Sensor-Stream Integrity and Decision Fidelity in Helmet-Mounted IMU Crash Detection

Usman Azfar

Author preprint (not peer reviewed).
This work has been submitted to the IEEE for possible publication. Copyright may be transferred without notice, after which this version may no longer be accessible.

***Abstract*—A high offline score does not guarantee that a learned inertial detector receives temporally valid samples or preserves its decisions in continuous operation. We evaluate a frozen dual-signal motorcycle crash-detection policy on a helmet-mounted inertial module and Raspberry Pi 4 using class-conditional decision fidelity, host-observed temporal validity, and decision-opportunity coverage. On 5,437 held-out windows, the fully composed causal-feature and INT8 deployment path agrees with its offline reference on 99.65% overall but 90.21% of crash-like decisions. In a 90-minute polling run, 267,891 error-free sensor reads coexist with 93.07% candidate temporal validity and only 80.99% decision coverage. A register-verified raw-domain comparison supports polling/FIFO agreement under matched stationary conditions before an A–B–A intervention. Controlled comparisons isolate a train/serve peak-jerk scale discrepancy; correcting it changes few decisions and leaves six confirmed benign-handling alerts unchanged. Three simple non-neural or linear policies demonstrate that the offline decision-validation procedure is not tied to the frozen network, although external datasets expose poor generalization and operating-point tradeoffs. Simulated and car-recorded data account for 94.1% of held-out normal windows; session-clustered intervals across ten sessions are wide. These results provide a reproducible method for interrogating edge sensing and decisions, while leaving natural-crash and notification reliability as separate validation tasks.**

***Index Terms*—Inertial sensing, sensor-stream integrity, decision fidelity, edge artificial intelligence, motorcycle crash detection, deployment validation.**

I. Introduction

INERTIAL sensing is attractive for motorcycle crash detection because a compact six-axis inertial measurement unit (IMU) can continuously observe translational and rotational motion without line-of-sight sensing. Prior powered-two-wheeler (PTW) studies span embedded threshold logic, learned hazard recognition, critical-event classification, and machine-learning collision detection [1]–[5]. Related IEEE Sensors Journal work investigates event detection with wearables [6]. Reconstruction-based and smartphone-oriented studies further motivate learned anomaly and notification pipelines [7], [8], while recent smart-helmet systems increasingly integrate sensing, local computation, and communication [9], [10]. Offline accuracy does not measure stream fidelity.

Sensor hardware, sampling and placement heterogeneity can impair inertial recognition across devices and operating systems [11]. Research on edge-sensor fault diagnosis further highlights the distinction between physical data quality and downstream model decisions [12]. Deployment changes more than model latency. Offline zero-phase filtering must become causal; quantization perturbs score distributions; periodic acquisition competes with an operating system; and successful I^2C reads do not guarantee that samples form a temporally valid inference window. MLPerf Tiny formalizes accuracy, latency, and energy as embedded-ML benchmarking dimensions [13], but does not address temporal validity of an externally sampled stream. Quantization work such as Jacob *et al.* [14] characterizes efficient integer inference, but not whether numerical changes cross an application-specific decision boundary. Automotive ML verification literature likewise argues that evaluation must extend beyond the learned model to the surrounding system context [15].

This paper evaluates preservation of an offline-developed IMU policy's decisions and timing assumptions during continuous helmet-mounted sensing, from acquisition to local GPIO.

We make four experimentally grounded contributions: (1) separate reference-to-deployment decision fidelity, class-specific transitions, host-observed temporal validity, and actual decision-opportunity coverage; (2) document sensing configurations, stationary noise, raw-domain acquisition equivalence, and repeated Linux scheduling conditions; (3) isolate a peak-jerk train/serve discrepancy and test whether it changes actual benign-handling alerts; and (4) apply the offline decision and preprocessing protocol to three transparent baseline policies, reporting session-clustered uncertainty and external-domain failure modes. The frozen detector is a fixed test case.

The methodological contribution is the ordered coupling of a register-level acquisition gate with decision-transition analysis and explicit opportunity denominators: a window can be classified only after acquisition and timing checks, and conversion agreement is reported by class at unchanged thresholds. The simple V_T and C_D ratios expose distinct failure modes when paired with a fixed decision policy.

II. Related Work and Deployment Gap

Motorcycle sensing studies investigate inertial crash triggers, critical-event recognition, and data-driven hazard models [1]–[5]. A bicycle-rider system in this journal [6] offers an adjacent wearable-sensing comparison, while edge sensor-fault work [12] addresses data quality independently of this paper's particular IMU policy. Smart-helmet prototypes also integrate computation and notification [9], [10]. Event accuracy does not measure stream integrity.

Inertial sensing heterogeneity can arise from sampling, device and software differences [11]. MLPerf Tiny addresses

U. Azfar is with the Department of Computer Science, Faculty of Computing and Information Technology, University of the Punjab, Lahore, Pakistan (e-mail: usmanazfarshafiq@gmail.com).

This research received no external funding.

accuracy, latency and energy benchmarking [13], while integer-only inference introduces a distinct numerical transformation [14]. Sculley *et al.* identify hidden dependencies across deployed ML systems [16]; Breck *et al.* specifically advocate checking feature equivalence between training and serving [17]. Verification reviews emphasize validation of the surrounding software system [15]. Linux scheduling introduces timing variability [18]. Independent sensor-timing calibration work in *IEEE Sensors Journal* [19] illustrates the stronger time-base evidence needed beyond host timestamps; the pertinent register map specifies the hardware FIFO and data-ready mechanisms [20]. We combine these threads in a deployment protocol that measures *host-observed* temporal validity, actual decision opportunities and class-conditional policy changes independently. Physical event capture remains unvalidated.

