated, and predictor probes, with ranges of three to six clips across seeds, against 55 for Heimdall. The ViT-g probes reach 36 and 27 hits for the hindsight and annotated targets, and the annotated probe with predictor tokens 33 at 9 false alarms, so a larger encoder or the predictor's future tokens do not close the gap. Alone, the audited probes reach 8 to 12 hits at $\tau = 0$ and 0 to 7 at three seconds under Eq. (1) (8 to 18 and 0 to 10 across all seeds), so most of the paired hits come from the shared verifier path. On the 482-clip corpus the annotated probe is the one exception: its seeds reach 69, 101, and 133 hits at 18 to 37 false alarms against 92 at 38 for Heimdall, a spread that points to the hard label and the domain mix, and at three seconds its seeds give 30, 26, and 47 against 37. Paired on the test split, the audited annotated probe gains 2 and loses 27 clips against Heimdall at $\tau = 0$ and gains none at three seconds. Ranked against an accident within three seconds over the 6,018 test windows, the hindsight probe scores an AUROC of 0.66 against 0.61 for the same-time probe and 0.63 for the annotated one, so the hindsight target does improve the probe's ranking, but every probe stays below every host (0.60 to 0.77), and the annotated host ranks best of all while tying Heimdall in hits and trailing it at three and five seconds, so ranking over all windows and timely hits under a cap are different quantities.

Table 5: Student architecture at the joint operating point with $B = 10$, audited seed, in percent. Hosts are the rows of Table 1; probes are frozen V-JEPA 2 ViT-L with an attention probe, scored offline. The last column is the AUROC of the student's risk score against an accident within three seconds over the 6,018 test windows.

		Development set				Test split				
Student	Target	R0	R3	R5	FAR	R0	R3	R5	FAR	AUROC_3
Qwen3-VL-2B host	annotation	32.4	10.8	8.1	5.0	30.4	14.9	12.2	6.4	0.77
Qwen3-VL-2B host	hindsight $\hat{p}(t)$	37.8	10.8	10.8	10.0	30.4	17.7	14.9	5.1	0.71
Qwen3-VL-2B host	same-time $q(t)$	37.8	8.1	5.4	10.0	22.7	15.5	12.2	6.4	0.67
V-JEPA 2 probe	annotation	13.5	5.4	5.4	10.0	16.6	11.6	9.9	6.4	0.63
V-JEPA 2 probe	hindsight $\hat{p}(t)$	13.5	5.4	5.4	5.0	17.7	12.7	11.0	6.4	0.66
V-JEPA 2 probe	same-time $q(t)$	8.1	2.7	2.7	5.0	16.0	12.7	10.5	5.1	0.61
V-JEPA 2 probe + predictor	hindsight $\hat{p}(t)$	10.8	2.7	2.7	5.0	16.0	12.2	10.5	6.4	0.66
V-JEPA 2 probe + predictor	annotation	13.5	5.4	5.4	10.0	18.2	12.2	10.5	11.5	0.64
V-JEPA 2 ViT-g/384 probe	annotation	8.1	2.7	2.7	10.0	14.9	11.6	9.9	6.4	0.63
V-JEPA 2 ViT-g/384 probe	hindsight $\hat{p}(t)$	16.2	5.4	2.7	10.0	19.9	13.3	11.6	3.8	0.68

Teacher horizon. A host distilled from a teacher queried at a fixed four seconds ahead reaches 16, 6, and 4 hits at one false alarm, against 16, 5, and 4 at two for the maximum over horizons of Eq. (3). At per-stream thresholds the two designs are equivalent end to end. At the joint operating point they are not: the fixed-offset student reaches 9 hits at one false alarm ($\lambda^\star = 6$) against 14 at two for the maximum over horizons, because at the joint thresholds its host path ranks the pool normals above more hazardous clips than the multi-horizon host does. The absolute scale of its scores is irrelevant, since Eq. (1) is invariant to monotone rescaling.

Decision cadence. Halving the verifier period raises hits from 16 to 20 at the same two false alarms and verifier calls from 635 to 1091, with the recalibrated verifier threshold moving only from 0.9994 to 0.9993. All four extra clips are ones the two-second arm alarmed on after the accident, and the

median lead over the hit set is unchanged at 1.37 s, so the denser grid buys warnings that arrive in time and not warnings that arrive earlier, in the direction of the cadence remark of Appendix A.

Annotation supervision. The annotation-supervised host of Table 11 is trained on human accident times in place of teacher verdicts, on different training data, and serves as a reference point. At per-stream thresholds it reaches 19, 7, and 5 hits at one false alarm. At its joint operating point ($\lambda^\star = 5$) it reaches 12, 4, and 3 at one, since its host calibrates at 0.88 and loses recall faster than the distilled host as the per-path budget tightens. On the test split at $B = 10$ it reaches 55, 27, and 22 hits at 5 false alarms against 55, 32, and 27 at 4 for Heimdall, and paired at $\tau = 0$ the two gain and lose 11 clips each, so they tie at zero margin and differ at the longer margins.

Run-to-run variation. Repeating the distilled system end to end gives 16, 6, and 5 hits at two false alarms against 16, 5, and 4. Paired clip by clip, the two runs agree on every clip at $\tau = 0$ and among the false alarms, and differ on one clip each at three and five seconds, so a one-clip difference at the longer margins is run-to-run variation. The three clips that write-back gains at per-stream thresholds are discordant in one direction and are therefore not decode jitter. Against the zero-shot pair, Heimdall gains eight clips and loses one at $\tau = 0$ and gains six and loses none at three seconds. The distilled verifier against the frozen one gains five and loses none at $\tau = 0$ and six and none at three seconds, and on the 482-clip corpus it gains seven clips and loses none while adding eight false alarms and removing none.

