Silent Collapse: A Distribution-Shift Teardown of a Production Driving Model and a Zero-Retraining Recurrent-State Monitor

Yusuf Guenena

***Abstract*—Production Level-2 driver-assistance stacks are validated largely in simulation, and that practice rests on an unstated assumption: that the shipped model behaves the same on rendered input as on real input, or at least signals when it does not. We test that assumption on one deployed model, openpilot v0.9.7 supercombo, the network that drives comma hardware on public roads. We first build a parity-exact reimplementation of its inference path, verified to within $\pm$0.5 m/s^2 of comma's own reference output on 100% of 1159 real-footage frames (median absolute delta 0.0409 m/s^2), so that any downstream anomaly is attributable to the model and not to our harness. Running the verified model on CARLA-rendered clean roads, we find that it fails silently: 8 of 10 output heads collapse to under 1% of their real-driving temporal activity and the 512-D recurrent feature freezes to about 1×10^{-5} of its real spread, while the model's own predictive-uncertainty heads rise only $1.20\times$ to $1.84\times$ and not one out-of-distribution frame (0 of 219) crosses the real-driving 95th percentile. Nothing the model emits on its exported output channels flags the collapse, even though an internal recurrent signal does carry the OOD information (demonstrated in the monitor experiment below). An alpha-blend sweep characterizes the collapse as a hard cliff on the Subaru source (transition width 0.015), though the cliff shape is segment-dependent (a gradient of width 0.274 on the RAM source), and a layer-by-layer probe localizes it downstream of the vision encoder, in the recurrent summarizer and action-block feedback path, not in perception. We then show that the signal the outputs hide is recoverable from the model's own recurrent feature with a single second-order statistic, the rolling temporal spread of the 512-D state: one O(d) quantity per forward pass, with no retraining and no architecture change. At a sensitivity-matched cross-corpus operating point (threshold set at 95% true-positive rate on the collapse set, identical protocol for every detector), the monitor flags 0 of 1160 held-out real frames: 0% false-positive rate at about 94.8% realised collapse detection across four real corpora, where every location-based baseline fails to transfer (KNN-50 60.8%, Mahalanobis 95.1%, Relative Mahalanobis 99.7%). Calibrated the agnostic, collapse-unaware way instead (1st-percentile-of-real-spread threshold, leave-one-corpus-out over N=4 corpora), it holds about a 2.4% real-driving false-positive rate (segment-level bootstrap 95% CI [0%, 5.2%], 6.9% max; the initial N=2 estimate of 1% was optimistic). Both numbers are true and answer different deploy questions: location detectors separate the collapse but do not calibrate cross-corpus, while the second-order spread monitor does both. It separates the collapse at AUROC 0.996 [0.992, 1.000] and fires about 0.23 blend-units before the outputs cliff, where the location-based feature scores one would default to (Mahalanobis, Relative Mahalanobis, KNN) each hit 100% leave-one-corpus-out FPR and fail to transfer across the real corpora. This is a single-model negative finding with a collapse-specific monitor, not a general OOD detector: an ImageNet-C sweep shows the silent collapse is sim-specific (no corruption reproduces it; at most 1 of 10 heads collapses on any of 75 corruption-severity cells, versus 7 of 10 under CARLA), and the monitor is near chance on most photometric corruptions. Two generalization probes bound the claim rather than broaden it: a second shipped version (v0.9.6) is also out-of-distribution-blind but fails by chaotic output amplification rather than a freeze and the monitor does not transfer to it (33% leave-one-corpus-out FPR), and a real night-and-glare axis at matched intrinsics does not induce the collapse, confirming it is predominantly sim-induced. The contribution is that output-side and location-based signals alone are insufficient for this one shipped model's safety case, and that a second-order recurrent-state monitor is a cheap complement that the present evidence does not claim generalizes.**

***Index Terms*—Out-of-distribution detection, autonomous driving, distribution shift, runtime monitoring, recurrent neural networks, simulation-to-reality gap, safety-critical machine learning.**

I. Introduction

Most Level-2 and autonomous-driving programs validate their driving policy largely in simulation, because simulation is a primary setting in which rare and dangerous scenarios can be exercised at scale and at low cost. The validity of that practice rests on an assumption that is rarely tested directly in the literature we surveyed: that the shipped model under test behaves on rendered input the way it behaves on real input, or, failing that, that it fails loudly enough for the test harness or a downstream safety monitor to notice. If a model can be shown a simulated scene and quietly stop doing its job while every output it emits still looks plausible, then a passing simulation is not evidence that the model works. It is evidence that the model produced a safe-looking default, and the two are indistinguishable from the outside.

We test that assumption on a single deployed model, openpilot v0.9.7 supercombo, the end-to-end network in comma's shipped openpilot driver-assistance system [1], and we find the dangerous answer. On CARLA-rendered clean roads, 8 of 10 of the model's output heads collapse to under 1% of their real-driving temporal activity, and the 512-D recurrent state that threads the model's memory across frames freezes to roughly 1×10^{-5} of its real spread. Yet the model's own predictive-uncertainty heads rise only $1.20\times$ to $1.84\times$, and not one out-of-distribution frame, 0 of 219, exceeds the 95th-percentile uncertainty the model exhibits in real driving. The failure is silent at the exported output level: the planning trajectory, the

Y. Guenena is with the Department of Electrical and Computer Engineering, Wayne State University, Detroit, MI 48202, USA (e-mail: yusuf.a.guenena@gmail.com). Code and data: https://github.com/yusufdxb/supercombo-blindspot

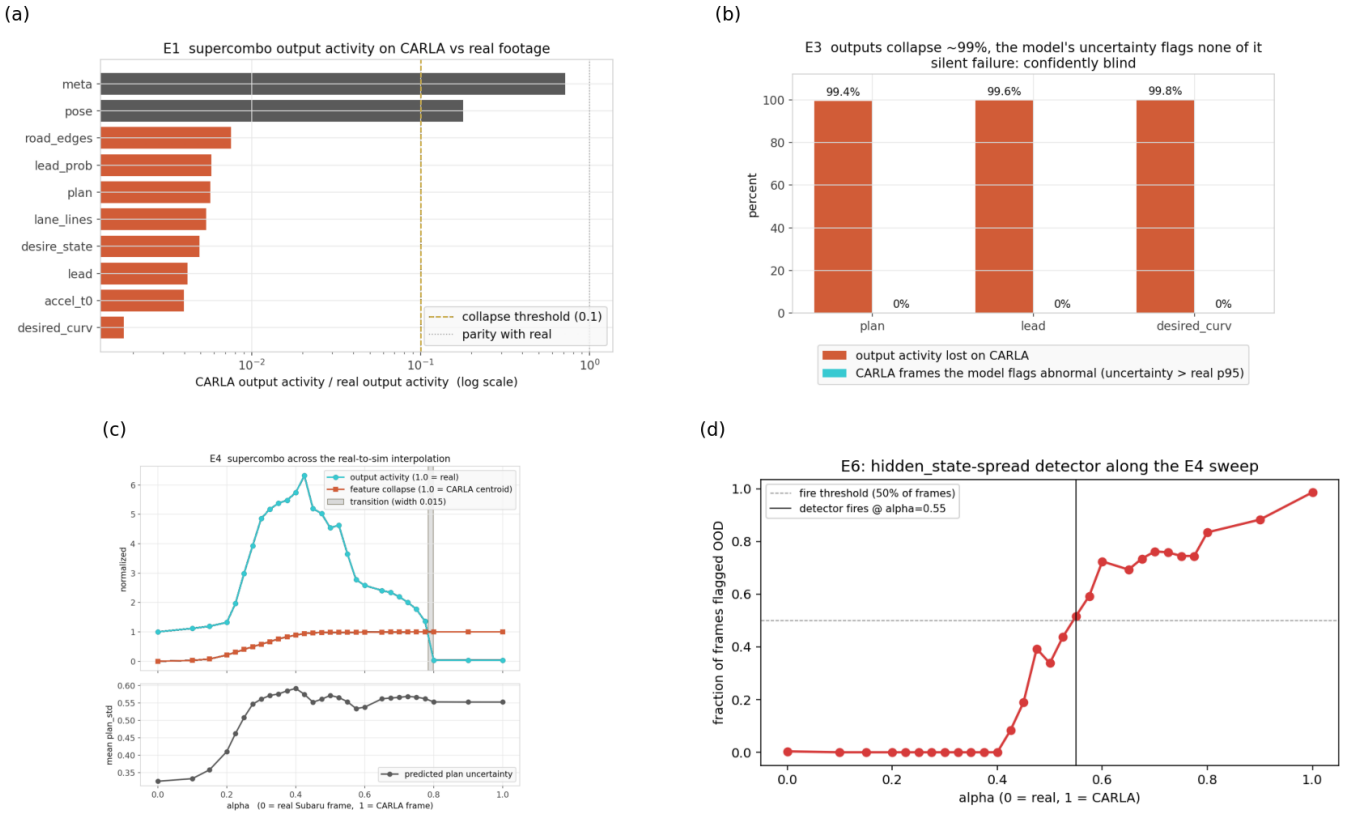

Fig. 1. Four findings at a glance: output collapse (E1), uncertainty silence (E3), the Subaru cliff (E4), and recurrent-state monitor detection (E6).

acceleration command, and the lane and lead geometry all go dark, but the exported uncertainty channel a safety case would watch to catch exactly that event stays quiet. Crucially, this is a property of the model's exported signals, not of the model's internal state: as the monitor experiment (Section 5.6) shows, the recurrent feature does carry an OOD signal, but none of the exported uncertainty heads surface it.

The failure mode this exposes is not hypothetical. Phantom braking under distribution shift, the model commanding a deceleration for an obstacle that is not there, is a known and user-reported failure of the shipped openpilot stack, documented in the project's own issue tracker (commaai issue #20704). We cite that report as motivation only; we make no causal claim linking the silent collapse we measure in simulation to any specific field braking event. The point it grounds is narrower and sufficient: a simulation "pass" can be the model having collapsed to a safe-looking default rather than the model perceiving the scene, and the output-side and uncertainty signals a downstream safety case would trust are precisely the signals that stay silent when it does. That is the gap.

