Technical report · IF-TR-2026-03 · version 1.0

Monocular range and TTC for a warning-only phone ADAS: v1 false warnings and the v2 chain in simulation

Sherin Joseph Roy 10 November 2026 Lyne ADAS v2.0.0 P1 · Perception Funded entirely by BluRabbit Mobility Private Limited.

Abstract

Lyne ADAS is a warning-only (Level 1) driver assistance app that runs object detection, range estimation and collision warning on an Android phone mounted on the windshield. Its first release estimated range from bounding-box width alone, which turned detector jitter into range noise and phantom forward-collision warnings. Release v2.0.0 replaces that chain with ground-plane geometry fused with a learned per-object width prior in inverse depth, a per-object Kalman filter, a looming-rate time-to-collision estimate with significance gates, time-based alert gating and online calibration of camera height. This report describes the design and reproduces the repository's JVM simulation suite at the release commit, comparing a v1 baseline with v2 on false warnings, range error and jitter, alarm timing in hard-braking and gentle-approach scenarios, and pedestrian cases. Three figures stated in the README differ from the author's run and are reported side by side. All results come from synthetic drives: no on-road accuracy measurement is published for this release, and no performance claim is made for any phone.

1 Problem

Lyne ADAS is a warning-only (Level 1) driver assistance app for Android. A phone on the windshield warns about vehicles the driver is closing on, pedestrians and riders in the path, tailgating, lane drift and speed limits. The app never controls the vehicle. The forward-collision warning (FCW) depends on two quantities from a single camera: range to the lead vehicle and time to collision (TTC).

The v1 releases estimated distance from detection-box width alone. The repository's replica of the v1 path (`LegacyFcw`) assumes one vehicle width for every class, smooths box growth frame to frame, takes TTC from differenced distance, debounces over two frames and emits one event per frame. The README records the outcome: a v1 drive logged against a car's radar showed large range errors, and most collision warnings did not match a real hazard. Its diagnosis is that a pixel of detector jitter became metres of range noise and then phantom warnings, and that an SUV "looked" closer than a hatchback. The README's figure for that drive is not repeated here, because no log of it is published.

For a warning-only system, false warnings are the dominant failure: a driver who hears alarms without a hazard learns to ignore them. The question for v2 is whether range becomes accurate and stable, and whether warnings fire only for statistically real closing while still firing early in hard braking.

The README also observes that the general-purpose detector sometimes labels auto-rickshaws as buses and riders as pedestrians, which the roadmap's Indian-roads detector is meant to address.

2 System design

Per the README and `ARCHITECTURE.md`, camera frames go to a single inference thread running a YOLOv8n INT8 detector, TwinLiteNet drivable-area segmentation (classical segmentation on the lowest tier) and a GTSRB-trained sign classifier through LiteRT. Detections pass to a tracker, the range and TTC chain in `fusion/`, risk logic, alert gates, and then to tones, voice, the HUD and logs. The fusion package has no Android types, so the JVM test suite replays synthetic drives through the code that runs on the phone.

Two range cues in inverse depth

`RangeEstimator` fuses two cues by inverse variance in inverse depth, q = 1/d. The ground-plane cue uses the depression angle of the box bottom and the camera height, so it does not depend on object size. The width cue uses box width and a real-width prior: a per-class value, or the width this object was measured to have while close (learned in `ObjectTracker` from the ground cue). A truncated box drops the affected cue, and cues that disagree beyond their combined uncertainty inflate the fused sigma. The source's reason for inverse depth is that both cues are linear in pixels in q, so noise there is Gaussian and value-independent, whereas weighting metres by their own error biases distances short.

Kalman filter and looming-rate TTC

Each track carries `RangeKalman`, a constant-velocity filter on (q, q′) that receives only the random part of the measurement noise and gates outliers. Since q′/q is the inverse TTC, TTC follows without differencing noisy frames, and the HUD shows range with its uncertainty. `ScaleRateEstimator` fits a recency-weighted least-squares line to log box scale over a short window and keeps its standard error. `FusionEngine` fuses both in inverse-TTC space and treats closing as real only at two standard deviations or more.

Alert gates and calibration

A caution needs a mature track and significant closing. A critical (brake) alarm also needs three standard deviations, a minimum closing speed, sufficient range confidence and physical plausibility (closing no faster than own speed plus a margin); low-confidence estimates are downgraded, and FCW is disarmed below a minimum own speed. `AlertGate` accumulates evidence in wall-clock time with a release delay and re-arm cooldown, so one incident yields one event at any frame rate. In-path decisions use a metric corridor curved by gyro yaw rate.

Mount pitch and roll come from the gravity sensor, gated against acceleration. Camera height starts from a vehicle preset; `HeightAutoCalibrator` histograms the ratio of width-prior distance to ground-plane distance over confirmed cars and nudges a bounded height multiplier toward the median.