D Cross-Model Score Fusion

Proposition 2 (Cross-model additivity)**.** Let a window carry two scores u and v from different models, conditionally independent given the class. Then the log-likelihood ratio of the pair is $\lambda_{uv} = \lambda_u + \lambda_v$ and thresholding it is uniformly most powerful. If both are calibrated logits under a logistic model with equal scale, the equal-weight sum $u + v$ induces the same test, and the improvement over the better single score is strict unless one is conditionally uninformative given the other.

The proposition assumes conditional independence, and the two scores of our system violate it.

The fusion rule mentioned in Section 4.2 was frozen before evaluation. The window statistic is the sum of the two logits at unit weight, the host contribution is the last score available at or before the moment the verifier query was issued, the clip statistic is the running maximum, and thresholds come from the same audited calibration code. Within each class the two window logits correlate at 0.61 and 0.53 on the development set and 0.78 and 0.69 on the confirmation set, and a permutation test rejects independence. At a matched five percent false-positive rate the sum's tail ratio is 3.99 against the verifier's 3.75 on the development set and 3.07 against 2.99 on the confirmation set, while the host score alone reaches 5.27. Neither column of Table 6 includes the host's urgent path. Entries are percentages of the 37 or 299 hazardous clips and of the 20 or 183 normal clips, here and in Table 7, and FAR is the rate realized on the evaluation normals.

Table 6: Sum of logits in place of the verifier score at three budgets on the shared 214-clip pool, in percent of the hazardous and normal clips.

		Verifier score alone				Sum of the two logits			
Corpus	B	R0	R3	R5	FAR	R0	R3	R5	FAR
Development	4	8.1	2.7	2.7	5.0	10.8	5.4	2.7	5.0
	10	16.2	2.7	2.7	5.0	13.5	5.4	2.7	5.0
	21	32.4	18.9	10.8	20.0	48.6	21.6	16.2	20.0
Confirmation	4	12.7	6.7	5.7	4.9	17.1	7.0	5.7	8.7
	10	17.1	7.7	6.0	9.3	19.7	8.4	6.7	12.0
	21	22.1	10.0	7.7	20.2	24.1	10.7	8.4	22.4

Timing. On the confirmation corpus, the sum never commits later than the verifier alone on clips both detect. It is earlier on three of 51 shared clips by up to 2.12 s, identical on the rest, and it adds eight clips while losing none. The development set loses one clip. A third arm with a coarsely quantized host behaves differently, committing a median of 1.06 s later and losing two clips, because its few high host scores raise the fused threshold more than its typical scores raise the statistic.

Weight sensitivity. Table 7 varies the weight after the fact. Doubling the host weight raises development hits to 8 at one false alarm, which is better than either frozen rule, and raises confirmation false alarms from 17 to 28. Taking the minimum of the two logits reaches 12 hits on the development set and 71 on the confirmation set at 40 false alarms. No setting improves both corpora, and none of these variants is used elsewhere in the paper.

Table 7: Weight variants at the main budget, in percent of the hazardous and normal clips.

	Development		Confirmation	
Window statistic	R0	FAR	R0	FAR
Verifier score alone	16.2	5.0	17.1	9.3
$s_{\mathrm{v}} + s_{\mathrm{h}}$, the frozen rule	13.5	5.0	19.7	12.0
$s_{\mathrm{v}} + 0.25\, s_{\mathrm{h}}$	16.2	5.0	18.4	10.4
$s_{\mathrm{v}} + 0.5\, s_{\mathrm{h}}$	16.2	5.0	20.1	12.0
$s_{\mathrm{v}} + 2\, s_{\mathrm{h}}$	21.6	5.0	22.4	15.3
$\max(s_{\mathrm{v}}, s_{\mathrm{h}})$	16.2	5.0	17.1	9.3
$\min(s_{\mathrm{v}}, s_{\mathrm{h}})$	32.4	10.0	23.7	21.9

E Protocol and Reproduction

Verdict write-back. A verdict can also be kept in force for T_v seconds after arrival, in a set $\mathcal{V}$ that both handlers of Algorithm 1 read after dropping expired entries. A positive verdict with $s_{\mathrm{v}} \geq \gamma_{+}$ lowers the urgent threshold of line 4 to $\theta_{\mathrm{h}} - \delta_{\mathrm{u}}$ while it is in force, and a negative verdict with $s_{\mathrm{v}} < \gamma_{-}$ suppresses the advisory path of line 8. Write-back lets one answer shape the decisions of the next few seconds at no extra call, but it also spends the false-alarm budget more than once: at the same ten pool clips its joint operating point falls from $\lambda^{\star} = 6$ to 3, and it reaches 12 hits against 14 on the development set and 83 against 92 on the confirmation corpus. It is therefore reported as a variant and kept out of the method.

Training windows and the sampled target. A training window is kept when $R_{k_4} > 0.2$, together with twice as many other windows drawn at random. The training windows are thus selected with the teacher score, so the student learns the average target over the selected windows, $\mathbb{E}[\hat{p}(t) \mid x_{\leq t}, I = 1]$ with I the selection indicator, a risk score under the sampled distribution whose threshold is set afterwards on the pool. R_k itself is not $\Pr[Y_t^k]$. A teacher that is exactly right at $s = t + k$ about an accident at $t + k + 0.5$ makes $R_k(t)$ large while $Y_t^k = 0$, and the largest of several probabilities is not the probability that at least one of the events occurs. The trained LoRA adapter is merged into the host for serving, so the served host has the cost of the base model.