The closest published monitors do not close it. The nearest AV-native neighbor watches the feature density of a frozen perception encoder one stage upstream of where the collapse lives (Keser *et al.* [2]), and the next-nearest watches the latent dynamics of a standalone trajectory predictor with provable changepoint guarantees [3]; neither targets the recurrent state of a shipped end-to-end driver. A recent line in language and vision-language models, EigenTrack [4], does stream a second-order statistic of hidden activations through a trained classifier with early warning, but on LLMs and VLMs, not on a driving model, and not under cross-corpus transfer. And the standard location-based feature-space scores one would reach for first (Mahalanobis, Relative Mahalanobis, KNN) each hit 100% leave-one-corpus-out false-positive rate on this model: calibrated on one real corpus, they flag the entirety of the held-out corpora.

This paper makes four contributions, each bounded to the single model under study. First, a parity-exact reimplementation of openpilot v0.9.7 supercombo inference, verified to within ± 0.5 m/s^2 of comma's reference output on 100% of 1159 real frames (median absolute delta 0.0409 m/s^2), including two non-obvious correctness details (recurrent-state threading and unnormalized YUV input) that a faithful reimplementation must get right. Second, an empirical demonstration of silent failure under visual distribution shift on this model: simultaneous output-head collapse, recurrent-feature freeze, and a non-responsive uncertainty channel, with 0 of 219 OOD frames exceeding the model's real-driving uncertainty p95. Third, a characterization of the collapse as a hard cliff on the Subaru source (transition width 0.015) whose shape is segment-dependent (a gradient of width 0.274 on the RAM source), localized downstream of the vision encoder in the recurrent summarizer and action-block feedback path (a partial localization; an ambiguity in the summarizer's variational bottleneck remains). Fourth, a zero-retraining monitor on the

model's own recurrent feature, the rolling temporal spread of the 512-D state, which at a sensitivity-matched cross-corpus operating point flags 0 of 1160 held-out real frames (0% FPR at about 94.8% realised collapse detection) while every location baseline fails to transfer (KNN-50 60.8%, Mahalanobis 95.1%, Relative Mahalanobis 99.7%), and which under collapse-unaware percentile calibration holds about a 2.4% real-driving FPR leave-one-corpus-out over N=4 corpora (the initial N=2 estimate of 1% was optimistic); it separates the collapse at AUROC 0.996 and fires about 0.23 blend-units before the output cliff, where the location-based scores fail to transfer; bounded by an ImageNet-C sweep that shows the collapse is sim-specific and the monitor collapse-specific, not a universal OOD detector. Figure 1 previews all four findings.

II. Related Work

This contribution sits at the intersection of two ancestral lines. One is location-based feature-space OOD detection, the family that this paper runs as baselines and shows fails to transfer across corpora rather than fails to separate within one. The other is internal-activation runtime monitoring, the family the proposed monitor belongs to, pushed onto the recurrent state of a shipped driver. We name the neighbors a reviewer will invoke and state, for each, exactly what it lacks that this work has. Table RW compresses the comparison onto five axes.

Location-based feature-space OOD (the baseline family). The output-side floor is maximum softmax probability [5] and its energy-based successor [6]; the feature-space ancestor is the Mahalanobis distance-from-fitted-Gaussian-mean score [7], refined for near-OOD by the relative Mahalanobis distance [8], and made non-parametric by deep nearest-neighbor distance [9]. ViM [10] is the modern feature-residual-plus-logit hybrid, and Mahalanobis++ [11] keeps the feature-Gaussian family live in 2025 with an l2-normalization fix, which is why running the Lee 2018 score here is a fair current comparison and not a strawman. OpenOOD [12] codifies this whole line into one taxonomy and codebase; it is the vocabulary anchor for our baselines, not a leaderboard this paper ranks on. We run Mahalanobis, relative Mahalanobis, and KNN-50 as baselines on the same 512-D feature our monitor reads. The honest result, developed in Section 5.6, is that KNN-50 ties our monitor on single-corpus separation (AUROC 1.000); the distinguishing axis is cross-corpus calibration, where all three location-based scores hit 100% leave-one-corpus-out FPR. MSP and Energy are output-side scores on a model whose output channel Section 5.3 shows is silent, and ViM requires a classifier weight matrix that supercombo's multi-head regression outputs do not provide; these three are structurally inapplicable, and we name and excuse them rather than omit them.

AV-native OOD and uncertainty monitoring. The closest published neighbor, Keser *et al.* [2], monitors the feature density of a frozen vision foundation-model encoder as an in-distribution score; that is one stage upstream of our substrate and is exactly the location-based class our baselines instantiate. The next-closest, Guo and Su [3], monitors the latent dynamics of a standalone trajectory predictor as a quickest-changepoint-detection problem with provable bounds on detection delay and false alarm; it differs from this work in model class (a standalone predictor, not a shipped end-to-end driver) and in evidence basis (a theoretical guarantee, not an empirical calibration). Filos *et al.* [13] is the canonical framing of distribution shift for autonomous vehicles, and the SelfOracle line [14] and its uncertainty-quantification successor [15] build misbehaviour predictors on the assumption that the model's confidence signal is informative; Section 5.3 is the contrary finding for this model. A 2025 position paper frames OOD detection explicitly as safety-case evidence [16], which is the niche this work speaks to. None of these targets the recurrent state of a shipped end-to-end driver under cross-corpus transfer.

Internal-activation and second-order monitors (the proposed monitor's family). Runtime neuron activation pattern monitoring [17] is the ancestor of internal-state monitoring: it stores binarized neuron patterns and compares them by Hamming distance, on feed-forward classifiers, a first-order discrete per-frame check. An early move to a higher-order feature statistic for OOD is the Gram-matrix method [18], the closest lineage point to a second-order rather than distance-from-mean choice. The proposed monitor inherits the higher-order-statistic idea from the latter and the internal-monitor idea from the former, but applies them to a recurrent state (neither did) on a shipped production driving model (neither did), and it is the statistic that catches the freeze mode, the recurrent spread crashing across the cliff, that a per-frame pattern or Gram check on outputs would miss. The single closest paper on mechanism is EigenTrack [4], a parallel and near-simultaneous line that streams covariance-spectrum statistics (eigenvalue gaps, spectral entropy, random-matrix-theory features) of hidden activations through a trained recurrent classifier, with early warning, to flag hallucination and OOD drift. It rhymes with this work on three counts (second-order, hidden-state, fires before surface errors), and it differs on three checkable ones: substrate (LLMs and VLMs, not a driving model), statistic (a full eigenspectrum fed to a trained classifier, not a single location-invariant trace with a calibrated threshold), and evaluation (language OOD benchmarks, not cross-corpus leave-one-corpus-out FPR on a recurrent driver state). We therefore do not claim to be first to use a second-order hidden-activation statistic for OOD detection: EigenTrack pre-dates this work on that framing. The available and defensible claim is narrower: this is the first second-order recurrent-state monitor on a shipped end-to-end driving model evaluated under cross-corpus leave-one-corpus-out transfer. NECO [19] builds on a neural-collapse property of classification heads, which supercombo's regression heads lack; we name and excuse it rather than run it.

openpilot and supercombo prior work. The reference academic teardown of supercombo is Openpilot-Deepdive [1], a static input, output, and architecture analysis plus a reimplementation; this paper extends that static teardown to a runtime distribution-shift teardown. The directed-falsification line on openpilot is von Stein and Elbaum [20], which generates adversarial inputs that violate stated properties; it

is related testing work and motivation, not a silent-collapse or recurrent-state-monitor study. A recent adversarial study [21] targets supercombo with deliberate perturbations and input-level defenses, a different failure mode (adversarial attack, not sim-rendered silent collapse) with no recurrent-state monitor.

Simulation testing of driving DNNs and corruption robustness. The DeepRoad line uses metamorphic and generative test synthesis: DeepTest [22] produces transformed driving images to surface erroneous behaviors, DeepRoad [23] uses GANs to render the same scene under new weather and checks for consistent behavior, and MarMot [24] applies metamorphic relations at runtime. These methods generate or transform driving scenes and test the model for consistent behavior, implicitly treating the generated input as a valid scene the model should handle; this paper inverts that premise by showing the simulated input can itself be out of distribution to the model, which undercuts coverage claims built on sim-based testing. ImageNet-C [25] is the corruption-robustness yardstick we use as the bounding OOD axis in Section 5.7, with Cityscapes-C [26] as its AV extension, and CARLA [27] is the simulator that supplies the primary OOD axis.

III. Threat Model

A safety case for a shipped driving model leans on a small set of runtime defenses, and the finding of this paper is that the specific failure mode it documents defeats exactly those defenses, silently and simultaneously. We define the threat precisely, walk through why each standard defense misses it, and position the proposed monitor as a cheap complementary layer rather than a replacement for any of them.

The threat. A shipped driving model is deployed in a visually shifted context: a rendered simulator, an unfamiliar geography, a degraded or unusual sensor condition. Under sufficient shift, three things happen at once and without warning. The output heads collapse to a plausible, nearly constant signal; the recurrent state that carries the model's temporal memory freezes; and the model's own predictive-uncertainty channel does not rise. The danger is not that the model is wrong, which a safety case expects and plans for, but that it is wrong while every signal available on the model's output side says it is fine.

Why uncertainty-head monitoring misses it. The most direct defense is to threshold the model's own predictive uncertainty: if the model says it is unsure, intervene. On this model that defense never fires. Under the collapse, predictive uncertainty rises only $1.20\times$ to $1.84\times$ across the monitored heads, and 0 of 219 OOD frames exceeds the 95th-percentile uncertainty the model exhibits in real driving (Section 5.3). A threshold calibrated on real driving, which is the only honest way to set it, never trips. This directly contradicts the SelfOracle and uncertainty-quantification line, which builds misbehaviour predictors on the premise that the confidence signal carries the information.

Why output-plausibility and temporal-jitter monitors miss it. A second defense checks that the model's outputs are physically plausible: bounded acceleration, feasible curvature, a sane plan. The collapsed outputs are plausible by construction. The plan head retains 0.6% of its real activity and the acceleration head 0.4%, so the model emits a smooth, near-constant trajectory that reads as a benign, stationary scene, and a plausibility check passes it. A third defense watches for temporal jitter or output disagreement as a sign of instability. The freeze produces the opposite of jitter: a frozen output has lower temporal variance than an active one, so a jitter monitor reads the collapse as increased stability, the very thing it is built to reward. (These are structural arguments from the measured activity ratios, not separate experiments.)