III. Decision-Centric Deployment Method

A. Frozen Detector and Decision Policy

The detector was frozen before deployment testing. Its 66 development sessions were obtained from third-party sources: CARLA simulated driving [21], a subset of the STRIDE car-mounted smartphone recordings [22], Boubezoul *et al.* fall and near-fall motorcycle recordings [23], [24], and DaRUS operational/collision simulations [25]. No team-recorded riding or impact traces entered model development. Session-level splitting (seed 42) yielded 46/10/10 train/validation/test sessions and 18,720/4,713/5,437 windows. STRIDE contributed six training, one validation and two held-out test sessions; no session was reused across splits. The held-out test split contains 5,294 normal and 143 crash-like windows. Of its normal windows, 2,614 are CARLA simulation and 2,368 come from two STRIDE car sessions, together 94.1% of normal windows. Accordingly, these results measure transformations on a fixed heterogeneous benchmark; they cannot establish real helmet-mounted motorcycle false-alert reliability. For context, the frozen deployed OR rule on this split has TP/FP/TN/FN=29/3/5,291/114, precision 0.906, recall 0.203, and F1 0.331; the 32 reference-positive windows comprise 17 classifier-only, one reconstruction-only, and 14 both-branch positives (archived L1 branch counts).

Each input contains 60 six-axis samples ($\mathbf{X}_t \in \mathbb{R}^{60\times 6}$; 1.2 s at 50 Hz) and seven auxiliary statistics (acceleration and angular-rate magnitude mean, SD and peak, plus peak jerk). These features were not re-optimized. A shared Conv1D encoder feeds a reconstruction decoder and a classifier. The classifier consumes the learned embedding, reconstruction error and auxiliaries. The two operational scores are crash probability p_t and reconstruction error r_t. Training used Adam (batch 128), early stopping on validation AUC, and 1:2 classification-to-reconstruction loss weights; the complete network, optimizer configuration, normalizer and hashes are archived.

The deployed window rule is

$$a_t = \mathbb{I}(p_t > \theta_c) \ \vee \ \mathbb{I}(r_t > \theta_r), \tag{1}$$

with displayed thresholds $\theta_c = 0.9892$ and $\theta_r = 10.879$. The exact frozen values are retained in the reproducibility package. On validation data, θ_c is the classifier threshold maximizing F1, while θ_r is the 99th percentile of reconstruction error among normal validation windows. A final trigger requires three consecutive anomalous windows. With a 15-sample stride, nominal decision spacing is 0.3 s; the minimum interval from the first anomalous decision to third-window confirmation is therefore approximately 0.6 s. The processing budget is 300 ms per eligible decision window.

B. Causal Preprocessing and Stream Integrity

Offline development removes gravity with a second-order 0.3-Hz Butterworth high-pass applied by zero-phase forward/backward filtering. A streaming implementation cannot use future samples, so the live pipeline uses the frozen causal exponential moving-average (EMA) gravity estimate

$$\begin{aligned} \alpha &= \exp(-2\pi f_c \Delta t), & f_c &= 0.3 \text{ Hz}, \\ \mathbf{g}_t &= \alpha \mathbf{g}_{t-1} + (1-\alpha)\mathbf{a}_t, & \mathbf{a}_t^{\text{dyn}} &= \mathbf{a}_t - \mathbf{g}_t, \end{aligned} \tag{2}$$

with target $\Delta t = 0.02$ s. This corresponds to $\alpha \approx 0.9630$ and an equivalent first-order time constant $1/(2\pi f_c) \approx 0.531$ s. The offline zero-phase Butterworth and deployed causal first-order EMA differ in phase, order, and roll-off despite their shared nominal 0.3-Hz corner. At 50 Hz, the exact deployed coefficient is $\alpha = 0.9630026534$. The reported F_D therefore measures fidelity of the complete preprocessing replacement, not isolated causality. A cutoff sweep and causal Butterworth control appear in Sec. IV-H. On cold start or after an explicit long-gap filter reset, the first acceleration sample initializes $\mathbf{g}$; no inference occurs until 60 fresh samples have accumulated.

Host-paced 50-Hz I^{2}C polling timestamps reads, not physical sample instants. The `SampleStreamGuard` flags gaps relative to the nominal 20-ms period, invalidates the pending window, and demands 60 fresh samples before scoring resumes. The initial polling campaigns use a 30-ms maximum inter-read interval and 10-ms too-fast diagnostic threshold. For the subsequent validation runs, the maximum permitted gap is 60 ms; the 10-ms diagnostic remains unchanged. Ordinary gaps clear the window while preserving the EMA; gaps of at least 1 s additionally reset the EMA. None of these limits was optimized using reported live test outcomes. Coverage results are conditional on their recorded guards and settings (Tables II and III, Supplement Table S1); 30- and 60-ms campaigns are not pooled.

C. Deployment-Fidelity Metrics

For a transformation T applied to a reference decision sequence a_i, window decision fidelity is

$$F_D(T) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\Big(a_i = a_i^{(T)}\Big). \tag{3}$$

Session-level fidelity is computed analogously from final trigger/no-trigger outcomes. For continuous operation, temporal validity is

$$V_T = \frac{N_{\text{scored}}}{N_{\text{scored}} + N_{\text{invalid}}}. \tag{4}$$

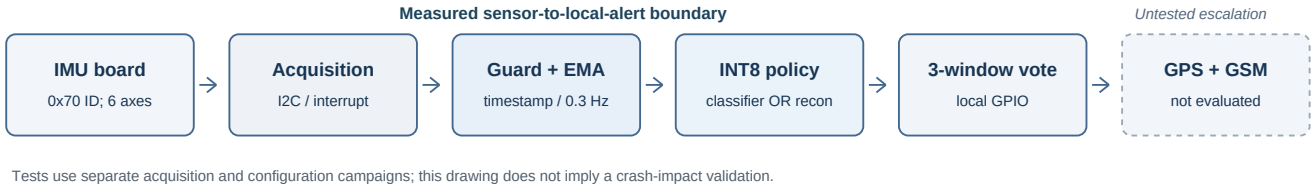


Fig. 1. HelmX sensing and emergency-escalation architecture. The solid path marks the experimentally validated sensor-to-local-alert boundary: an identity-qualified IMU acquisition path, timestamp guarding, causal preprocessing, the frozen dual-signal model, threshold OR logic, three-window confirmation, and GPIO/local alert. The dashed extension denotes the integrated but unvalidated downstream chain: an approximately 30-s cancellation interval followed, if uncancelled, by NEO-6M GPS localization and SIM800L GSM/SMS escalation. GPS acquisition and message-delivery reliability are not characterized in this study.