Real-time protocol. Every batch ran in real time on a GPU held for it, with the model server's image and prefix caches disabled and the container identity and package versions recorded with the outputs. A run writes one row per decision window with the frame hashes, the issue and arrival instants, the returned probability, and the validity flags, and a ledger of every request including those answered after the horizon. Thresholds are computed by Eq. (1) from a pool run under the same configuration and written to disk with the hash of the pool outputs before any evaluation run reads them. Failed runs are kept. A pool computed with the host in 16-bit floating point produced three non-finite verifier scores and was recomputed in bfloat16, a driving anticipation baseline had no feasible threshold at any budget and is reported as such, and the cross-model fusion rule appears only in Appendix D.

Data split and exposure. Table 8 lists every corpus with its role and Table 9 what each had been used for before this protocol. Training windows come from the training split of PaSBench-Video, split at the video level. The development set is a subset of its validation split. The 214-clip pool comes from two external normal sources, the NWPU Campus test normals (Cao et al., 2023) and the SmartHome-Bench normals (Zhao et al., 2025a), and is disjoint from training, development, and confirmation. The 482-clip confirmation corpus is drawn from OOPS, SmartHome-Bench, PaSBench-Video, EDF, OCCU, OmniFall, and CAUCAFall (Epstein et al., 2020; Zhao et al., 2025a; 2026a; Zhang et al., 2014; Schneider et al., 2026; Eraso Guerrero et al., 2022). Its 54 PaSBench-Video clips are all hazardous and also belong to our test split. Of its clips, 19 are also in the host's training manifest and 309 have a prior experimental record in our archive, and no pool clip is among them.

Teacher windows and student windows of a training clip are in the same partition by construction. In Table 9, training manifest refers to the host's training windows and prior record means any earlier scoring or experiment in our archive.

Table 8: Corpora and their roles, with sizes in clips unless stated.

Corpus	Source	Size	Role
Training	PaSBench-Video, our train split	364 videos, 3573 windows	host distillation
Validation	PaSBench-Video, our val split	69 videos	early stopping
Development	57 validation videos	37 hazardous, 20 normal	method choice, Table 11
Test	PaSBench-Video, our test split	181 hazardous, 78 normal	Table 1
Pool	two external normal sources	$118 + 96$	calibration only
Driving pool	driving normals	97	driving thresholds
Confirmation	seven sources	299 hazardous, 183 normal	transfer, Table 15
Self-collected	one site, scripted enactments	35 hazardous, $15 + 15$ normal	site shift

Table 9: Exposure of each evaluation corpus before this protocol.

Corpus	In training manifest	Prior record	Used to choose the method
Development	none	yes	yes
Test	none	scored offline once	no
Pool	none	none	no
Confirmation	19 clips (1 hazardous)	309 clips	no
Self-collected	none	none	no

Configurations. Table 10 fixes what each row name of Tables 1 and 11 denotes. All two-model rows use the online policy of Algorithm 1 with $P = 2\,\text{s}$, $T_a = 3\,\text{s}$, budgets $B \in \{4, 10, 21\}$ on the 214-clip pool, and thresholds from Eq. (2), the host serves the five-bucket time-to-accident question, and the verifier serves the risk question. The host's five-way question has the buckets no accident within 5 s, 3 to 5 s, 2 to 3 s, 1 to 2 s, and within 1 s, so the horizons of Eq. (3) are $k \in \{1, 2, 3, 5\}$ s.

Table 10: Configurations behind the rows of Tables 1 and 11 (U urgent, A advisory, WB write-back).

Row	Host weights	Teacher and target	Verifier	Paths
Zero-shot pair	Qwen3-VL-2B	none	frozen 8B	U+A
Heimdall	LoRA, Eq. (3)	8B five-bucket, max over $(t, t+k]$	frozen 8B	U+A
+ write-back	same	same	frozen 8B	U+A+WB
Same-time teacher	LoRA	8B five-bucket at t	frozen 8B	U+A
Fixed-offset teacher	LoRA	8B five-bucket at $t + 4\,\text{s}$	frozen 8B	U+A
Annotation-supervised	LoRA	human accident times	frozen 8B	U+A
Heimdall, distilled verifier	LoRA, Eq. (3)	host as above; verifier from 8B binary state at $t + 4\,\text{s}$ on five frames	distilled 8B	U+A

The online policy and replay. The deployed warner runs the two event handlers of Algorithm 1 with the constants of Section 4, and no handler waits on the verifier. Every run logs in shadow mode. The host scores every window up to H_i and the verifier is asked at every grid point whether or not an alarm has been committed, so the recorded rows of a run do not depend on its thresholds. An alarm is committed within two milliseconds of the event that triggers it, and using the wall-clock instant in place of the logical instant changes no hit at any margin. Replay is therefore exact. The per-stream thresholds of every arm were frozen live before any score was read. The joint operating point of Eq. (2) is computed from the recorded pool rows. For each common per-path budget λ the two thresholds are set by Eq. (1), the full policy is replayed over the recorded host and verifier rows of every pool clip, and the largest λ whose replayed false-alarm count is at most B is kept. Evaluation rows are then replayed under the joint thresholds, and the commit time of each replayed alarm is the recorded availability or arrival instant of the row that triggers it.