Why same-architecture ensembles and input-quality checks miss it. A fourth defense runs an ensemble and flags disagreement. Section 5.5 localizes the collapse downstream of the vision encoder, in a path every instance of the same architecture shares, so an ensemble of the same model would collapse together rather than disagree. A fifth defense screens the input itself for quality. The CARLA-clean renders are typically sharper and less noisy than real road footage, so an image-quality screen would rate the simulated input as good, not anomalous. (Structural arguments, not new experiments.)

The complementary signal. The defenses above all read the output side or the input side. The finding that makes a complement possible is that the model's own recurrent features carry the OOD signal its outputs do not surface. The proposed monitor (Section 5.6) is a complement, not a replacement: it is one O(d) statistic computed from a forward pass that already runs, it requires no retraining and no architecture change, and it is calibrated against a real-driving false-positive rate rather than against simulated negatives. It is also bounded, and we state the bounds here so they travel with the proposal: it is collapse-specific (Section 5.7), it is demonstrated offline only, and its false-positive rate, 2.4% mean across four real corpora (the initial two-corpus estimate of 1% was optimistic), is still not a fleet-scale production number.

IV. Method

This section describes the parity-verified harness, the data, the metrics, the monitor design, and the baseline set in enough detail to trust and reproduce the teardown. The load-bearing content is the parity number and the design descriptions; all figures live in Section 5.

Parity-exact reimplementation. We reconstruct openpilot v0.9.7 supercombo inference from the released ONNX model and comma's own reference files (modeld, the output parser, the YUV loader kernel, and the model constants). Because the central result is a negative finding about a production model, the harness must be trustworthy before any anomaly can be attributed to the model rather than to the reimplementation, so we establish parity first.

Recurrent-state threading. supercombo is a temporal model: it consumes a buffer of recent features and its own previous desired-curvature output, and it must roll that state forward by shift-and-append after each inference, with a zero initialization only on the first frame. A naive per-frame zero reset of the state produces a multi-second initialization transient

that, in this model, looks exactly like a spurious deceleration, a self-inflicted phantom brake of the reimplementation rather than the model. Getting this right is a precondition for both parity and the collapse measurement, because every collapse and corruption sequence is re-run with the same correct state handling.

Unnormalized YUV input. The model's input loader (loadyuv) converts the camera Y, U, and V channels to float with no rescaling: the model consumes uint8 values in the range 0 to 255, not values divided by 255. Dividing by 255, the reflexive normalization, shifts the entire input distribution and silently degrades parity. We feed unnormalized uint8 YUV, matching the kernel.

Parity result. On 1159 real-footage frames (after a 40-frame, 2-second warm-up trim), our reimplemented longitudinal-acceleration output matches comma's logged reference within ± 0.5 m/s^2 on 100.00% of frames, with a median absolute delta of 0.0409 m/s^2, a mean of 0.0541, and a worst-case single-frame delta of 0.2899 m/s^2 (no frame exceeds 0.5). This is the load-bearing harness-trust claim for the negative result, and we report it prominently.

Data. The real in-distribution data are two comma corpora, denoted subaru and ram, of 320 frames each, with the first 100 frames discarded as warm-up. After warm-up and the rolling-window step, 219 analysis frames remain per corpus. The out-of-distribution data are CARLA-rendered clean-road frames from the openpilot v0.9.7 simulation pipeline. The sim camera is configured as a matched-intrinsics pinhole: it renders at the comma 3 fcam native resolution (1928x1208) at the field of view that reproduces fcam's focal length (2648 px), so the rendered frames carry exactly the production camera (`_ar_ox_config.fcam`) intrinsics, and because the sim camera is mounted on the device axes (zero extrinsic calibration), its warp to the model frame reduces to the same intrinsic remap (`K_fcam @ inv(K_medmodel)`) that the real path applies after its `liveCalibration` extrinsics (`src/sim_preprocessor.py`). The intrinsics and the model-input preprocessing are therefore identical to the parity-verified real path, and the distribution shift is confined to rendered image content (photometry, texture, and the absence of sensor noise); the collapse is a response to rendered scene content, not an artifact of a mismatched camera model. The interpolation axis for the cliff analysis (Section 5.4) is a pixel-space alpha-blend of a real sequence (Subaru or RAM) with the CARLA sequence, with alpha=0 the real frame and alpha=1 the CARLA frame, swept over 29 alpha values. We state the underlying counts wherever a percentage appears: for example, the "0%" of Section 5.3 is 0 of 219 CARLA frames. For the threshold-free metrics (Section 5.6), the in-distribution set is subaru and ram concatenated (n=638 stored frames) and the out-of-distribution set is the alpha=1.0 CARLA frames (n=319 stored frames); the rolling-window warm-up leaves 609 and 290 valid (non-NaN) scores respectively, and the threshold-free metrics are computed on those valid subsets.

Metrics. Output activity is the sum of per-element temporal standard deviation over a window; a head is "collapsed" when its CARLA-to-real activity ratio is small. Feature spread is the trace of the recurrent-state covariance over a rolling window. Detection is scored threshold-free with AUROC, AUPR, and FPR at 95% TPR, each with a stratified bootstrap 95% confidence interval (n=1000 iterations, seed 42). Calibration uses a leave-one-corpus-out (LOCO) protocol across the real corpora: a threshold is set on the held-in corpora and its false-positive rate is read on the held-out corpus, averaged over folds. The headline calibration uses four real corpora (subaru, ram, ev6_night, bronco_night), for which we report a segment-level bootstrap 95% confidence interval (resampling whole corpora, not autocorrelated frames). An earlier two-corpus subset (subaru, ram) gave an optimistic 1.03% mean and is identified as the N=2 two-fold estimate wherever it appears. We also report a sensitivity-matched operating point, LOCO FPR@95%TPR, in which the threshold is fixed at 95% true-positive rate on the collapse set under each fold's calibration-corpus ID model and the FPR is then read on the held-out real corpus, applied with an identical protocol to every detector so the cross-corpus comparison is sensitivity-matched and fair.

The monitor (E6). The monitored quantity is the rolling temporal spread of the 512-D recurrent feature the model emits, computed as the trace of the covariance of a 30-frame window of the state. The detection threshold is the 1st percentile of the real-driving rolling-spread distribution, which targets a roughly 1% false-positive rate by construction, and it is calibrated leave-one-corpus-out. The monitor adds one O(d) statistic per forward pass, with no retraining and no extra heads, and its statistic is location-invariant (a second-order spread), which is the property that distinguishes it from the location-based baseline family.

Baselines and structural exclusions. The three applicable post-hoc feature-space scores, Mahalanobis, relative Mahalanobis (RMD), and KNN-50, are computed on the same 512-D feature the monitor reads, with a PCA-Mahalanobis variant as an ablation. RMD's background distribution is fit as a two-component Gaussian mixture, because with a single in-distribution class the Ren et al. marginal Gaussian degenerates to the class Gaussian and the relative score collapses to zero. MSP, Energy, and ViM are structurally inapplicable to supercombo's multi-head Gaussian-mixture regression and existence-probability outputs (there is no softmax head, no logit vector, and no classifier weight matrix), and we state this explicitly so reviewers read it as an exclusion with a reason, not an omission.

V. Experiments and Results

The experiments unfold in the order the argument requires: parity establishes trust (Section 4), collapse establishes the phenomenon (E1), the freeze and the silent uncertainty channel establish the gap (E2, E3), the cliff sweep characterizes the shape (E4), the layer probe localizes the mechanism (E5), the monitor and baselines deliver the solution (E6), and the corruption sweep bounds the scope (E7).

A. E1: Output collapse

We run the parity-verified model on CARLA-rendered clean roads and measure, for each of the 10 output heads,

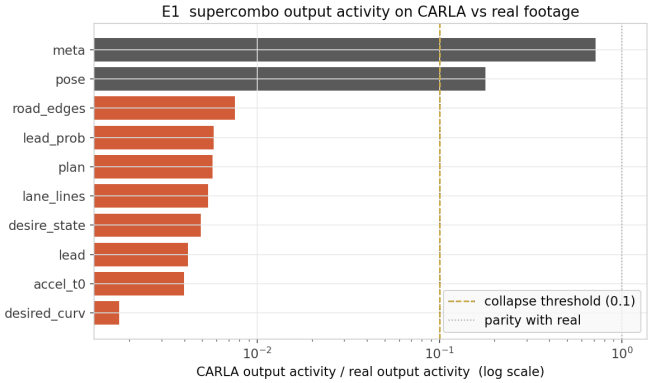

Fig. 2. E1: per-head CARLA-to-real temporal-activity ratio. Eight of ten output heads collapse below 1 percent of real activity; pose and meta survive.

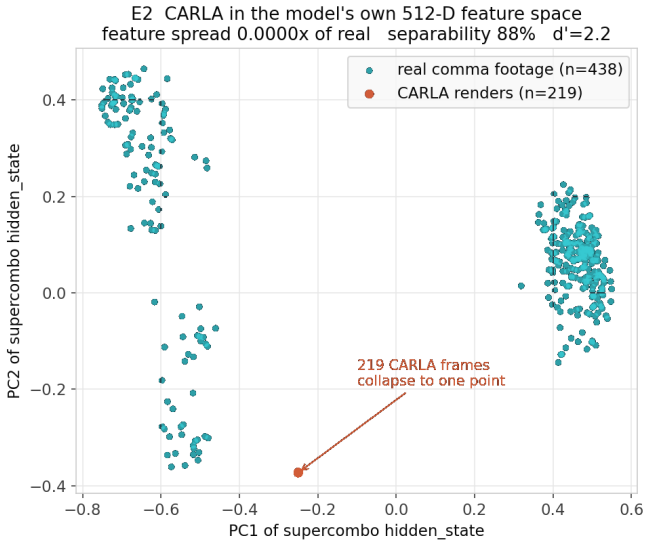

Fig. 3. E2: projected 512-D recurrent feature space. Real-driving states spread out; CARLA states freeze to a single point.

the ratio of its CARLA temporal activity to its real-driving temporal activity. Eight of the ten heads collapse to under 1% of real activity: desired_curv (0.0018), accel_t0 (0.0040), lead (0.0042), desire_state (0.0049), lane_lines (0.0054), plan (0.0057), lead_prob (0.0058), and road_edges (0.0076). These eight are the safety-critical driving signals: the planning trajectory, the acceleration command, the lane and road-edge geometry, the lead-vehicle detection, and the curvature command. Two heads survive: pose (0.1788) and meta (0.7181), the ego-motion and meta-state outputs. The model's primary driving outputs are functionally inactive on CARLA-rendered input while its ego-motion and meta outputs partially persist. Figure E1 plots the per-head ratios.