V_T describes validity among explicitly formed candidates, but a gap also suppresses stride positions while the 60-sample buffer refills. We therefore report decision coverage

$$C_D = \frac{N_{\text{scored}}}{N_{\text{nominal}}}, \tag{5}$$

where N_{nominal} is the number of stride-aligned decision opportunities implied by the received stream duration. An earlier replay helper also evaluated uninterrupted 90-sample (1.8-s) sequence continuity. Successful bus transactions can therefore coexist with $V_T < 1$ and lower C_D. Processing latency is reported at p50/p95/p99/max; the 300-ms stride is a compute budget.

D. Hardware, Mounting, and Experimental Conditions

Deployment used a Raspberry Pi 4 Model B Rev. 1.5 (8-GB nominal RAM), Debian 12 Bookworm, CPU-only TensorFlow Lite, one inference thread, and an MPU6050-labeled module confirmed by I^{2}C communication but not by silicon identification. The frozen polling configuration used $\pm 2g$ acceleration, $\pm 250^\circ$/s gyroscope, DLPF configuration 0 (wide-band/DLPF-off behavior), sample-rate-divider register 0, FIFO disabled, DATA_RDY interrupt disabled, and 100-kHz host I^{2}C. The host polling loop imposes the model-facing 50-Hz rate. The deployed model is INT8 TFLite with float32 external I/O. The CPU operated at approximately 1.8 GHz in the reported campaigns. Physical tests used the fully assembled DOT-approved Super Air 912 helmet. The IMU module was installed internally in the left-ear region and adhesively fixed to the rear side of an internal plastic helmet element behind the inner foam. The mount remained fixed across trials; its transfer function and resonance were not characterized.

Figure 1 identifies the measured local-alert boundary and separates it from downstream escalation. The deployed peak-jerk feature differs from the training-time time-normalized gradient. Section IV-E measures this train/serve difference under otherwise fixed paths, and the composed comparison retains the original deployed formula. Evaluation combines frozen replay/conversion tests, physical polling and motion trials, matched scheduler runs, raw-domain acquisition comparisons, and offline baselines. Detailed campaign inventory and guard settings appear in Supplement Table S1. After confirmation, a 30-s cancellation interval precedes the unvalidated NEO-6M GPS/SIM800L GSM alert extension (Fig. 1). Measured local latency spans eligible-window completion through scoring and decision update, not physical crash onset.

TABLE I
ISOLATED AND COMPOSED OR-DECISION FIDELITY, N=5,437

Path against offline Keras	Normal	Crash-like	Overall
Causal EMA only (L2)	99.96%	96.50%	99.87%
INT8 only (L3)	100.00%	95.80%	99.89%
Composed, original runtime (L1)	99.91%	90.21%	99.65%

IV. RESULTS

A. Decision Preservation and Model Limitations

A held-out test contains 5,294 normal and 143 weakly labelled crash-like windows from ten sessions. The offline Keras reference OR rule yields 29 true-positive and three false-positive windows (precision/recall/F1 = 0.906/0.203/0.331). Table I identifies each comparison explicitly. *Causal only* compares offline Keras/zero-phase/consistent training auxiliaries against Keras/causal EMA with identical auxiliaries (L2). *INT8 only* holds offline features fixed and changes the Keras model to its INT8 counterpart (L3); the reported values refer to the OR rule. *Composed* compares offline Keras/zero-phase/training auxiliaries with causal EMA/INT8/runtime auxiliaries (L1). These are window-level, frozen-threshold decisions, not independently labelled physical crash events.

The composed OR path has 13 negative-to-positive and six positive-to-negative decision transitions, with 99.65% overall agreement; on crash-like windows 10 negative-to-positive and four positive-to-negative transitions yield 90.21% agreement. Its weak-label precision/recall/F1 are 0.897/0.245/0.385 (35 of 143 labelled windows detected), compared with reference recall 0.203. All ten held-out session trigger/no-trigger outcomes match for the composed path, but several trigger indices shift and only five sessions contain crash-like events. A 2,000-draw session bootstrap gives composed recall interval 0.139–1.0, highlighting severe independent-event uncertainty.

A separate fixed-INT8 filter replay gives 99.61% overall agreement with INT8 inference on both comparison sides; it measures a filter-path substitution under quantized inference, rather than a combined filter/quantization change. Isolated L2 remains the matched Keras one-factor comparison. In classifier-only INT8 conversion, the frozen-threshold agreement

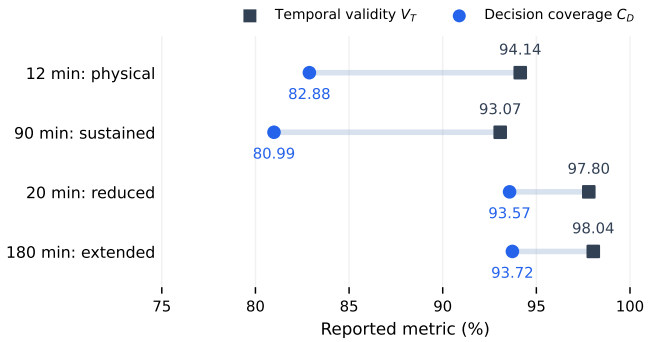


Fig. 2. Temporal validity and decision coverage across four archived host-polling campaigns (30-ms guard). Each point is a whole-run aggregate; the runs differ in duration/background conditions and are not repeated paired interventions. No time-resolved checkpoints are inferred.

is also 99.89% overall and 95.80% crash-like; FP16 yields 100% classifier-threshold agreement. The INT8 calibration set contains 192 normal and eight crash-like validation windows. At the largest recorded reconstruction outlier (index 5,410), Keras/FP16 reconstruction error 751.84 becomes 49.29 under INT8; the OR decision remains positive, and its numerical mechanism is undetermined. Diagnostic calibration alternatives were not substituted for the frozen deployed model.