The distilled verifier. The verifier is distilled with a binary hazard-state prompt from its own verdicts four seconds later and served with the risk question every other configuration uses. On the pool it takes 123 distinct clip values against 78 for the frozen verifier. On 479 held-out signal windows against 728 normal windows, scored under the training prompt at matched false-positive rates, its

ρ rises from 0.23 to 0.32 at 5 percent, from 0.24 to 0.46 at 10 percent, and from 0.48 to 0.67 at 20 percent. On the test split at $B = 10$ it reaches 56, 32, and 23 hits at 8 false alarms against 55, 32, and 27 at 4 for the frozen-verifier pair, and paired on the same clips it gains 8 and loses 7 at $\tau = 0$. Its false alarms fall on 7 of the 32 driving normals against 3, and its hits move toward driving, 23 against 18, and away from daily life, 25 against 32. The pool holds no driving normals, so its cap does not bind on the footage where the distilled verifier fires most readily. On the confirmation corpus it reaches 87, 34, and 20 hits at 35 false alarms against 92, 37, and 23 at 38 for the frozen-verifier pair, and paired at $\tau = 0$ it gains no clip and loses 5.

Reproduction. Every threshold and table row was recomputed from the raw per-window logs by analysis passes that share no code with the runners, and every threshold and count reproduces exactly. Figure 2(a) is regenerated from the frozen development capture.

Dependencies and cost. Hosts and verifiers are served with vLLM in bfloat16, prefix and image caches disabled. The edge closed loop uses a four-bit GGUF host under llama.cpp (Appendix H); a GPTQ (Frantar et al., 2023) four-bit copy served only the server-side equivalence check. Distillation uses LoRA rank 8, one epoch, batch size one with gradient accumulation eight, and learning rate 10^{-4}, and the 8B verifier distillation takes two hours on one DGX Spark. In real time, a development-set arm of 57 clips takes about 23 minutes, a 214-clip calibration arm about 110, and a 482-clip confirmation arm about 95. Code and frozen thresholds will be open-sourced upon acceptance.

F Additional Results on the Development Set

Table 11: Development set, in percent of the 37 hazardous clips (Rτ) and of the 20 normal clips (FAR). Two-model rows are at the joint operating point of Eq. (2) with $B = 10$, and single-stream rows use Eq. (1). Bold marks the best row outside the annotation reference.

System	R0 (%)	R3 (%)	R5 (%)	FAR (%)
Published baselines				
HERMES memory-free control, 3B	18.9	10.8	8.1	10.0
OASIS, event memory	32.4	16.2	10.8	10.0
HERMES, released configuration	32.4	21.6	**21.6**	20.0
SPRINT probe, native operating point	48.6	24.3	21.6	30.0
PaSBench-Video protocol, native operating point	5.4	5.4	2.7	0.0
StreamSense router, zero-shot pair	16.2	13.5	10.8	30.0
StreamSense router, distilled host	29.7	16.2	10.8	25.0
Heimdall and its ablations, joint operating point Eq. (2)				
Zero-shot pair	24.3	2.7	2.7	15.0
Same-time teacher	37.8	8.1	5.4	10.0
Fixed-offset teacher	24.3	8.1	8.1	5.0
Heimdall	37.8	10.8	10.8	10.0
+ verdict write-back	32.4	10.8	8.1	10.0
Heimdall, distilled verifier ‡	**51.4**	18.9	16.2	5.0
Annotation-supervised host (reference)	32.4	10.8	8.1	5.0

‡ arm frozen last, with its test-split row in Table 1.

StreamSense under the pool protocol. StreamSense's router (Wang et al., 2026a) decides at each step whether to invoke the large model, skip it on the small model's score, or defer, with confidence thresholds that move with the time since the last decision. Its released code is run unchanged at the repository defaults, with the host's per-window risk score as the small model and the frozen verifier's per-window probability, scored on every window in a separate pass, as the large model. The router fires at a fixed probability of one half, so its operating point is set on the pool as in the FAR-matched comparison of prior work: both probability columns pass through the monotone piecewise-linear map that sends τ to one half, the unmodified router is re-run, and the frozen τ is the smallest value on a grid of 0.01 whose pool count stays within B. With the zero-shot host the cap of 10 is met at $\tau = 0.98$ with 10 pool clips, and $B = 4$ is infeasible on the grid. With the distilled host it is met at $\tau = 0.74$ with 9 pool clips. A clip counts as alarmed when the earliest delivered positive falls inside the horizon. At the native operating point of one half the router alarms on 49 and 22 of the 214 pool clips and on 42 and 24 of the 78 test normals, at 126 and 95 hits. Under the cap of 10 it reaches

47 and 62 hits at 13 and 8 test false alarms, and 6 and 11 hits at 6 and 5 on the development set. Transferred unchanged to the confirmation corpus, the same points give 42 and 62 hits at 32 and 35 false alarms, against 79 at 26 and 92 at 38 for the zero-shot and distilled pairs (Table 15, Figure 4(c)). A deferred step reads the large model's answer before deciding whether it counted, so the router's own call count understates the calls a live system pays by a factor of 1.2 to 1.4 at the native operating point, and both counts are recorded.