B. E2: Recurrent-feature freeze

We measure the rolling covariance trace of the 512-D recurrent hidden state on real versus CARLA frames. On CARLA the feature spread falls to about 1×10^{-5} ($0.00001\times$) of its real value: the recurrent state freezes to a near-constant vector.

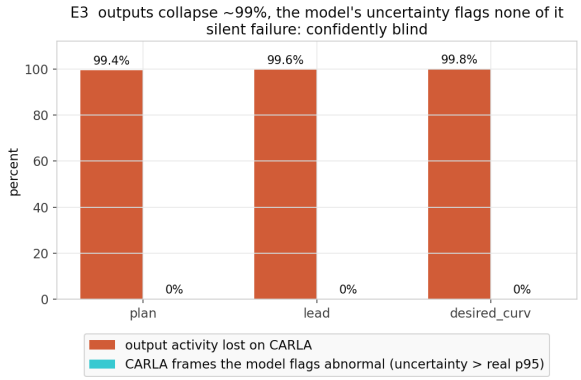

Fig. 4. E3: predictive-uncertainty distributions, real versus CARLA, against the real-driving 95th percentile. Uncertainty barely moves while outputs collapse.

Despite that freeze, the frozen vector is linearly separable from the real-driving states at 87.9% along the centroid-difference direction (d' = 2.19), which establishes that the recurrent state carries a strong out-of-distribution signal even while the output heads are dark. This is the signal the monitor of Section 5.6 reads. It also previews the mechanism that explains the Mahalanobis failure in Section 5.6: the frozen vector lands near the high-density center of the in-distribution Gaussian, so a distance-from-mean score reads it as in-distribution. Figure E2 shows the projected feature space with the CARLA freeze point.

C. E3: Uncertainty silence

This is the safety-relevant centerpiece. For three representative heads we report, side by side, the fraction of output activity retained on CARLA, the ratio of CARLA to real predictive uncertainty, and the fraction of the 219 CARLA frames that exceed the real-driving 95th-percentile uncertainty for that head. The plan head retains 0.6% of its activity while its uncertainty rises $1.35\times$; the lead head retains 0.4% at $1.20\times$; the desired_curv head retains 0.2% at $1.84\times$. In all three cases, 0 of 219 CARLA frames exceeds the real-driving p95. The outputs lose roughly 99.5% of their activity, but the uncertainty channel barely moves and never crosses the threshold a real-calibrated monitor would set.

The implication is the gap this paper turns on. A safety monitor that thresholds the model's own exported uncertainty, calibrated on real driving, never fires under the collapse, because nothing on the model's output side flags it. We state this strictly as an empirical finding about supercombo v0.9.7's exported uncertainty heads under CARLA input: those heads stay quiet. This is not a claim that the model has no internal OOD signal; E6 demonstrates that the recurrent feature does carry such a signal. The claim is the narrower, precise one: the exported uncertainty channel is not a reliable OOD monitor for this collapse. We draw no causal line from this finding to any field phantom-braking incident. Figure E3 plots the uncertainty distributions for real and CARLA frames against the real-driving p95 line, making the silence visually unambiguous.

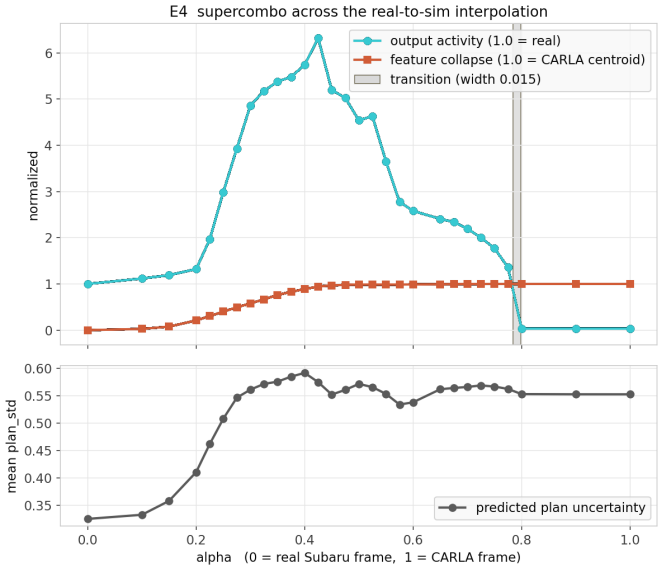

Fig. 5. E4 (Subaru): a hard cliff. Output activity falls from 0.9 to 0.1 of real over a transition width of 0.015.

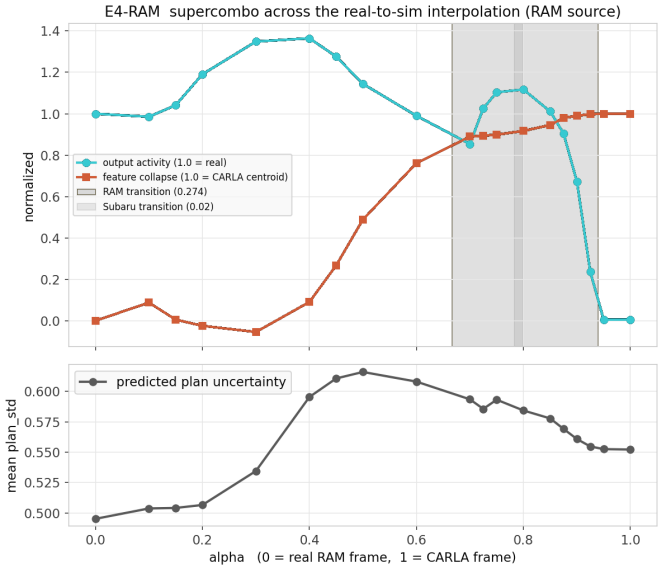

Fig. 6. E4 (RAM): a gradient (transition width 0.274), not a cliff. The cliff shape is segment-dependent.

D. E4: Cliff characterization and segment dependence

To characterize the shape of the collapse, we sweep the pixel-space alpha-blend from the real frame (alpha=0) toward the CARLA frame (alpha=1). On the Subaru source the response is two-phase. Output activity first balloons to $6.32\times$ of the real baseline at alpha=0.425, a thrash driven by the ghosted-input interference of the half-blended frame, and then collapses through a hard cliff, falling from $0.9\times$ to $0.1\times$ of real activity over the narrow alpha band 0.784 to 0.799, a transition width of 0.015. The feature spread crashes from 0.25 to 0.00 by about alpha=0.78. Through the entire transition the predictive uncertainty never spikes, consistent with E3.

The cliff shape is segment-dependent, and this is a bound

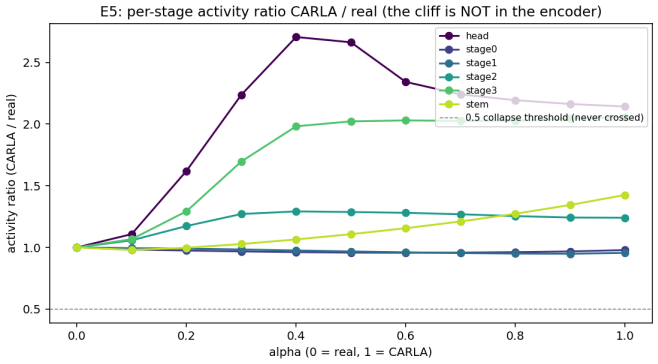

Fig. 7. E5a: per-stage vision-encoder activity ratio. Every encoder stage stays at or above real activity; the collapse is not in perception.

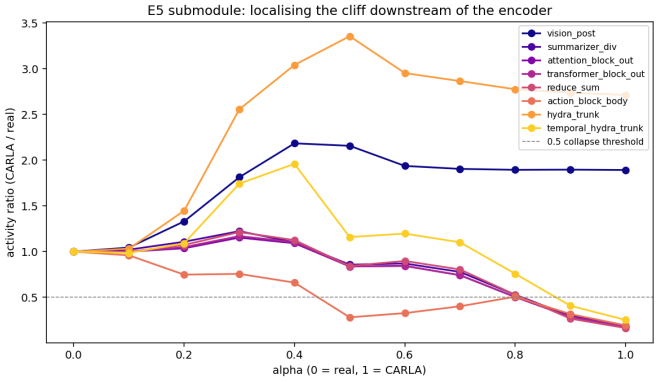

Fig. 8. E5b: per-submodule cliff-alpha. The collapse enters at the recurrent summarizer (cliff 0.900) and the action-block feedback path (cliff 0.500).

on the early-warning headroom the monitor exploits. On the RAM source the same sweep produces a gradient, not a cliff: output activity falls from $0.9\times$ to $0.1\times$ of real over the wide alpha band 0.666 to 0.940, a transition width of 0.274. On the RAM source the monitor's firing point (alpha=0.850) lands inside that band rather than before it, so the early-warning headroom is negative (-0.184) where on Subaru it is positive (0.234). Cliff headroom therefore cannot be assumed to generalize across segments; the characterization is partial. We use the authoritative `report/e4_results.md` and `report/e4_ram_results.md` values (Subaru width 0.015, RAM width 0.274) throughout; where a figure legend rounds these (for example a Subaru legend reading "0.02"), the results-file value governs. Figure E4 places the Subaru cliff and the RAM gradient side by side.