B. Live Polling and Sensor Front End

Table II isolates the initial 30-ms guard experiments rather than comparing their coverage directly with later 60-ms runs. Most importantly, the 90-min soak recorded 267,891 successful I^{2}C reads and no recorded bus failures while $V_T = 93.07\%$ and $C_D = 80.99\%$: bus availability is not decision opportunity. The 180-min test recorded 539,981 samples, 33,734 scored windows, and no sensor failures, reconnects, exceptions, local alerts, or >300-ms processing observations. Its processing p99/max were 3.03/11.04 ms, mean process CPU 2.40%, mean resident memory 51.6 MB, maximum reported CPU temperature 55°C, and no thermal-throttle flag. These were bench/controlled operation, not real-road false-alert trials. Figure 2 compares V_T and C_D from the four independently archived run summaries. All four recorded zero bus read errors; the campaign-level points are not intermediate time-series checkpoints.

The register audit found `WHO_AM_I=0x70` in 300/300 reads over three boots, consistent with an MPU-6500-family register map [20]; the board label "MPU6050" is not proof of silicon identity. An unpaced 10-min mounted stationary capture in the original $\pm 2g/\pm 250°$/s setting recorded **186,081** reads, zero bus failures, acceleration magnitude 9.764 ± 0.043 m/s^2 and gyro magnitude 2.469 ± 0.401 deg/s (mean±SD). Wide-band DLPF-off host polling at 50 Hz cannot establish anti-alias performance; raw integer rail counts show the $\pm 250°$/s gyroscope saturated during vigorous benign handling (63 of 3,034 samples, or 2.08%, at the rails of one axis in a controlled phase). The accelerometer generally did not clip during those handling tasks; neither range was verified under impact excitation.

A separate filtered high-range configuration ($\pm 16g$, $\pm 2000°$/s, approximately 20-Hz DLPF and 50-Hz nominal output) was tested without retraining or replacing the model. A 20-min run collected 59,913 samples and 3,988 scored windows, with one guarded gap, 0 I^{2}C errors, 0 alerts, and $C_D = 0.999$ under the 60-ms guard. The high-range 11-phase benign-motion battery contained classifier-confirmed alerts during donning, removal, rotation, placement, and walking; reconstruction threshold crossings were zero. Better headroom did not eliminate false alarms or validate impact dynamics. Development data sources lack complete native sensor-placement and filter provenance, limiting extrapolation to mounted helmet crashes.

C. Hardware-FIFO Acquisition and Equivalence

Before comparing acquisition modes, we audited the hardware-FIFO implementation, register configuration, 12-byte signed-axis packet format, and host timestamp semantics. An initial harness had incompatible FIFO-capacity, divider and burst-timestamp assumptions; its apparent score changes were not accepted as sensor-level evidence. The validated hardware-FIFO configuration used `FIFO_EN=0x78`, filtered 50-Hz output, divider 19, and frequent FIFO draining. A matched 10-min stationary raw-domain gate showed per-axis polling/FIFO mean differences no larger than 0.0051 in their recorded units, SD ratios 0.98–1.04, no clipping, no I^{2}C errors, and no FIFO overflow.

Table III shows no reconstruction crossings or local alerts in any arm. In this stationary test, hardware-FIFO acquisition did not generate a new detector response after the equivalence gate. That result supports the gate procedure, not universal interchangeability under motion or impacts.

D. Scheduler Mechanisms and Alternative Acquisition

Direct `perf sched` profiling in an interactive loaded/reduced-background pair recorded maximum runqueue wait of 258.2/36.5 ms, 22.45/9.75 gaps per minute, 14.28/5.40% invalid candidates, and 83.04/18.42 ms maximum processing latency (loaded/reduced). These are mechanism clues from one pair, not a robust effect-size estimate.

Table IV gives replicated results from three successive 20-min pairs under the 30-ms guard, with the desktop/IDE stack present in both conditions. All six runs had zero bus failures and no post-window processing time above 300 ms; the apparent background-service effect cannot be generalized to an unattended helmet. The legacy 90-sample continuity estimate is reported separately in the Supplement because it was not recomputed using the corrected post-gap replay rule; neither metric estimates crash sensitivity.

Figure 3 shows the paired coverage observations, including the third-pair reversal; the average improvement under reduced load should not be mistaken for a deterministic or universally causal effect.