SPRINT at its native operating point. SPRINT (Qiu et al., 2026) asks a large model whether danger is occurring or about to occur over an anchored window that grows from the clip start, sampled at 2 frames per second up to 32 frames, and answers yes or no. Our adapter keeps the prompt, the anchored window, and the sampling, serves the model through an OpenAI-compatible endpoint, reads frames from our 10 frame-per-second extraction, advances the window end online every 0.5 s from 0.5 s so that a warning time exists, and leaves out the paper's cause-attribution metrics. The answer is binary, so there is no score to calibrate and the system is reported at its native operating point: it alarms on 26 of the 214 pool clips, and on the development set it reaches 18, 9, and 8 hits at 6 false alarms. A warning issued at t is delivered at t plus the latency of that call. The batch ran 24 requests in parallel, so the recorded latencies describe throughput. A serial probe over the first ten development clips gives $1.59 + 0.033\,n$ seconds for a call with n frames, 1.9 s at the median, and this fit is used for delivery. On the test split the probe makes 38 calls per clip, 67 s of 8B time at the median, against 11.5 verifier calls per clip for our loop, 0.33 s each at the median and 3.3 s per clip. On the confirmation corpus it reaches 104, 47, and 34 hits at 46 false alarms against 92, 37, and 23 at 38 for Heimdall, gaining 41 clips and losing 29 at $\tau = 0$, at 23 calls per clip (Table 15).

The PaSBench-Video protocol. PaSBench-Video (Zhao et al., 2026a) evaluates models under a streaming protocol but released no code, so we reimplement the protocol from its paper. Evaluation starts at 0.5 s and advances every 0.3 s; each query sees the 10 frame-per-second frames of the last three seconds, up to 31, on one 1344×1344 contact sheet of 6×6 cells read left to right and top to bottom; the system and user prompts are the paper's verbatim, the model returns a JSON object with a warn decision, a risk level, a confidence, and a state summary, and the state summary is passed back on the next query without the previous decision. Generation uses temperature zero; hazardous clips run to the accident and normal clips to our 40 s horizon, and the loop does not stop at the first warning. The decision is the model's own warn field. Its confidence is saturated, with 86 percent of test-split warnings at 0.98 or more, so the protocol is reported at its native operating point. It alarms on 17 of the 214 pool clips and on the development set reaches 2, 2, and 1 hits at no false alarm. A serial probe gives 3.84 s per call, which is used for delivery in place of the 7.4 s recorded under 24 parallel requests. On the test split the protocol makes 63 calls per clip and warns on 13 hazardous and 9 normal clips, and on the confirmation corpus on 7 hazardous and 12 normal clips, so under the same 8B it is far more conservative than our loop, which reads the same verifier through a risk question on eight frames.

Two baselines kept out of Table 11. DRIVE (Bao et al., 2021), with official DADA-2000 weights on the driving subset, scores 76 of the 97 normals of a separate driving pool at exactly 1.0, so the only feasible operating point never alarms and its recall is zero. Its weights do not transfer to this footage. Our own risk-scheduled cascade queries the verifier on a risk-dependent schedule and calibrates an advisory threshold on site normals with a rule other than the pool protocol. It reaches 27.0, 5.4, and 5.4 percent at 5.0 percent FAR on the development set.

Table 12: Rows kept out of Table 11, development set, in percent: single-model systems at Eq. (1) and the write-back variant of the distilled-verifier pair at its joint point. ‡ arm frozen last.

System	R0	R3	R5	FAR
Zero-shot host alone	16.2	0.0	0.0	15.0
Frozen verifier alone	16.2	2.7	2.7	5.0
Distilled host alone	32.4	10.8	8.1	10.0
Distilled verifier alone ‡	37.8	18.9	13.5	10.0
Heimdall, distilled verifier, write-back ‡	48.6	16.2	13.5	10.0

Margin curves. The earliest possible alarm on the development set is 2.11 s, so 7 clips are unreachable at $\tau = 3$ and 11 at $\tau = 5$. Table 13 replaces the three discrete margins of Table 11 with the full curve on the development set, plotted in Figure 5(a), using the urgent and advisory union, which is the only channel comparable to the published baselines. Two-model rows are at their joint operating points,

and OASIS and HERMES are single-stream systems as run. Our systems lead at a near-zero margin and HERMES reaches parity between one and two seconds and leads over the rest of the range. With the distilled verifier the parity point moves to about three seconds and HERMES leads beyond it. Conditional on hitting, the median lead over each system's own hit set is 1.37 s for the distilled system, 1.62 s with write-back, 3.10 s for OASIS and 13.96 s for HERMES, and the share of hits with a lead under two seconds is 58 to 62 percent for ours against 33 and 17 percent. At $\tau = 0.5$ s, HERMES commits at its own first observation on 9 of 12 hits against 2 of 19 for Heimdall, and pays four false alarms of twenty for it. Computed from each system's own first available observation, the recorded schedules permit almost the same number of hits for both systems at every margin, 37 against 37 at $\tau = 0$ down to 26 against 25 at five seconds, while the realized curves differ by a factor of two, so the comparison is about decision rules and not latency.

Table 13: Development set: share of the 37 hazardous clips warned at least τ seconds before the accident, and FAR on the 20 normal clips, in percent.

System	0	0.75	1.0	1.25	2.0	3.0	5.0	10	FAR
Zero-shot pair	24.3	13.5	8.1	5.4	2.7	2.7	2.7	2.7	15.0
Heimdall	37.8	24.3	21.6	16.2	10.8	10.8	10.8	5.4	10.0
+ verdict write-back	32.4	18.9	16.2	10.8	10.8	10.8	8.1	2.7	10.0
one-second cadence	43.2	24.3	18.9	16.2	10.8	10.8	10.8	5.4	5.0
Heimdall, distilled verifier	51.4	35.1	29.7	27.0	21.6	18.9	16.2	8.1	5.0
+ verdict write-back	48.6	32.4	32.4	24.3	18.9	16.2	13.5	8.1	10.0
OASIS (Liang et al., 2026)	32.4	24.3	24.3	21.6	21.6	16.2	10.8	8.1	10.0
HERMES (Zhang et al., 2026)	32.4	32.4	32.4	32.4	27.0	21.6	21.6	18.9	20.0
Orin NX loop, per-stream thresholds	32.4	27.0	27.0	24.3	21.6	18.9	16.2	5.4	25.0
Orin NX loop, joint thresholds	21.6	13.5	13.5	13.5	13.5	10.8	5.4	2.7	15.0
AGX Orin loop, joint thresholds	27.0	10.8	10.8	10.8	10.8	10.8	8.1	8.1	5.0