E. E5: Localization downstream of the vision encoder

We first ask whether the collapse is the vision encoder failing, by measuring the CARLA-to-real activity ratio of each encoder stage across the alpha sweep. It is not. Every encoder stage stays at or above real activity across the full sweep: at alpha=1 the stem is at $1.43\times$, stage3 at $2.06\times$, and the head at $2.14\times$, and the minimum ratio over all stages and all alpha is about 0.96. No encoder stage ever crosses the 0.5 collapse threshold. The encoder is in fact more active on

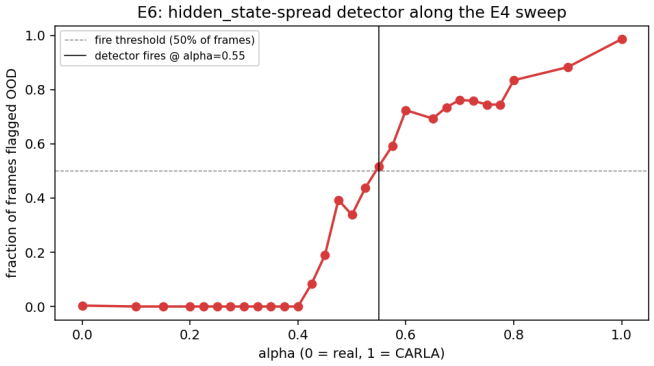

Fig. 9. E6: monitor fired fraction versus alpha. The monitor fires at alpha 0.550, before the output cliff at 0.784.

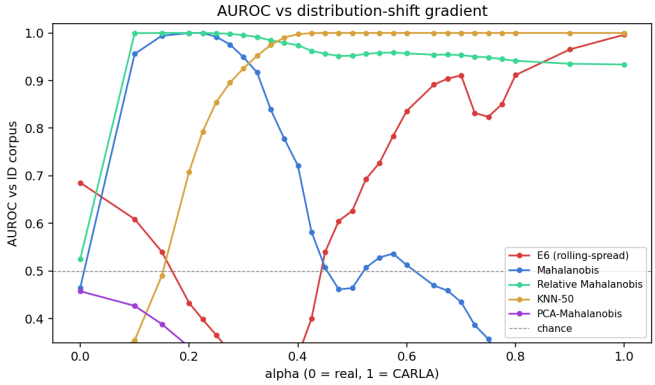

Fig. 10. E6: AUROC versus alpha for all five detectors.

CARLA than on real input, so the collapse seen at the output is not perception failing; it originates downstream. The structural statement is explicit: the failure is in the summarizer and action block, not in the encoder.

We then probe eight tensors between the summarizer and the per-head outputs. Two cross the 0.5 collapse threshold. The recurrent summarizer's variational bottleneck (summarizer_div, which is the hidden_state the monitor reads) has a cliff at alpha=0.900, with its mean shifting to 0.023 of real, a near two-order-of-magnitude collapse of the rolling mean of the 512-D vector; this is the entry point of the collapse. The action-block body (action_block_body), the last residual block before the curvature head, has a cliff a full alpha step earlier, at alpha=0.500, because it folds in the model's own previous desired-curvature output through the recurrent feedback loop, so once that already-collapsing signal joins, the action stack saturates fast. The transformer self-attention, feed-forward, and reduce-sum stages track the summarizer to within 2 to 11% and introduce no additional collapse; they are passive relays. The post-encoder projection (vision_post, $1.89\times$ at alpha=1) and the non-temporal hydra trunk (hydra_trunk, $2.71\times$ at alpha=1) show no cliff, consistent with the two surviving heads of E1. The localization is partial: the summarizer ends in a mu-over-sigma variational reparameterization, and we have not separated the mu path from the sigma normalization, so part of the apparent summarizer collapse could be variance normalization rather than information loss. We state this ambiguity rather than claim a complete mechanistic account. Figures E5a and E5b show the per-stage and per-submodule ratios.

F. E6: Monitor detection and baseline comparison

We now test whether the signal the outputs hide is recoverable from the recurrent feature, and how a second-order monitor of that feature compares to the location-based scores one would default to.

The threshold-free comparison is at alpha=1.0 (full CARLA shift), on the in-distribution set of subaru and ram concatenated (n=638 stored frames) against the CARLA OOD frames (n=319 stored frames), with stratified bootstrap 95% confidence intervals; after the rolling-window warm-up, the metrics are computed on the 609 and 290 valid scores that remain. The rolling-spread monitor reaches AUROC 0.996 [0.992, 1.000], AUPR 0.995 [0.990, 1.000], and FPR at 95% TPR of 0.000. KNN-50 reaches AUROC 1.000 [1.000, 1.000]. We do not claim to beat KNN-50: on single-corpus separation the two tie. Relative Mahalanobis reaches AUROC 0.934 [0.914, 0.952]. Mahalanobis (0.159 [0.130, 0.190]) and PCA-Mahalanobis (0.152 [0.124, 0.179]) both score below chance. Table E6 collects these with their confidence intervals.

Mahalanobis scores below chance for a mechanical reason worth stating, not just reporting. Under the collapse the recurrent state freezes to a near-constant vector that lands near the center of the in-distribution Gaussian (E2), so the OOD frames receive lower Mahalanobis distance than the real frames: a distance-from-mean score cannot detect collapse-to-the-mean, and inverts. PCA-Mahalanobis inherits the same inversion.

The distinguishing axis is not single-corpus separation but cross-corpus calibration, and here the location-based scores fail to transfer. The sharpest way to see this is a sensitivity-matched operating point: fix every detector's threshold at 95% true-positive rate on the collapse set (under each fold's calibration-corpus model) and read its false-positive rate on the held-out real corpus, the identical protocol for all detectors. Under this LOCO FPR@95%TPR, the rolling-spread monitor flags 0 of 1160 held-out real frames: a 0.00% cross-corpus false-positive rate across all four corpora (segment bootstrap CI [0.00%, 0.00%]), at an approximately 94.8% realised collapse-detection rate. The location-based baselines fail at the same operating point: KNN-50 reaches 60.82% LOCO FPR@95%TPR (max 100%), Mahalanobis 95.14%, and Relative Mahalanobis 99.69%. Table E6-OP collects these. We disclose honestly that the approximately 5% of collapse frames the monitor misses at this operating point (15 of 290) are of two kinds: about half are onset and warmup transients, frames where the 30-frame rolling window straddles the real-to-collapse boundary and the spread has not yet crossed; the rest sit at the upper tail of the collapsed-spread distribution, marginally above the threshold (spread near 7.4×10^{-6} against a 7.4×10^{-6} cutoff). The realised true-positive rate is 94.8%, not exactly 95%.

The collapse-unaware calibration tells the complementary

TABLE I
SENSITIVITY-MATCHED CROSS-CORPUS OPERATING POINT (LOCO FPR@95%TPR)

Detector	LOCO mean FPR@95%TPR	95% CI	LOCO max FPR
E6 (rolling spread)	0.00%	[0.00%, 0.00%]	0.00%
KNN-50	60.82%	[35.89%, 85.74%]	100.00%
Mahalanobis	95.14%	[91.77%, 98.51%]	100.00%
Relative Mahalanobis	99.69%	[99.06%, 100.00%]	100.00%

story. Calibrating the threshold the agnostic way (the 1st percentile of the real-driving rolling-spread distribution, no access to the collapse set) and evaluating leave-one-corpus-out, all three applicable location-based baselines (Mahalanobis, relative Mahalanobis, KNN-50) hit 100% LOCO FPR: a threshold calibrated on one corpus flags the entirety of the others. The PCA-Mahalanobis variant improves to 11.91% LOCO mean FPR, a partial recovery but still well above the monitor. The rolling-spread monitor holds 1.03% LOCO mean FPR (max 2.07%) on the subaru-ram pair used for this sweep comparison; scaling the calibration set to four real corpora raises the monitor's cross-corpus FPR to 2.41% mean (max 6.90%, segment-level bootstrap 95% CI [0%, 5.17%]), so its honest agnostic operating point is about 2.4%, still N=4 and not a fleet-scale production number. The two calibrations answer different deployment questions, and both are reported: the collapse-unaware percentile threshold is what you can set with no labeled collapse data, while the sensitivity-matched operating point is what you set once you have a target detection rate, and there the monitor calibrates cross-corpus to 0% while the location baselines do not. The geometric reason the location-based scores fail is that the subaru and ram corpora occupy disjoint regions of the 512-D feature space whose inter-corpus separation dwarfs the within-corpus radius (visible as the two real clusters in Figure E2), so any absolute-position score calibrated on one corpus flags the whole of the other. The monitor reads the second-order trace, which is location-invariant, so it both separates and calibrates across the corpus shift.

Table E6-OP. Sensitivity-matched cross-corpus operating point (LOCO FPR@95%TPR). Threshold fixed at 95% true-positive rate on the collapse set under each fold's calibration-corpus ID model; false-positive rate measured on the held-out real corpus; identical protocol for every detector; N=4 corpora; segment-level bootstrap 95% CI.

E6 flags 0 of 1160 held-out real frames at a realised collapse-detection rate of approximately 94.8%.

The monitor also fires before the outputs cliff. Sweeping alpha, the monitor's fired fraction crosses 50% at alpha=0.550, where the E4 Subaru output cliff is at about alpha=0.784, an early-warning gap of about 0.23 blend-units. (As Section 5.4 notes, this headroom is Subaru-specific; on the RAM gradient the firing point falls inside the transition band.) Figures E6a and E6b show the fire rate and the AUROC trajectory across alpha, and Figure E6c gives the ROC and PR curves at alpha=1.0.

A split-conformal detector built on the same KNN-50 nonconformity score, added for completeness, behaves exactly as the rest of the location-based family: it matches KNN-50 on single-corpus separation (AUROC 1.000) but its leave-one-corpus-out FPR is 100%, since cross-corpus exchangeability fails. Reading detection lead time against single-corpus AUROC makes the distinction concrete. KNN-50 and conformal each reach AUROC 1.000 with an apparent +0.459 blend-unit lead, yet both carry 100% LOCO FPR, so those fire points are meaningless under deployment; the monitor is the only detector with both a calibrated cross-corpus threshold (2.41% LOCO across all four real corpora, 0% at the sensitivity-matched operating point) and a positive lead (+0.234 blend-units). High single-corpus AUROC does not imply useful early warning. Figure E6d plots lead time against AUROC for all detectors.

The honest positioning against the nearest neighbors follows from these numbers. Against the location-based family (Keser-style density, Lee Mahalanobis, Ren RMD, Sun KNN), the result is not that the monitor separates better, since KNN-50 ties it, but that the location-based scores fail to transfer across corpora (100% LOCO FPR, and 60.8% to 99.7% even at a sensitivity-matched operating point) while the second-order monitor calibrates (about 2.4% LOCO over N=4 corpora, and 0% at the sensitivity-matched operating point). Against EigenTrack [4], the nearest mechanism-twin, the deltas are substrate (a shipped driving model, not an LLM or VLM), statistic (a single location-invariant spread trace with a calibrated threshold, not a full eigenspectrum fed to a trained classifier), and evaluation axis (cross-corpus LOCO FPR on real-driving frames, not language OOD benchmarks). We frame the result as transfer and calibration, not as outperforming baselines.