In a distinct six-pair randomized headless study (60-ms guard), 359,970 samples yielded eight timing gaps, all under Linux `SCHED_FIFO` priority. Priority scheduling decreased paired processing p99 by 0.113 ms (reported 95% interval $[-0.150, -0.065]$ ms) but decreased C_D by 0.0023. Six randomized interactive-service pairs without an IDE recorded

TABLE II
LIVE HOST-POLLING CAMPAIGNS UNDER THE 30-MS TIMING GUARD

Campaign	min	Scored	Gaps/min	V_T	C_D	p99 (ms)	Max (ms)	Read errors
Physical pipeline	12	1,975	10.25	94.14%	82.88%	3.19	8.72	0
Sustained soak	90	14,461	11.96	93.07%	80.99%	3.25	82.86	0
Reduced background	20	3,739	4.20	97.80%	93.57%	3.49	18.39	0
Extended soak	180	33,734	3.74	98.04%	93.72%	3.03	11.04	0

TABLE III
MATCHED POLLING–HARDWARE-FIFO–POLLING (60-MS GUARD; 20 MIN/ARM)

Metric	Poll A1	HW-FIFO B	Poll A2
Samples	59,837	60,595	59,956
Scored windows	3,986	4,029	3,994
Timing gaps	0	2	0
Decision coverage	1.000	0.998	1.000
Processing p99 (ms)	13.27	13.75	8.96
Reconstruction crossings	0	0	0
Confirmed alerts	0	0	0

TABLE IV
THREE MATCHED LOADED/REDUCED PAIRS (30-MS GUARD)

Endpoint	Loaded mean±SD	Reduced mean±SD
Gaps/min	7.05 ± 1.60	4.62 ± 1.68
V_T (%)	96.14 ± 0.94	97.55 ± 0.94
C_D (%)	88.25 ± 2.33	92.37 ± 2.48
Processing p99 (ms)	4.19 ± 0.40	3.69 ± 0.54

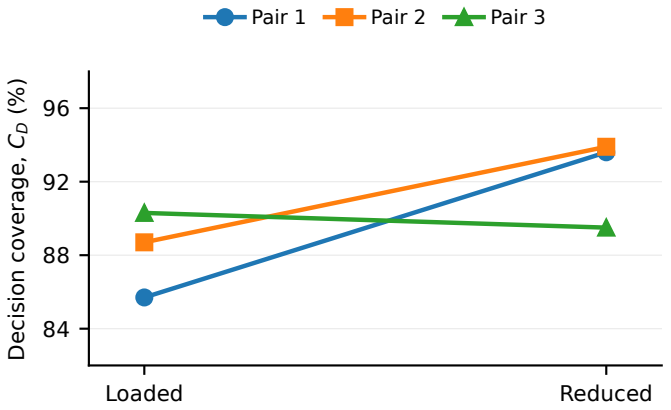


Fig. 3. Decision coverage in three paired 20-min loaded/reduced polling comparisons under the 30-ms guard (P1–P3). Values are from the corrected N3 replay of archived timestamps (post-gap sample retained); Pair 3 reverses the mean trend. These are descriptive pairs rather than randomized causal effects.

359,999 samples and no gaps in either condition. Thus `SCHED_FIFO` is not automatically superior to `SCHED_OTHER` for the measured stream.

A `DATA_RDY` interrupt-acquisition test obtained 60,688 samples and 4,042 windows in 20 min with no observed timing gaps; recorded kernel edge intervals averaged 19.773 ms (SD 0.066 ms). Hardware-FIFO plus `DATA_RDY` gave one host gap in 60,712 samples; a Linux IIO buffered-driver test collected 60,043 samples without guard resets, but kernel I^2C error telemetry was unavailable. FIFO burst-read host timestamps, edge timings and sensor sample clocks are distinct. Offline interpolation, guard-threshold and dt-adaptive EMA sensitivity experiments are not validated real-time safety mitigations.

The matched hardware-FIFO p99 processing times of 8.96–13.75 ms differ from the 3.03 ms in the three-hour polling campaign. The former uses a separate filtered acquisition harness, measurement epoch, and instrumentation path; this is *not* a matched latency comparison, so the archived data do not isolate a causal explanation for that difference.

E. Peak-Jerk Train/Serve Isolation

The training pipeline takes the maximum time-normalized gradient of acceleration magnitude, whereas the frozen runtime takes the maximum absolute adjacent-sample difference without dividing by time. Their held-out median values are 46.2 and 1.34 in different physical units; the median of within-window training/runtime ratios is 33.4, rather than the ratio of those medians; the normalized runtime auxiliary is concentrated near $z = -0.46$. This is train/serve skew. With the Keras model and offline filtering fixed, using the runtime jerk changes one of 5,437 classifier decisions relative to training features (classifier score MAE 0.0043). The corrected gradient restores the training definition but is a diagnostic replay, *not* a replacement for the frozen live runtime.

TABLE V
CLASSIFIER-BRANCH JERK DISCREPANCY: FROZEN MODEL, HELD-OUT SPLIT

Evaluation path	TP	FP	F1
Offline Keras, training jerk	29	2	0.333
Offline Keras, runtime jerk only	28	2	0.324
Offline Keras, corrected jerk	30	2	0.343
Live-style INT8/EMA, runtime jerk	34	3	0.378
Live-style INT8/EMA, corrected jerk	37	3	0.404

All five rows in Table V are *classifier-branch* results; for the historical composed OR rule, reference-to-deployment crash-like agreement is 90.21% and composed OR precision/recall/F1 are 0.897/0.245/0.385. Correcting jerk makes little difference to the OR decision sequence and does not explain the bulk of composed-path disagreement. Crucially, replaying the eleven accepted high-range handling phases (with the frozen guard, filter, INT8 model, thresholds and event logic) reproduces all original live decisions and six classifier-confirmed alarms. Correcting only jerk preserves *the same six alerting phases* (donning, removal, gentle and moderate rotation, soft placement and walking): 48 versus 53 classifier-positive windows, but six versus six confirmed alerts. Replay on an x86 host produced three subthreshold score discrepancies, none crossing a decision boundary; this is not new physical validation.

TABLE VI
HELD-OUT BASELINES (10 SESSIONS; 0.418 H NORMAL EXPOSURE)

Detector	ROC	F1	Crash sess.	FA/h
HelmX classifier	.946	.333	5/5	0.0
Accel-peak	.661	.107	4/5	28.7
Gyro-peak	.432	.051	5/5	67.0
Logistic regression	.830	.152	4/5	16.8

F. Cross-Policy Baselines and Session-Aware Uncertainty

As summarized in Table VI, three interpretable detectors were evaluated using the existing session split and frozen windows: a validation-calibrated acceleration-peak threshold, a gyroscope-peak threshold, and class-balanced logistic regression on seven statistics. Their event decisions use the same three-positive-window rule and 60-s cooldown as HelmX, with thresholds selected on validation data only. These are offline decision and filter tests; the baselines have no independent Pi timing validation.