Thresholds and cost. Every compared system uses the running-maximum rule of Eq. (1), and clips with equal scores are never split to fill a budget. The candidate set of Eq. (1) is $\Theta = \{0\} \cup \{\mathrm{nextafter}(S_i, +\infty)\}$, which holds zero and the next representable number above each pool maximum. An alarm raised after the accident is not a hit at any margin, and a hazardous clip whose accident falls beyond H_i can still be hit but is listed separately. Table 14 lists the frozen thresholds. Under per-stream calibration, the full policy realizes 19 to 22 false alarms on the 214-clip pool against the nominal 10, which is the gap Eq. (2) closes. The per-stream verifier threshold is 0.99945 for every configuration that uses the frozen verifier except the one-second cadence (0.99929), and 0.92414 for the distilled verifier. The 3B backbone of the memory baselines calibrates at 0.50 and the small host calibrates between 0.18 and 0.82 depending on how it was trained. HERMES runs one 3B model, against our 2B host and 8B verifier. On the development set the distilled system issues 1295 host calls and 635 verifier calls over 57 clips, and halving the decision period raises the verifier calls to 1091. Client-side request time for a full development run is 195 s of verifier time and 177 s of host time inside a wall-clock hour dominated by real-time playback, so latency sums are not a measure of GPU occupancy.

G Confirmation Corpora and Precursor Length

Exposure and denominators. In Table 15 recall is a share of 299 hazardous clips for the confirmation corpus, 298 with the training overlap removed, 245 with the test-split overlap removed, and 35 for the self-collected corpus. Of the 482 clips, 19 appear in the host's training manifest (1 hazardous, 18 normal) and 309 carry a prior experimental record, which is why it serves as a confirmation corpus after every threshold is frozen. With the overlap removed the rows have 298 hazardous and 165 normal clips, and the paired gain of the distilled pair over the zero-shot pair is unchanged at 19 clips gained and 6 lost. With the test-split overlap removed, the distilled pair gains 14 clips and loses 6 at zero margin and gains 12 and loses none at three seconds. Site calibration on the self-collected corpus uses the 15 normals the corpus designates for that role, which are disjoint from the 50 evaluation clips. On this corpus the development thresholds produce no alarm, because the verifier scores this camera lower on 33 of 35 hazardous clips; recalibration moves the threshold but leaves the ordering of the clips unchanged.

Table 14: Thresholds on each configuration's own 214-clip normal pool: per-stream host thresholds of Eq. (1) at $10/214$ with the pool false-alarm rate the full policy then realizes, and the joint $\lambda^\star$ of Eq. (2) with its thresholds.

	Per-stream		Joint			
Configuration	θ_h	FAR (%)	$\lambda^\star$	θ_h	θ_v	FAR (%)
Zero-shot pair	0.9665	8.4	6	0.9694	0.99984	4.7
Heimdall	0.5309	8.9	6	0.5863	0.99984	4.7
+ write-back	0.5309	9.8	3	0.6543	0.99994	3.3
one-second cadence	0.5268	9.3	5	0.6028	0.99988	4.7
Same-time teacher	0.4156	8.4	5	0.5513	0.99984	4.7
Fixed-offset teacher	0.1830	8.9	6	0.2190	0.99984	4.7
Annotation-supervised	0.8246	9.3	5	0.8788	0.99984	4.7
Heimdall, distilled verifier	0.5268	8.4	5	0.6028	0.94660	4.2
+ write-back	0.5268	10.3	4	0.6037	0.95257	4.7
OASIS	0.99683	4.2	single stream, unchanged			
HERMES, released	0.50000	1.9	single stream, unchanged			
HERMES, context-matched	0.50000	4.2	single stream, unchanged			
HERMES, memory-free	0.53121	1.9	single stream, unchanged			
Risk-scheduled cascade	0.6000	n/a	own site rule, verifier 0.79014			

Table 15: Confirmation corpora with thresholds frozen on the development pools, in percent of the hazardous clips (Rτ) and of the normal clips listed (FAR). Rows marked training overlap removed exclude the 19 clips shared with the host's training manifest, and rows marked test-split overlap removed exclude the 54 PaSBench-Video clips shared with the test split.

Corpus and configuration	R0	R3	R5	FAR	Normals
Confirmation, zero-shot pair, per-stream	29.4	8.4	6.0	18.6	183
Confirmation, zero-shot pair, joint	26.4	7.0	5.7	14.2	183
training overlap removed	26.2	7.0	5.7	14.5	165
test-split overlap removed	25.3	1.6	0.8	14.2	183
Confirmation, Heimdall, per-stream	34.1	14.7	9.0	29.0	183
Confirmation, Heimdall, joint	30.8	12.4	7.7	20.8	183
training overlap removed	30.5	12.4	7.7	22.4	165
test-split overlap removed	28.6	6.5	1.2	20.8	183
Confirmation, Heimdall + write-back, per-stream	36.5	16.1	9.7	33.3	183
Confirmation, Heimdall + write-back, joint	27.8	10.0	6.0	19.7	183
Confirmation, Heimdall with distilled verifier, joint	29.1	11.4	6.7	19.1	183
Confirmation, StreamSense router, zero-shot host, pool τ	14.0	7.0	5.7	17.5	183
Confirmation, StreamSense router, distilled host, pool τ	20.7	7.0	5.0	19.1	183
Confirmation, SPRINT probe, native operating point	34.8	15.7	11.4	25.1	183
Confirmation, PaSBench-Video protocol, native operating point	2.3	1.3	1.3	6.6	183
Confirmation, zero-shot pair, two-fold recalibration	15.1	4.3	4.0	3.8	183
Confirmation, Heimdall, two-fold recalibration	18.4	6.0	5.0	5.5	183
Self-collected, pool threshold	0.0	0.0	0.0	0.0	15
Self-collected, site calibration	5.7	0.0	0.0	13.3	15