G. E7: Corruption sweep (two-way bound)

The final experiment bounds the contribution in both directions: it bounds the failure mode (is the silent collapse a general model-robustness failure, or sim-specific?) and it bounds the monitor (is the monitor a general OOD detector, or collapse-specific?). We apply the 15 ImageNet-C corruptions at 5 severities each to the real Subaru frames (in raw RGB, before YUV conversion), re-run supercombo with the correct recurrent-state handling on each corrupted sequence, and evaluate the monitor and the baselines, with a validation gate that first reproduces the published CARLA collapse (7 of 10 heads) before any corruption result is read.

The collapse is sim-specific. Running the same E1 output-collapse metric on every corruption-severity cell, no ImageNet-C corruption reproduces the output collapse: 0 of 75 cells reach the collapse criterion (5 or more of 10 heads), and the maximum on any single cell is 1 of 10 heads, against 7 of 10 under CARLA. The silent collapse is a property of full-sim rendering, not of photometric or blur corruptions of real frames. One consequence is immediate: there are zero false negatives, because there is no output collapse anywhere in the sweep for the monitor to miss, so its quiet response on fog, brightness, blur, and the rest is correct.

The monitor is collapse-specific. On most corruptions the monitor sits near chance. Its mean per-corruption AUROC is

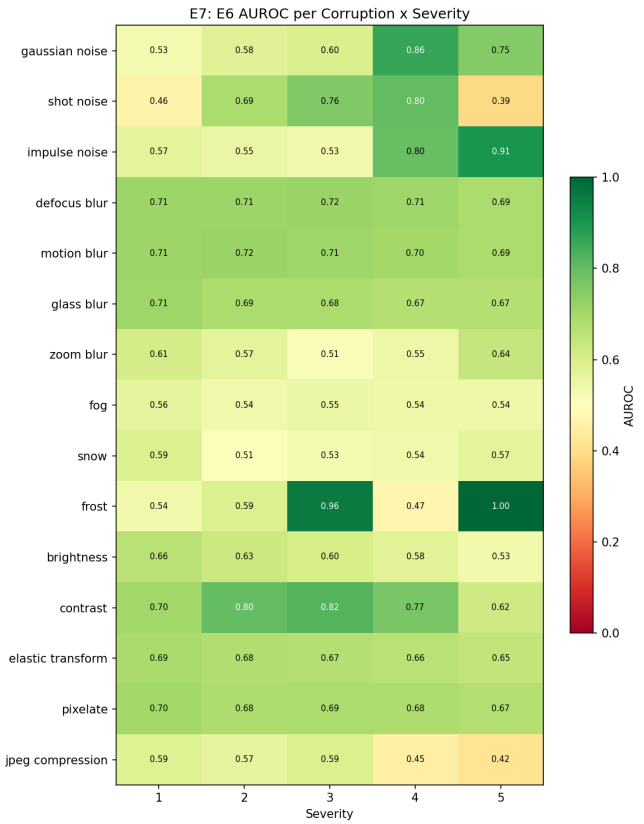

Fig. 11. E7a: monitor AUROC across the 15 by 5 corruption-severity grid.

0.55 on fog, 0.55 on snow, 0.52 on jpeg compression, 0.60 on brightness, and 0.58 on zoom blur, rising only to about 0.68 to 0.71 on the defocus, motion, and glass blur families, all well below the operating regime it holds on CARLA. The monitor fires on only four of the 75 cells at the calibrated threshold: frost severity 3 (AUROC 0.958), frost severity 5 (AUROC 1.000), gaussian-noise severity 4 (AUROC 0.861), and impulse-noise severity 5 (AUROC 0.906). Crucially, all four of those cells have zero output collapse, so on this corpus the monitor's firings track a recurrent-feature-spread shift, not the collapse mode. The monitor is therefore correctly quiet on the real-frame corruptions the model tolerates and is not a universal corruption detector.

Baselines under corruption. The location-based baselines fire on many corruptions at their calibrated thresholds (Mahalanobis and relative Mahalanobis fire at high rates across fog, frost, several noise families, snow, contrast, and zoom), but recall from Section 5.6 that both carry 100% LOCO FPR: they also flag held-out real driving, so their high corruption fire rates are not calibrated detection. KNN-50 fires only on the heaviest noise and frost. None of the baselines calibrates cross-corpus to the operating point the monitor holds (2.41% LOCO over N=4 corpora, 0% at the sensitivity-matched operating point).

Synthesis. The corruption sweep narrows the contribution precisely. The silent collapse, the dangerous mode, is sim-specific; the corruption sweep shows it is not a general model-robustness failure, and the cell-for-cell overlay shows the monitor is not a general corruption detector. The contribution is a targeted monitor for the specific, dangerous, sim-induced silent-collapse mode at about a 2.4% real-driving FPR (N=4 corpora; the initial N=2 estimate of 1% was optimistic), where output-side and location-based feature scores do not detect it. We do not claim the monitor generalizes beyond CARLA as a collapse detector, because no clean non-CARLA output collapse was induced for it to have caught; the one real-segment near-collapse we did find (Section 5.8) has an unexplained trigger. Figures E7a, E7b, and E7c give the AUROC heatmap, the severity sweep, and the cell-for-cell collapse-versus-detection overlay.

H. Generalization probes: a second model and a real adverse-weather axis

Two probes test whether the finding and the monitor extend beyond one model and one synthetic axis. Both bound the contribution rather than broaden it.

A second shipped model (openpilot v0.9.6). We ported the harness to v0.9.6 supercombo, the immediately preceding release, whose input and output contract is identical to v0.9.7 apart from two navigation inputs that are zeroed at the stock no-navigation operating point. Parity holds: v0.9.6 matches comma's own v0.9.6 reference within ± 0.5 m/s^2 on 100% of 560 frames (median absolute delta 0.034), with the frame alignment corroborated by a tight-metric offset sweep rather than by the saturating ± 0.5 criterion alone. On the same CARLA frames, v0.9.6 does not reproduce the silent collapse. Only 1 of 10 output heads collapses, versus 8 of 10 for v0.9.7, and the alpha-blend sweep has the opposite shape: a gradient in which output activity peaks at $14.6\times$ the real baseline near alpha=0.4 and remains $3.3\times$ at full CARLA, rather than cliff-collapsing toward zero. The recurrent feature still separates cleanly from real input (spread ratio 0.44, d-prime 6.8, 100% linear separability) and the uncertainty heads stay silent, so v0.9.6 is also out-of-distribution-blind, but its output-space failure is chaotic amplification rather than a freeze. Calibrated on v0.9.6, the monitor does not transfer across corpora: leave-one-corpus-out mean FPR is 33% on the subaru-ram pair (versus 1.03% for v0.9.7 on the same pair, and 2.41% for v0.9.7 over N=4 corpora). Adjacent shipped versions can therefore fail out-of-distribution in qualitatively different ways, and neither the collapse signature nor the monitor is assumed to carry across them.

A real adverse-weather axis. To separate the collapse from CARLA rendering, we fed three real comma-3 segments at matched fcam intrinsics through v0.9.7: two night segments with oncoming-headlight and tail-light or sign glare, and a daytime-dry control. Real night and glare do not collapse the model. Both night segments collapse 0 of 10 heads (versus 8 of 10 under CARLA) and the monitor fires on 0% of their frames (versus 100% under CARLA), so the silent collapse is predominantly sim-induced and real low-light and glare lie within openpilot's training distribution. One caveat sharpens this without overturning it: the daytime-dry control intermittently enters a near-zero recurrent attractor

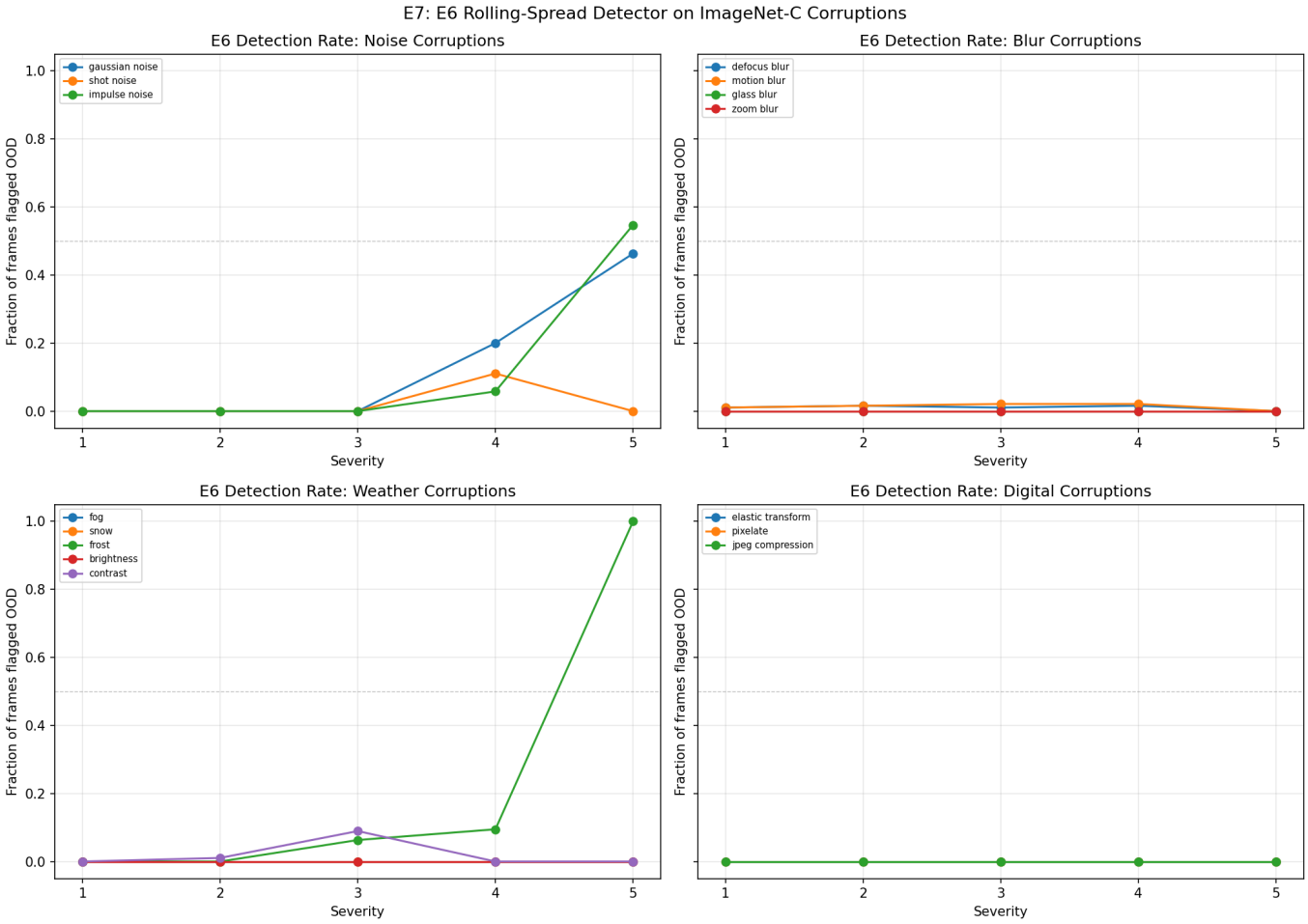

Fig. 12. E7b: monitor fire rate versus severity per corruption family.