Table VI uses the HelmX classifier, rather than the binary OR rule, as its reference because ROC-AUC requires a continuous score and the OR rule differs by only 0.002 in held-out F1. With 10,000 paired *session-clustered* bootstrap draws (five crash-containing and five all-normal sessions resampled separately), baseline-minus-HelmX F1 differences and approximate 95% intervals are: logistic regression -0.181 $[-0.597, -0.035]$; acceleration-peak -0.227 $[-0.770, -0.052]$; and gyroscope-peak -0.283 $[-0.944, +0.342]$. The gyro interval includes zero; with only ten independent sessions, inference remains exploratory. The classifier F1 interval is 0.19–0.95; windows overlap by 75%.

External sets reveal a different tradeoff. On nine scooter crash sessions from the public Sanchez-Iborra *et al.* dataset [26], [27], the HelmX classifier triggered on 6/9 (ROC area .774) versus 9/9 for the acceleration-peak rule (ROC area .838), but the latter generated 14 versus four false-alert events on five normal sessions. On 20 SimRa bicycle near-miss sessions from a publicly released Berlin collection [28], [29], HelmX triggered on 0/20 (ROC area .465), whereas the gyro rule triggered on 17/20 with 355 false alerts on 25 normal sessions. Under leave-DaRUS-out retraining the archived HelmX classifier ROC area is .352, below the chance-level ranking of .5, against .870–.934 for the peak rules; the calibrated simple-rule thresholds nevertheless alert on all 14 non-crash DaRUS sessions. The below-chance classifier ranking is consistent with substantial source/label or placement shift, but does not isolate a causal mechanism. It is evidence against assuming that frozen test-split ranking transfers across sensing domains. The additional STRIDE normal-only evaluation comprises 14 sessions (0.290 h, 4.10 km) disjoint from the nine STRIDE-derived development sessions: HelmX and logistic regression each generated one alert. Because all 23 sessions belong to the same car-recorded source, this is held-out within-dataset evaluation, not an independent external domain. Scooter, SimRa and DaRUS constitute separate-source evaluations. Outside DaRUS, peak-acceleration-derived positive labels partially favor the acceleration-peak baseline.

Logistic regression also preserves session alerts in 8/10, 10/10, and 9/10 at 0.1, 0.3, and 0.4–0.5 Hz, respectively. Other simple rules differ, so no cutoff is declared universally optimal.

TABLE VII
VALIDATION STAGES AND FINDINGS IN THIS STUDY

Check	Recorded endpoint	Finding / boundary
Identity and range	Register reads; rail count	ID 0x70; low-range gyro rails
Raw acquisition	Axis means/SD; packets	Stationary polling/FIFO agreement
Decision transform	Class-stratified F_D	Composed crash-like agreement 90.21%
Continuous stream	V_T, C_D, 90-sample continuity	80.99% coverage; zero I^2C errors
Execution policy	Gaps; processing tails	Priority: faster p99, lower coverage
Transfer checks	Cross-policy events	Domain shift, handling alarms

TABLE VIII
CAUSAL FILTER SENSITIVITY WITHOUT THRESHOLD RETUNING

Filter	Window agreement	Trigger sessions
EMA 0.1 Hz	98.68%	8/10
EMA 0.2 Hz	99.47%	10/10
EMA 0.3 Hz (deployed)	**99.61%**	**10/10**
EMA 0.4 Hz	99.78%	9/10
EMA 0.5 Hz	99.82%	9/10
Causal Butterworth 0.3 Hz	99.61%	10/10

G. Reusable Deployment Assessment Sequence

Table VII summarizes the ordered assessment: verify device and registers; check raw acquisition; quantify fixed-threshold decision transitions; measure temporal validity and decision opportunities; then examine runtime scheduling and transfer. Each step exposes a different failure mechanism. The sequence is diagnostic, not safety certification.

H. Causal-Filter Sensitivity

Table VIII summarizes the archived 60-row frozen-INT8 sensitivity replay, which compares each causal pipeline with the offline zero-phase decision sequence; unlike L2 in Table I, this is not a clean one-factor Keras comparison. Higher window agreement at 0.4–0.5 Hz did not preserve as many session trigger outcomes. The 0.3-Hz EMA is retained because the aim is to characterize the frozen deployment, not post-hoc optimize a detector on the held-out test set.

I. Local Alert Timing and Operational Scope

Ten injected-trace trials included five controls without triggers and five triggered traces. The reported median injected-onset-to-GPIO-call time was 2.382 s (five triggers), while confirmation-to-GPIO-call time was about 0.5 ms. This timing includes accumulation of consecutive analysis windows and is *not* a physical-impact-to-user-notification measurement. The 300 ms stride is a computational scheduling budget, not a safety response guarantee. Electrical power, enclosed-helmet thermal transfer, helmet mechanical dynamics, physically instrumented impact sensitivity, and downstream GPS/GSM delivery were not measured.

V. Discussion and Limitations

The dominant empirical lesson is that stream health, compute feasibility, and model decisions describe *different* properties. In the 90-min polling counterexample, there were no I^2C failures yet $V_T = 93.07\%$ and $C_D = 80.99\%$. The conservative guard invalidates windows after gaps and requires a 60-sample refill; lost decision coverage therefore partly reflects rejection policy, rather than proven sensor sample loss. Host timestamps and kernel interrupt edges do not supply an independently clocked ground truth of all IMU sample instants. Filter and guard-threshold sensitivity, including interpolating short gaps, are retrospective counterfactuals, not established real-time safety mitigations.