Two-fold recalibration. The two-fold rows split the 183 normals into two folds stratified by source with a fixed seed. Each fold serves as the pool for Eq. (2) at the nominal rate of the main configuration, $B = 5$ and 4 for 94 and 89 normals, and the resulting thresholds are evaluated on all 299 hazardous clips and the normals of the other fold. Hits are averaged over the two folds and false alarms are summed over the held-out normals, so every normal clip is held out once. The verifier threshold lands at 0.99993 in three of the four folds, and the host threshold rises from 0.59 to 0.74 and 0.80 for Heimdall and from 0.97 to 0.98 for the zero-shot pair. The realized rates return to the nominal five percent and the ordering of the two pairs holds at every margin.

By source. Table 16 splits the joint operating point of Table 15 by source. The gain over the zero-shot pair and the extra false alarms both come from OOPS, and the leads at three and five seconds come from OOPS and from the PaSBench-Video clips shared with the test split. On the four sources with the longest precursors, EDF, OCCU, OmniFall, and CAUCAFall, neither pair warns at three seconds, and on SmartHome-Bench both pairs alarm on about one normal clip in five while hitting one or two

Table 16: Confirmation corpus by source at the joint operating point with $B = 10$. Hits at $\tau = 0$, 3, and 5 s; FP counts normal clips with an alarm. EDF and OCCU are two subsets of one release.

			Zero-shot pair		Heimdall	
Source	Hazardous	Normal	R0/R3/R5	FP	R0/R3/R5	FP
OOPS	75	37	46/3/1	7	58/15/2	17
SmartHome-Bench	55	88	2/1/1	18	1/1/1	21
PaSBench-Video	54	0	17/17/15	–	22/21/20	–
EDF	42	21	0/0/0	0	0/0/0	0
OCCU	42	21	2/0/0	1	0/0/0	0
OmniFall	21	11	4/0/0	0	4/0/0	0
CAUCAFall	10	5	8/0/0	0	7/0/0	0
All	299	183	79/21/17	26	92/37/23	38

hazardous clips. The source thresholds were frozen on a pool of campus and home normals, so these rows also show which sources the pool does not represent and where a site pool would be needed.

Test split by source. At the joint budget the four hosts of Table 1 warn before the accident on 7, 13, 10, and 18 of the 60 driving clips (zero-shot, same-time, fixed-offset, Heimdall), on 26, 25, 28, and 32 of the 98 daily-life clips, and on 4, 3, 6, and 5 of the 23 healthcare and industrial clips; false alarms fall on 5, 4, 3, and 3 of the 32 driving normals and on 3, 1, 1, and 1 of the 46 daily-life normals. The distilled-verifier pair warns on 23 driving, 25 daily-life, and 8 healthcare and industrial clips, with 7 driving and 1 daily-life false alarms.

Every hazardous clip in the development set and 213 of the 299 in the confirmation corpus carry an annotated onset of the visible precursor, so the precursor length D, the accident time minus that onset, is known. Table 17 breaks the hits of the joint operating points down by D. The 86 confirmation clips without an annotated onset are hit at 16.3 and 14.0 percent by the two systems and are not in the table. Precursors are short. The median D is 2.4 s on the development set and 2.2 s on the confirmation corpus, and 84 and 75 percent of annotated clips have $D < 5$ s. On the development set every hit falls on a precursor shorter than five seconds, and the six longer precursors are missed by every configuration. On the confirmation corpus, Heimdall hits 57 of the 160 short precursors and 23 of the 53 long ones, and its hits at three and five seconds are almost all on precursors of ten seconds or more, which come mostly from one recording site. The zero-shot pair hits 47 of 160 and 18 of 53. Precursor length therefore bounds the lead a warning can have but does not explain the gap between sources. Alarms that precede the annotated onset are examined below.

Table 17: Hits by precursor length D at the joint operating points, in percent of each bucket at $\tau = 0$, with three and five seconds in parentheses.

System	$D < 2$	$2 \le D < 5$	$5 \le D < 10$	$D \ge 10$
Development, clips per bucket	13	18	3	3
Zero-shot pair	38.5 (0.0, 0.0)	22.2 (5.6, 5.6)	0.0	0.0
Heimdall	46.2 (0.0, 0.0)	44.4 (22.2, 22.2)	0.0	0.0
with distilled verifier	61.5 (15.4, 7.7)	61.1 (27.8, 27.8)	0.0	0.0
Confirmation, clips per bucket	97	63	15	38
Zero-shot pair	39.2 (2.1, 0.0)	14.3 (0.0, 0.0)	20.0 (20.0, 13.3)	39.5 (39.5, 36.8)
Heimdall	47.4 (10.3, 0.0)	17.5 (6.3, 1.6)	33.3 (26.7, 20.0)	47.4 (47.4, 47.4)