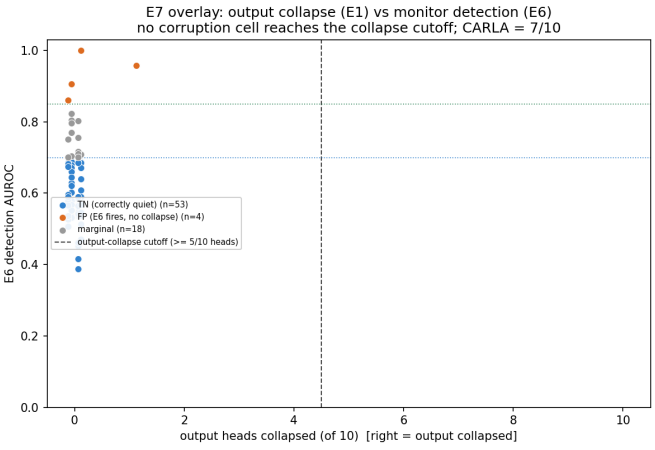

Fig. 13. E7c: cell-for-cell output-collapse count versus monitor AUROC. No corruption cell reaches the collapse cutoff.

that resembles the CARLA collapse, with the output heads suppressed in that regime and the monitor firing on 58% of frames, on clean correctly-warped input. The trigger is unexplained, and an initial steer-and-speed hypothesis was falsified because a night segment reaches a higher peak steer at the same speed without collapsing. A non-CARLA near-collapse therefore exists on real footage and the monitor fires on it, but its cause is open and the night-and-glare axis specifically does not induce the collapse.

VI. Limitations

We state the boundaries that define where the evidence does and does not extend, so that the bounded finding is not mistaken for a general result.

A. Claim taxonomy

The following table classifies every headline claim by its evidential status. The purpose is to separate what the evidence confirms from what it bounds, contradicts, or leaves open, so reviewers can locate each claim precisely.

Models tested: v0.9.7 and v0.9.6. The headline collapse and the monitor are characterized on supercombo v0.9.7. We additionally ran the full teardown on the immediately preceding shipped version, v0.9.6 (Section 5.8), and it does not behave the same: it is also out-of-distribution-blind but fails by chaotic output amplification rather than a freeze, and

TABLE II
CLAIM TAXONOMY BY EVIDENTIAL STATUS

Bucket	Claims
VERIFIED (v0.9.7, CARLA, Subaru/RAM corpora)	E1: 8/10 output heads collapse to under 1% of real activity. E2: recurrent feature is OOD and linearly separable from real at 87.9% ($d' = 2.19$). E3: exported predictive-uncertainty heads rise only 1.20–1.84×; 0/219 CARLA frames exceed real p95, so the exported uncertainty channel is not a reliable OOD monitor for this collapse. E4: collapse arrives as a hard cliff on the Subaru source (transition width 0.015) and as a gradient on the RAM source (width 0.274).
REPLICATED on v0.9.6	v0.9.6 is also out-of-distribution-blind in feature space (100% linear separability, $d' = 6.8$).
CONTRADICTED / DIFFERS on v0.9.6	The silent output freeze does not replicate: only 1/10 heads collapses versus 8/10; the alpha-blend sweep shows chaotic amplification (peaks 14.6× real) rather than a cliff. The E6 monitor does not transfer (33% LOCO mean FPR vs 2.4% on v0.9.7).
MONITOR-ONLY (E6)	The rolling recurrent-spread detector catches the temporal-collapse mode with AUROC 0.996. At a sensitivity-matched cross-corpus operating point (95% TPR on the collapse set) it flags 0 of 1160 held-out real frames (0% LOCO FPR@95%TPR, approximately 94.8% realised detection) while every location baseline fails to transfer (KNN-50 60.8%, Mahalanobis 95.1%, Relative Mahalanobis 99.7%); under collapse-unaware percentile calibration it holds 2.41% cross-corpus LOCO FPR ($N = 4$; the original $N = 2$ subset gave an optimistic 1.03%). E7 shows it is a collapse detector, not a universal OOD detector: photometric corruptions evade it (mean AUROC 0.52–0.74 across corruption types).
DEPLOYMENT-UNSUPPORTED	Scaling the clean-real calibration set from $N = 2$ to $N = 4$ raised the LOCO mean FPR from an optimistic 1.03% to 2.41% (segment-level bootstrap 95% CI [0%, 5.17%], 6.90% max). Fleet-scale FPR is still unproven and likely higher; $N = 4$ is honest progress, not a production number.
HYPOTHESIS / OPEN	A real daytime-dry segment intermittently enters a near-zero recurrent attractor (monitor fires on 58% of frames) on clean correctly-warped input; the trigger is unexplained and an initial steer/speed hypothesis was falsified.

the v0.9.7 monitor does not transfer to it (33% leave-one-corpus-out FPR). Adjacent shipped versions therefore fail out-of-distribution in qualitatively different ways, and neither the collapse signature nor the monitor is claimed to generalize across versions. No Tesla, Mobileye, Waymo, or research imitation-learning stack was tested; whether silent collapse is a property of an architecture, a training recipe, or end-to-end driving models broadly remains the cross-architecture study this paper does not attempt.

N=4 corpora; still not fleet-scale. The cross-corpus calibration now rests on four real corpora (subaru, ram, ev6_night, bronco_night), enough to report a segment-level bootstrap 95% confidence interval. Scaling from the initial two corpora was informative: it raised the LOCO mean FPR from an optimistic 1.03% to 2.41% (95% CI [0%, 5.17%], 6.90% max on the ram fold), confirming that the two-fold estimate understated the cross-corpus false-positive rate. Four corpora is honest progress but not a production number; the held-out FPR remains a calibration estimate, and a far larger and more diverse real corpus is needed before any single fleet-scale FPR can be quoted.

Monitor scope: collapse-specific and offline-only. The corruption overlay (Section 5.7) shows the monitor is collapse-specific and near chance on most real-frame corruptions, so it is not a universal OOD detector. It is demonstrated offline on logged, rendered, and corrupted frames only; no on-road, in-stack, or real-time deployment was run, and no causal link to field incidents was established. We did probe a real adverse-weather axis (Section 5.8): real night and headlight or tail-light glare at matched intrinsics do not induce the collapse, which places the silent-collapse mode predominantly in the synthetic sim-to-real gap rather than in real low-light. One real in-distribution daytime segment does intermittently enter a near-zero recurrent attractor that the monitor fires on, but its trigger is unexplained, so a clean non-CARLA output collapse with a known cause, and real rain footage specifically, remain pending.

Partial localization. The collapse is pinned to the summarizer's variational bottleneck and the action-block feedback path by ruling out the encoder and probing eight submodules, but the mu-versus-sigma ambiguity inside the summarizer's reparameterization is unresolved, so the localization is partial, not a complete mechanistic account.

VII. CONCLUSION

A shipped Level-2 driving model, openpilot v0.9.7 supercombo, shown CARLA-rendered input, collapses to a plausible near-constant while its exported predictive-uncertainty heads do not rise: 8 of 10 output heads fall to under 1% of real activity, the recurrent state freezes to about 1×10^{-5} of its real spread, and 0 of 219 out-of-distribution frames exceeds the model's real-driving uncertainty p95. A simulation "pass" can therefore be the model collapsed to a safe-looking default rather than the model perceiving, and the output-side and exported uncertainty signals a safety case would trust are exactly the ones that stay silent. The model's internal recurrent state does carry an OOD signal (demonstrated in E6), but the exported uncertainty heads do not surface it: the precise failure is that the output-side uncertainty channel is not a reliable OOD monitor for this collapse, not that the model contains no detectable OOD information.

The signal those outputs hide is recoverable from the model's own recurrent feature with a single second-order statistic, the rolling temporal spread of the 512-D state: one O(d) quantity per forward pass, with no retraining and no architecture change. At a sensitivity-matched cross-corpus operating point it flags 0 of 1160 held-out real frames (0% FPR at about 94.8% realised collapse detection across four real corpora) while every location-based baseline fails to transfer (KNN-50 60.8%, Mahalanobis 95.1%, Relative Mahalanobis 99.7%); calibrated the collapse-unaware way instead it holds about a 2.4% real-driving LOCO FPR over N=4 corpora (the

initial N=2 estimate of 1% was optimistic). It separates the collapse at AUROC 0.996 and fires about 0.23 blend-units before the output cliff on the Subaru source, where the location-based feature scores one would default to (Mahalanobis, relative Mahalanobis, KNN) each hit 100% leave-one-corpus-out FPR and fail to transfer across the real corpora.