Acquisition modes must be compared first in the raw domain under matched register settings, then at the model's operational thresholds. Once the hardware-FIFO packet/configuration checks and equivalence gate were satisfied, the anomalous alert behavior of an earlier incompatible harness did not persist. The randomized headless scheduling experiment likewise cautions against declaring Linux `SCHED_FIFO` universally superior from reduced compute latency alone. The measurable train/serve jerk mismatch had little decision impact and did not cause the six benign-handling alerts; detecting a defect is different from establishing its consequence.

The multi-policy comparison extends *offline* decision-fidelity checks to simpler detectors; physical Pi acquisition was tested with HelmX alone. The benchmark mixes car-mounted phones, simulated driving and motorcycle-source traces, with weak labels, overlapping windows and five crash-containing test sessions. The frozen OR policy's reference window recall is 29/143 (0.203), and its composed-path recall 35/143 (0.245); neither estimates real impact sensitivity. In the benign-motion battery, widening full-scale range reduces observed rail hits but does not prevent handling alarms. The test sources differ markedly in sensor placement, severity, sampling and label mechanism, as the external baselines also demonstrate. No controlled headform impact, reference-IMU transfer test, calibrated anti-alias verification, real-road alert reliability, electrical power measurement, enclosed-helmet thermal validation, or GPS/GSM delivery-reliability trial was performed. The 300-ms processing budget is not an onset-to-notification deadline. Results remain configuration-specific.

Ethics and Study Conduct

Model-development and benchmark event traces were obtained from previously published third-party datasets. The author alone performed this study's original stationary IMU, acquisition, guided helmet-handling, and controlled module/helmet-drop demonstrations; no other individuals were recruited or measured as study participants. The academic supervisor verbally reviewed and endorsed the project activities. No institutional ethics committee application was submitted, and no formal approval or exemption is claimed. Identifiable project media are not included in this article or the public research deposit. These demonstrations are separate from calibrated human-impact or natural-crash validation.

VI. Reproducibility and Data Availability

The unchanged model, normalizer, Pi logs, and evidence index are available in Zenodo v1.0.0 (https://doi.org/10.5281/zenodo.23401265). Version 1.1.0 contains the additional baseline, jerk-isolation, and session-bootstrap scripts, machine-readable outputs, and 5,437 saved predictions [30]. Version 1.1.0 was published on November 2, 2026, at https://doi.org/10.5281/zenodo.23770989. The historical Pi experiments and frozen runtime remain unchanged. Raw third-party datasets are obtained from their original providers under their terms [22], [23], [25], [26], [29]; the v1.1.0 addendum records the source attribution, split boundaries and provenance limitations.

VII. Conclusion

Class-conditional transformation checks, acquisition-equivalence tests and decision-opportunity metrics reveal failure modes invisible to aggregate accuracy, low inference latency, or error-free bus reads. A frozen-policy jerk discrepancy was quantified rather than assumed harmful, and independent offline baseline policies exposed different threshold and external-domain behavior. This study contributes a measurement sequence for accountable edge-sensor deployment, not a validated real-world crash-alert system.

Acknowledgment

The author thanks Tayyaba Tariq for supervision and technical guidance, and Abdul Hanan, Awais Imtiaz, and Hamza Ahmad for hardware integration and project support. OpenAI ChatGPT assisted with prose drafting and LaTeX organization in Sections I–VII and the Supplement; the author reviewed the scientific content and citations.

References

[1] A. Cismas, I. Matei, V. Ciobanu, and G. Casu, "Crash detection using IMU sensors," in *Proc. 21st Int. Conf. Control Syst. Comput. Sci. (CSCS)*, Bucharest, Romania, 2017, pp. 672–676, doi: 10.1109/CSCS.2017.103.

[2] D. Selmanaj, M. Corno, and S. M. Savaresi, "Hazard detection for motorcycles via accelerometers: A self-organizing map approach," *IEEE Trans. Cybern.*, vol. 47, no. 11, pp. 3609–3620, Nov. 2017, doi: 10.1109/TCYB.2016.2573321.

[3] F. Attal, A. Boubezoul, A. Samé, L. Oukhellou, and S. Espié, "Powered two-wheelers critical events detection and recognition using data-driven approaches," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 12, pp. 4011–4022, Dec. 2018, doi: 10.1109/TITS.2018.2797065.

[4] A. J. Mahariba, R. A. Uthra, and G. B. Rajan, "An efficient automatic accident detection system using inertial measurement through machine learning techniques for powered two wheelers," *Expert Syst. Appl.*, vol. 192, Art. no. 116389, 2022, doi: 10.1016/j.eswa.2021.116389.

[5] P. Rodegast, S. Maier, J. Kneifl, and J. Fehr, "On using machine learning algorithms for motorcycle collision detection," *Discover Appl. Sci.*, vol. 6, Art. no. 326, 2024, doi: 10.1007/s42452-024-06014-w.

[6] F. Tabei, B. Askarian, and J. W. Chong, "Accident detection system for bicycle riders," *IEEE Sensors J.*, vol. 21, no. 2, pp. 878–885, 2021, doi: 10.1109/JSEN.2020.3021652.

[7] L. Deng, *Development of a Crash Detection Algorithm for Motorcycle Drivers Using Machine Learning*. Gothenburg, Sweden: Chalmers Univ. Technol., 2021, Master's thesis.

[8] S. A. Mian, *Development of a Smartphone-Based Crash Notification System for Motorcycle Drivers Using Machine Learning*. Gothenburg, Sweden: Chalmers Univ. Technol., 2021, Master's thesis.