Alarms before the annotated onset. At the joint operating point Heimdall commits before the annotated onset on 4 of its 14 development hits and on 24 of the 92 confirmation hits among the 213 clips with an onset. All but two of these are urgent-path alarms in the first two to four seconds of the clip. Two checks bound how much of this is a scene prior. A control that uses only the first available host window and the first verifier answer, at the same thresholds,

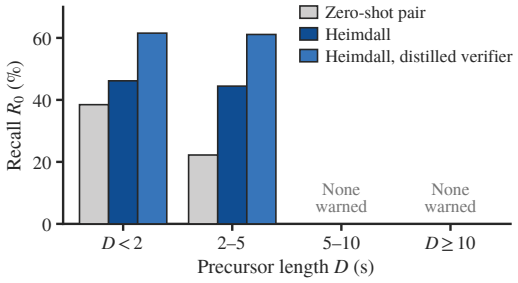


Figure 8: Share of each precursor-length bucket hit at $\tau = 0$ on the development set.

reaches 3 of 37 hits at no false alarms on the development set and 35 of 299 at 3 false alarms on the confirmation corpus, against 14 and 92 for the full policy, so most hits need more than the opening scene. Among host windows that precede the annotated onset of a hazardous clip, 5.7 percent on the development set and 10.5 percent on the confirmation corpus score above θ_{h}, against 1.6 to 3.5 percent of windows on normal clips, so the host does respond to something in the pre-onset footage of hazardous clips more often than on normal footage. Both an earlier cue and a scene prior fit this record, and the first-window control bounds how much the scene prior explains.

Miss taxonomy. Every hazardous clip missed at $\tau = 0$ by Heimdall at its joint operating point is assigned to the first cause that applies. The confirmation corpus is dominated by a cause that no decision rule can address, and the development set is not.

Table 18: Causes of the clips Heimdall misses at its joint operating point, in percent of the misses.

	Cause	Confirmation	Development
A	No valid verifier answer before the accident	3.4	0.0
B	Verdicts arrived, none ever reached 0.5	55.6	8.7
C	A verdict reached 0.5 but not θ_{v}, and no host score reached θ_{h}	36.7	43.5
D	Alarm raised inside the horizon but after the accident	2.9	43.5
E	Accident beyond the 40 s horizon	1.4	4.3

Class B is not a near miss. On those 115 clips, the verifier never assigns even one half to the hazard, and sweeping θ_{v} downward on the recorded logs recovers them only below a probability of 10^{-6}, where every normal clip alarms, so these misses are not a threshold choice and raising the verifier's discriminability is the only route to them, which is what Section 5 proposes. Class C, a verdict above one half that the verifier's own threshold does not accept, is the largest development-set class together with class D, an alarm that arrives late by a median of about one second. Both are where a better host or a lower verifier threshold at the same budget would act. The two corpora therefore fail for different reasons, the confirmation corpus for lack of verifier evidence and the development set mostly for timing, where a better host would act.

H Edge Deployment

Orin NX. For the closed loop the host runs on a 16 GB Jetson Orin NX as a four-bit GGUF model under llama.cpp, because no vLLM build supports this model on the Orin. The four-bit GPTQ weights (Frantar et al., 2023) used for the server-side equivalence check are a different quantization. Median latency per window is 1.03 to 1.07 s for the two input formats and 1.17 to 1.22 s for the same model in 16-bit. Board power is 19.5 to 20.8 W under load against 5.4 W idle, and net energy is about 15 J per window. An earlier run at per-stream thresholds placed the host on the board and the verifier on a server over a local network. It completed the 57 development clips in 25.4 min of real time with a median verifier round trip of 0.47 s, and 15 answers arrived after their horizon. The per-stream run is kept in Table 13, and the joint-threshold run took 25.6 min. At the joint thresholds of the main configuration ($\theta_{\mathrm{h}} = 0.5863$, $\theta_{\mathrm{v}} = 0.99984$) the board reaches 8, 4, and 2 pre-accident hits at three false alarms against 14, 4, and 4 at two for the same configuration on the server. The host's 1.1 s per window is slower than the one-second grid, so the schedule drifts and 95 of the 666 nominal verifier requests are never issued. The verifier round trip over the network is 0.46 s at the median, and 13 of the 571 issued requests returned after the horizon and were excluded from the decisions. Alarms that the server commits shortly before the accident arrive after it on the board, which removes zero-margin hits and leaves the three-second count unchanged. The verifier server's cache state was not recorded for these two runs, and a later probe found no cache acceleration on either server. On this board quantization, the inference stack, and the schedule drift act together. The latency histograms of Figure 6(b) cover the same 1295 development windows on both machines.

AGX Orin. The same loop was repeated on a 64 GB Jetson AGX Orin with the host in 16-bit GGUF form under the same llama.cpp stack, the same joint thresholds, and the same verifier server, reached over an overlay network instead of the wired one. The host takes 0.554 s per window at the median and at most 0.673 s, so every window fits the one-second grid. Of the 666 nominal verifier

requests 631 are issued, two answers arrive after their horizon and are excluded, and the verifier round trip is 0.36 s at the median. The loop completes the 57 clips in 23.1 min and reaches 10, 4, and 3 pre-accident hits at one false alarm. Paired with the server run it gains one clip and loses five at $\tau = 0$, and on the nine clips both catch the board commits 0.29 s later on average. The GPU and CPU rails together draw 17.9 W on average and 30.1 W at peak during the loop. Since the host keeps the grid, the remaining gap comes from the GGUF inference stack, whose image preprocessing and arithmetic differ from the vLLM deployment on the server.