A capability probe selects one of three device tiers. The README lists their frame rates as targets, not measurements.

3 Method

The author checked out tag v2.0.0 (commit a68060b) and ran the JVM unit tests (`:app:testDebugUnitTest`), with no device or emulator. `AccuracyTest` prints its figures to standard output, captured in the JUnit XML report; every value below is read from that output.

In `DriveSim.kt`, a `Projector` places simulated vehicles and pedestrians in road coordinates and projects them through a true pinhole camera. It adds Gaussian noise to each box edge, drops a fraction of detections and, in some scenarios, adds pitch bounce; the app's geometry can be built from deliberately wrong height and pitch. Synthetic boxes go directly into `FusionEngine`; the neural detector is not run.

The v1 baseline is `LegacyFcw`, a test-code replica of the v1 FCW path, fed the same noisy detections in the same run; it is not the v1 binary. The range-error test gives it the correct field of view, and the jitter and soak tests the projector's default. In the soak, v1 warnings are counted on onset; in steady following, once per frame on which v1 would emit, matching v1's per-frame behaviour. v2 is counted as emitted FCW events. Alarm timing is the ground-truth TTC (true gap over true closing speed) at the first frame the alarm level is reached.

The author also ran the self-test of the field validation script `tools/validate_accuracy.py`, which checks that its analysis recovers known errors injected into synthetic logs.

4 Validation

The suite reported `21 tests`, with no failures or skips.

False warnings

In the 20-minute random-traffic soak (a lead whose gap wanders slowly, plus an adjacent-lane car), v2 emitted `0 warnings` and the v1 baseline `626 warnings`. The minimum ground-truth TTC in the scenario was `12.4 s`, well above the app's caution threshold, so none of these warnings is justified.



Figure 1. False forward-collision warnings in the 20-minute simulated random-traffic soak, v1 counted on onset; JVM suite at v2.0.0; reproduced by the author.

In two minutes of steady following at 30 m, the v1 baseline produced `10 events` at 8 fps, `112 events` at 15 fps and `514 events` at 30 fps; v2 produced `0 events` at each rate. Because v1 is counted per frame here, part of its growth with frame rate reflects per-frame emission rather than more incidents.

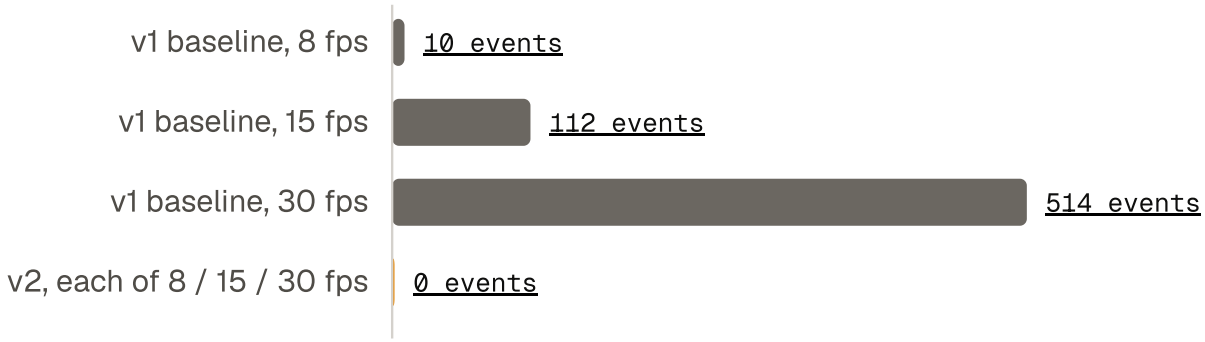


Figure 2. False FCW events in two minutes of simulated steady following at 30 m; v1 counted per frame, v2 per emitted event; JVM suite at v2.0.0; reproduced by the author.

Range error and jitter

Across a hatchback, sedan and pickup at seven distances, with the app's height and pitch slightly wrong, mean absolute range error was `2.07 m` for v2 and `7.9 m` for the v1 baseline. The README states `2 m` for v2; the two differ, and the README value is lower. For a static lead at 30 m, the standard deviation of reported range was `1.09 m` for v2 and `5 m` for v1. The README states `1 m` for v2; again the two differ and the README value is lower. The v1 figures match the README.

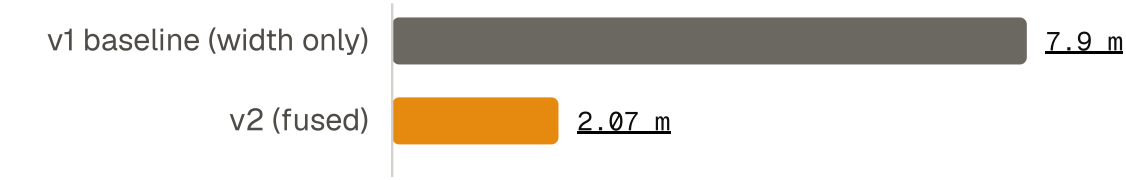


Figure 3. Mean absolute range error, three vehicle widths at 8 to 60 m, app calibrated with small height and pitch errors; JVM suite at v2.0.0; reproduced by the author. The README states a different v2 value (Table 2).