Output-side monitoring alone is insufficient for the safety case of this shipped driving model, and a second-order recurrent-state monitor is a cheap complement. Its cost is negligible in the loop: a C++ implementation of the rolling-spread statistic matches the reference to within 1×10^{-12} and runs in about 0.4 microseconds per frame on x86 (about 0.0008% of a 20 Hz control budget), with on-device Jetson Orin NX timing left to future hardware validation. This is a collapse-specific, offline-only result: an ImageNet-C sweep shows the silent collapse is sim-specific and the monitor is collapse-specific. Two generalization probes bound rather than broaden it: a second shipped version (v0.9.6) is also out-of-distribution-blind but fails by chaotic amplification rather than a freeze and the monitor does not transfer to it, and a real night-and-glare axis does not induce the collapse. Generalization to other architectures, a clean real-world collapse, and on-vehicle deployment are the next studies, not this one's claims.

VIII. Reproducibility Note

The headline analyses rerun from committed result caches on a fresh public clone without a GPU or CARLA. The committed caches are `report/teardown_collected.npz` (E1, E2, E3), `report/e4_collected.npz` (the E4 Subaru cliff), `report/baselines_collected.npz` (the Mahalanobis, RMD, and KNN-50 baseline scores), and `report/metrics_collected.npz` (the E6 AUROC/AUPR/FPR bootstrap table); the ablations regenerate from those caches as well. The bootstrap is pinned (n=1000, seed=42) and the run environment is pinned in `requirements.txt`. The parity test reruns from the released openpilot v0.9.7 ONNX and comma's logged reference, which are fetched separately and are not redistributed in the repo.

Three supporting result caches are not committed to the public repo because of their size, and the experiments that read them therefore do not rerun from a fresh clone: the E5 submodule cache (`report/e5_submodule_collected.npz`, about 98 MB), the E7 corruption cache (`report/e7_collected.npz`, about 110 MB), and the E4-RAM cache (`report/e4_ram_collected.npz`, about 28 MB). For each, a regeneration path is documented: the corresponding `--collect` pass regenerates the cache from the released ONNX, the source frames, and a GPU, after which the analysis reruns as for the committed experiments. The committed result files (`report/e5_submodule_results.md`, `report/e7_results.md`, `report/e7_overlay_results.md`, `report/e4_ram_results.md`) and the committed figures record the outputs of prior successful collection passes and are readable as-is. The large E5 layer-collection file (`report/e5_collected.npz`, about 3.9 GB) is not committed and is additionally corrupt on the author's machine; it will be replaced by a small committed summary array (per-stage, per-alpha activity ratios), which is all the layer analysis reads. The cache-distribution decision is therefore settled: the four headline caches are committed and reproduce from a fresh clone, and the three smaller supporting caches are regenerated via the documented `--collect` passes rather than shipped in the public repo.

The second-model and real-weather additions (Section 5.8) follow the same pattern. `scripts/fetch_upgrade_data.py` fetches the v0.9.6 ONNX, its comma model-replay reference, and the real comma-3 night, glare, and daytime-control segments, none of which are redistributed in the repo. The v0.9.6 results (`report/teardown_v096_results.md`, `report/e4_v096_results.md`, `report/e6_v096_results.md`, and the parity in `report/parity_v096_results.md`) and the real-weather results (`report/real_weather_results.md`) are committed; the small v0.9.6 teardown cache is committed and the larger collection caches regenerate via the documented `--collect` passes.

References

[1] L. Chen, T. Tang, Z. Cai, Y. Li, P. Wu, H. Li, J. Shi, J. Yan, and Y. Qiao, "Level 2 autonomous driving on a single device: Diving into the devils of openpilot," *arXiv preprint arXiv:2206.08176*, 2022, technical report; no proceedings venue confirmed. [Online]. Available: https://arxiv.org/abs/2206.08176

[2] M. Keser, H. I. Orhan, N. Amini-Naieni, G. Schwalbe, A. Knoll, and M. Rottmann, "Benchmarking vision foundation models for input monitoring in autonomous driving," in *arXiv preprint arXiv:2501.08083*, 2025. [Online]. Available: https://arxiv.org/abs/2501.08083

[3] T. Guo and L. Su, "Latent dynamics-aware ood monitoring for trajectory prediction with provable guarantees," *arXiv preprint arXiv:2603.14603*, 2026, venue unconfirmed; pin at camera-ready. [Online]. Available: https://arxiv.org/abs/2603.14603

[4] F. Ettori, S. Darabi *et al.*, "EigenTrack: Spectral activation feature tracking for hallucination and out-of-distribution detection in llms and vlms," in *arXiv preprint arXiv:2509.15735*, 2025. [Online]. Available: https://arxiv.org/abs/2509.15735

[5] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," in *International Conference on Learning Representations (ICLR)*, 2017. [Online]. Available: https://arxiv.org/abs/1610.02136

[6] W. Liu, X. Wang, J. D. Owens, and Y. Li, "Energy-based out-of-distribution detection," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. [Online]. Available: https://arxiv.org/abs/2010.03759

[7] K. Lee, K. Lee, H. Lee, and J. Shin, "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. [Online]. Available: https://arxiv.org/abs/1807.03888

[8] J. Ren, S. Fort, J. Liu, A. G. Roy, S. Padhy, and B. Lakshminarayanan, "A simple fix to mahalanobis distance for improving near-ood detection," *arXiv preprint arXiv:2106.09022*, 2021, venue unconfirmed; pin at camera-ready. [Online]. Available: https://arxiv.org/abs/2106.09022

[9] Y. Sun, Y. Ming, X. Zhu, and Y. Li, "Out-of-distribution detection with deep nearest neighbors," in *Proceedings of the 39th International Conference on Machine Learning (ICML)*, 2022. [Online]. Available: https://arxiv.org/abs/2204.06507

[10] H. Wang, Z. Li, L. Feng, and W. Zhang, "ViM: Out-of-distribution with virtual-logit matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. [Online]. Available: https://arxiv.org/abs/2203.10807

[11] M. Mueller and M. Hein, "Mahalanobis++: Improving ood detection via feature normalization," in *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025, venue unconfirmed; pin at camera-ready. [Online]. Available: https://arxiv.org/abs/2505.18032

[12] J. Yang, P. Wang, D. Zou, Z. Zhou, K. Ding, W. Peng, H. Wang, G. Chen, B. Li, Y. Sun, X. Du, K. Zhou, W. Zhang, D. Hendrycks, Y. Li, and Z. Liu, "OpenOOD: Benchmarking generalized out-of-distribution detection," in *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2022. [Online]. Available: https://arxiv.org/abs/2210.07242

[13] A. Filos, P. Tigas, R. McAllister, N. Rhinehart, S. Levine, and Y. Gal, "Can autonomous vehicles identify, recover from, and adapt to distribution shifts?" in *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020. [Online]. Available: https://arxiv.org/abs/2006.14911

[14] A. Stocco, M. Weiss, M. Calzana, and P. Tonella, "Misbehaviour prediction for autonomous driving systems," in *Proceedings of the 42nd International Conference on Software Engineering (ICSE)*, 2020. [Online]. Available: https://arxiv.org/abs/1910.04443

[15] R. Grewal, P. Tonella, and A. Stocco, "Predicting safety misbehaviours in autonomous driving systems using uncertainty quantification," in *Proceedings of the 17th IEEE International Conference on Software Testing, Verification and Validation (ICST)*, 2024. [Online]. Available: https://arxiv.org/abs/2404.18573

[16] V. J. Hodge, C. Paterson, and I. Habli, "Out-of-distribution detection for safety assurance of ai and autonomous systems," *arXiv preprint arXiv:2510.21254*, 2025, venue unconfirmed; pin at camera-ready. [Online]. Available: https://arxiv.org/abs/2510.21254

[17] C.-H. Cheng, G. Nührenberg, and H. Yasuoka, "Runtime monitoring neuron activation patterns," in *Design, Automation and Test in Europe Conference (DATE)*, 2019, pp. 300–303, arXiv preprint 1809.06573 dated 2018; published at DATE 2019. [Online]. Available: https://arxiv.org/abs/1809.06573

[18] C. S. Sastry and S. Oore, "Detecting out-of-distribution examples with gram matrices," in *Proceedings of the 37th International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 119, 2020, pp. 8491–8501. [Online]. Available: https://proceedings.mlr.press/v119/sastry20a.html

[19] M. Ben Ammar, N. Belkhir, S. Popescu, A. Manzanera, and G. Franchi, "NECO: Neural collapse based out-of-distribution detection," in *International Conference on Learning Representations (ICLR)*, 2024. [Online]. Available: https://arxiv.org/abs/2310.06823

[20] M. von Stein and S. Elbaum, "Finding property violations through network falsification: Challenges, adaptations and lessons learned from openpilot," in *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering (ASE '22), Industry Showcase*, Rochester, MI, USA, 2022.

[21] C. Chen, Y. Wang, N. S. Munir, X. Zhou, and X. Zhou, "Revisiting adversarial perception attacks and defense methods on autonomous driving systems," *arXiv preprint arXiv:2505.11532*, 2025. [Online]. Available: https://arxiv.org/abs/2505.11532

[22] Y. Tian, K. Pei, S. Jana, and B. Ray, "DeepTest: Automated testing of deep-neural-network-driven autonomous cars," in *Proceedings of the 40th International Conference on Software Engineering (ICSE)*, 2018. [Online]. Available: https://arxiv.org/abs/1708.08559

[23] M. Zhang, Y. Zhang, L. Zhang, C. Liu, and S. Khurshid, "DeepRoad: GAN-based metamorphic autonomous driving system testing," in *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering (ASE)*, 2018. [Online]. Available: https://arxiv.org/abs/1802.02295

[24] J. Ayerdi, A. Iriarte, P. Valle, I. Roman, M. Illarramendi, and A. Arrieta, "MarMot: Metamorphic runtime monitoring of autonomous driving systems," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 2024. [Online]. Available: https://arxiv.org/abs/2310.07414

[25] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," in *International Conference on Learning Representations (ICLR)*, 2019. [Online]. Available: https://arxiv.org/abs/1903.12261

[26] C. Michaelis, B. Mitzkus, R. Geirhos, E. Rusak, O. Bringmann, A. S. Ecker, M. Bethge, and W. Brendel, "Benchmarking robustness in object detection: Autonomous driving when winter is coming," *arXiv preprint arXiv:1907.07484*, 2019, venue unconfirmed; pin at camera-ready. [Online]. Available: https://arxiv.org/abs/1907.07484

[27] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning (CoRL)*, 2017. [Online]. Available: https://arxiv.org/abs/1711.03938