[9] V. Viswanath, S. Unnikrishnan, S. Saeed, V. S. Dhanya, P. P. Deepa, and K. Selvan, "Accident detection and reporting system for powered two-wheelers using Internet of Things and Edge AI augmented with MCUSUM features," in *Proc. 2025 Emerging Technol. Intell. Syst. (ETIS)*, 2025, pp. 1–6, doi: 10.1109/ETIS64005.2025.10961751.

[10] K. Moeez Ur Rahman, K. Maaz Ur Rahman, J. Akbar, R. Hamood, A. U. Rehman, and K. H. U. Rahman, "Battery-efficient IoT-enabled smart helmet for real-time motorcycle accident detection and notification," in *2025 IEEE 22nd Int. Conf. Smart Communities: Improving Quality of Life Using AI, Robotics and IoT (HONET)*, 2026, pp. 46–50, doi: 10.1109/HONET67928.2025.11318515.

[11] A. Stisen, H. Blunck, S. Bhattacharya, T. S. Prentow, M. B. Kjærgaard, A. Dey, T. Sonne, and M. M. Jensen, "Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition," in *Proc. 13th ACM Conf. Embedded Networked Sensor Syst. (SenSys)*, 2015, pp. 127–140, doi: 10.1145/2809695.2809718.

[12] S. Attarha and A. Förster, "AssureSense: A framework for enabling sensor fault detection in low-power IoT edge devices," *IEEE Sensors J.*, vol. 24, no. 20, pp. 33791–33805, Oct. 2024, doi: 10.1109/JSEN.2024.3451349.

[13] C. Banbury *et al.*, "MLPerf Tiny benchmark," in *Proc. Neural Inf. Process. Syst. Track Datasets Benchmarks*, vol. 1, 2021.

[14] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, and D. Kalenichenko, "Quantization and training of neural networks for efficient integer-arithmetic-only inference," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 2704–2713, doi: 10.1109/CVPR.2018.00286.

[15] M. Borg, C. Englund, K. Wnuk, B. Duran, C. Levandowski, S. Gao, Y. Tan, H. Kaijser, H. Lönn, and J. Törnqvist, "Safely entering the deep: A review of verification and validation for machine learning and a challenge elicitation in the automotive industry," *J. Automot. Softw. Eng.*, vol. 1, no. 1, pp. 1–19, 2020, doi: 10.2991/jase.d.190131.001.

[16] D. Sculley *et al.*, "Hidden technical debt in machine learning systems," in *Advances in Neural Information Processing Systems*, vol. 28, 2015, pp. 2503–2511.

[17] E. Breck, S. Cai, E. Nielsen, M. Salib, and D. Sculley, "The ML test score: A rubric for ML production readiness and technical debt reduction," in *Proc. IEEE Int. Conf. Big Data*, 2017, pp. 1123–1132, doi: 10.1109/BigData.2017.8258038.

[18] F. Reghenzani, G. Massari, and W. Fornaciari, "The Real-Time Linux Kernel: A Survey on PREEMPT_RT," *ACM Comput. Surv.*, vol. 52, no. 1, Art. no. 18, 2019, doi: 10.1145/3297714.

[19] J. Rehder and R. Siegwart, "Camera/IMU calibration revisited," *IEEE Sensors J.*, vol. 17, no. 11, pp. 3257–3268, Jun. 2017, doi: 10.1109/JSEN.2017.2674307.

[20] InvenSense, "MPU-6500 Register Map and Descriptions," Doc. RM-MPU-6500A-00, Rev. 2.1, Sept. 2013. [Online]. Available: https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6500-Register-Map2.pdf

[21] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. 1st Annu. Conf. Robot Learning*, *Proc. Mach. Learn. Res.*, vol. 78, 2017, pp. 1–16.

[22] A. Khandakar *et al.*, "STRIDE (Sensor Technologies for Road Insights and Driving Evaluation) Dataset," Figshare, 2024, doi: 10.6084/m9.figshare.25460755.

[23] A. Boubezoul, F. Dufour, S. Bouaziz, B. Larnaudie, and S. Espié, "Dataset on powered two wheelers fall and critical events detection," *Data Brief*, vol. 23, Art. no. 103828, 2019, doi: 10.1016/j.dib.2019.103828.

[24] A. Boubezoul, F. Dufour, S. Bouaziz, and S. Espié, "Corrigendum to 'Dataset on powered two wheelers fall and critical events detection'," *Data Brief*, vol. 30, Art. no. 105577, 2020, doi: 10.1016/j.dib.2020.105577.

[25] P. Rodegast, S. Maier, J. Kneifl, and J. Fehr, "Simulation data from motorcycle sensors in operational and crash scenarios," DaRUS, ver. 1, 2023, doi: 10.18419/DARUS-3301.

[26] R. Sanchez-Iborra, L. Bernal-Escobedo, and J. Santa, "Dataset on personal mobility vehicle's regular riding and fall events," *Data Brief*, vol. 61, Art. no. 111681, 2025, doi: 10.1016/j.dib.2025.111681.

[27] R. Sanchez-Iborra, L. Bernal-Escobedo, and J. Santa, "scooter-riding," Zenodo dataset, 2025, doi: 10.5281/zenodo.14887437.

[28] A.-S. Karakaya, J. Hasenburg, and D. Bermbach, "SimRa: Using crowdsourcing to identify near miss hotspots in bicycle traffic," *Pervasive Mobile Comput.*, vol. 67, Art. no. 101197, 2020, doi: 10.1016/j.pmcj.2020.101197.

[29] A.-S. Karakaya and D. Bermbach, "SimRa Rides Berlin 08/23–03/24," TU Berlin, DepositOnce research dataset, 2024, doi: 10.14279/depositonce-20446.

[30] U. Azfar, "HelmX: Reproducibility package for deployment-fidelity validation of a helmet-mounted IMU event-detection policy on Raspberry Pi 4," Zenodo, ver. 1.1.0, 2026, doi: 10.5281/zenodo.23770989.