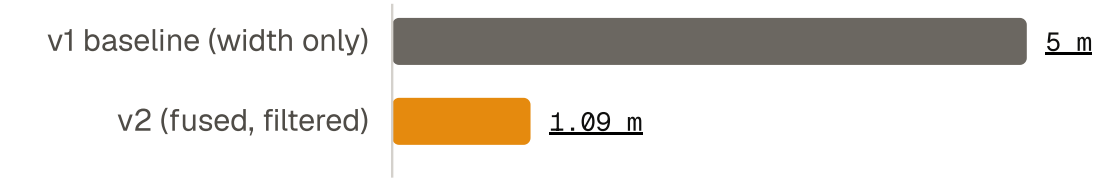


Figure 4. Standard deviation of reported range, static lead at 30 m; JVM suite at v2.0.0; reproduced by the author. The README states a different v2 value (Table 2).

Alarm timing

When the lead brakes hard from 35 m, the real TTC at first caution over eight noise seeds was `1.4068645, 1.8833354, 1.7559159, 1.4068645, 1.5181836, 1.8833354, 1.0981997, 1.0981997 s`, and at first critical alarm `0.8208347, 0.8208347, 1.3000017, 0.8208347, 0.8208347, 1.002633, 1.0981997, 1.002633 s`. Both test assertions passed (every seed reaches critical with margin; median caution above a floor). The README states the median critical alarm as `≈0.8 s`. That is close to the smallest per-seed value; several seeds fired at that value and the rest later in real TTC, so the README median and the recorded values differ. Here the README understates the warning margin. Its median caution, `≈1.5 s`, is approximate; the per-seed values are the measured record.

In a gentle approach, caution came at a real TTC of `2.0666754 s` (README: `2.1 s`). With one truck reported as both truck and bus every frame, caution fired at `1.7571429 s`; the test's comment notes that it does not reproduce the lead flipping seen on device.

Quantity	Value	Conditions	Ba
Real TTC at first FCW caution, lead brakes 6 m/s^2 (per seed)	`1.4068645, 1.8833354, 1.7559159, 1.4068645, 1.5181836, 1.8833354, 1.0981997, 1.0981997 s`	8 noise seeds; lead at 35 m, both at 22 m/s, lead brakes at 6 m/s^2 from t=1.5 s, 15 fps; test asserts sorted[4] > 1.2 s	Re 20
Real TTC at first FCW critical (brake) alarm, lead brakes 6 m/s^2 (per seed)	`0.8208347, 0.8208347, 1.3000017, 0.8208347, 0.8208347, 1.002633, 1.0981997, 1.002633 s`	same 8 seeds and scenario; test asserts every seed fires with real TTC > 0.5 s Differs from another source; both are listed.	Re 20
Real TTC at FCW caution, gentle approach	`2.0666754 s`	closing 5 m/s from 50 m, ego 20 m/s, 15 fps, seed 4; test asserts > 1.8 s	Re 20
Real TTC at FCW caution with duplicate truck/bus detections	`1.7571429 s`	truck at 30 m reported as both truck and bus each frame, lead brakes 5 m/s^2, 10 Hz; test asserts > 0.8 s	Re 20
Real time-to-reach at VRU critical, pedestrian in path	`1 s`	pedestrian from 35 m, ego 10 m/s (36 km/h), 8 fps; test asserts > 0.8 s	Re 20

Quantity	Value	Conditions	Bas
VRU critical events, pedestrian at constant 8 m	`0 events`	10 s at 8 fps, ego speed input 14 m/s, pedestrian gap held at 8 m; test asserts 0	Re 20
Track ids used for one car crossing the frame at 7 fps	`1 track ids`	car moving 3.5 m/s laterally from x=-5 m at 12 m, 3 s at 7 fps; test asserts <= 1	Re 20
Camera height learned by auto-calibration	`1.42 m`	configured 1.20 m, true 1.45 m, 40 random lead cars x 4 s at 15 fps; test asserts |learned - 1.45| < 0.08 m	Re 20
Minimum true TTC in the 20-min soak scenario	`12.4 s`	ground-truth TTC of the simulated lead; test asserts > 4 s	Re 20

`Table 1.` Table 1. Alarm timing, pedestrian, tracking and auto-calibration scenarios for v2; synthetic drives in the JVM suite at v2.0.0; reproduced by the author. “Reproduced” rows were run by the author; “Reported” rows are quoted from the repository at the commit shown.

Pedestrians, tracking and auto-calibration

With a pedestrian holding a constant gap ahead of a moving ego vehicle, the count of critical pedestrian warnings was `0 events`; the warning uses measured closing speed, not own speed. A pedestrian standing in the path at an approach of 36 km/h reached critical at a real time-to-reach of `1 s`. For a car crossing the frame at 7 fps, the number of track identities used was `1 track ids`. From a wrongly configured camera height, auto-calibration over simulated traffic learned `1.42 m`, matching the README (`1.42 m`).

README figures against the run

Table 2 lists the README’s “Measured, reproducibly” table. The soak, steady-following, v1 range and auto-calibration figures agree with the run; v2 range MAE, v2 jitter and the hard-brake median critical alarm do not. The repository does not show how the README figures were obtained.

Quantity	Value	Conditions	Basis
False collision warnings, 20 min random traffic (v1)	626 warnings	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce; true TTC never under 12 s	Reported 2026-11-08
False collision warnings, 20 min random traffic (v2)	0 warnings	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce; true TTC never under 12 s	Reported 2026-11-08
False warnings, 2 min steady following at 30 m (v1, 8 / 15 / 30 fps)	10 / 112 / 514 warnings	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce	Reported 2026-11-08
False warnings, 2 min steady following at 30 m (v2, 8 / 15 / 30 fps)	0 / 0 / 0 warnings	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce	Reported 2026-11-08
Mean range error, hatchback / sedan / pickup at 8 to 60 m (v1)	7.9 m	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce; with 5 cm height and 0.3 deg pitch calibration error	Reported 2026-11-08
Mean range error, hatchback / sedan / pickup at 8 to 60 m (v2)	2 m	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce; with 5 cm height and 0.3 deg pitch calibration error Differs from another source; both are listed.	Reported 2026-11-08
Distance jitter at 30 m, std dev (v1)	5 m	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce	Reported 2026-11-08
Distance jitter at 30 m, std dev (v2)	1 m	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce Differs from another source; both are listed.	Reported 2026-11-08
Real TTC at caution, lead brakes 6 m/s^2 from 35 m at 80 km/h (median of 8 runs)	≈1.5 s	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce	Reported 2026-11-08
Real TTC at brake alarm, lead brakes 6 m/s^2 from 35 m at 80 km/h (median of 8 runs)	≈0.8 s	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce Differs from another source; both are listed.	Reported 2026-11-08
Caution at real TTC, gentle approach closing at 18 km/h	2.1 s	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce	Reported 2026-11-08
Camera height learned by auto-calibration (configured 1.20 m, true 1.45 m)	1.42 m	README 'Measured, reproducibly' table; synthetic drives, 1.3 px per box edge at 256 px input, 5% missed detections, road bounce	Reported 2026-11-08

Table 2. Table 2. Figures stated in the README for the same simulation suite; reported in the repository at v2.0.0. “Reproduced” rows were run by the author; “Reported” rows are quoted from the repository at the commit shown.

Field validation tooling

The self-test of `tools/validate_accuracy.py` passed `4 checks`, recovering a static scale ratio of `1.048 ratio`, a clock offset of `1240 ms` and a relative drive MAE of `2.45 %` from synthetic logs. This is not a measurement of the app.

Quantity	Value	Conditions	Basis
JVM unit tests passed	21 tests	of 21 run (AccuracyTest 16, AlertGateTest 3, StaleTrackTest 2); 0 failed, 0 skipped	Reproduced 2026-11-10
AccuracyTest suite execution time	0.369 s	JUnit testsuite time for 16 simulation tests, excluding build	Reproduced 2026-11-10
Unit-test command wall time	104.236666209 s	whole ./gradlew --no-daemon :app:testDebugUnitTest invocation incl. Gradle daemon start, dependency resolution and Kotlin compilation (BUILD SUCCESSFUL in 1m 43s)	Reproduced 2026-11-10
Validation-script selftest checks passed	4 checks	of 4 printed checks ([PASS] static scale recovery, drive auto time offset, drive error recovery, warning audit); SELFTEST PASSED, exit 0	Reproduced 2026-11-10
Selftest: recovered static range scale ratio	1.048 ratio	expected 1.05; check passes within 0.01	Reproduced 2026-11-10
Selftest: recovered clock offset between synthetic logs	1240 ms	expected +1200 ms; 1817 frames; check passes within 60 ms	Reproduced 2026-11-10
Selftest: recovered relative MAE on synthetic drive	2.45 %	expected ~2.4% for 3% injected noise; check passes in 1.5-3.2%	Reproduced 2026-11-10
Validation-script selftest wall time	0.190178353 s	whole selftest command, CPU	Reproduced 2026-11-10

Table 3. Table 3. Test-suite outcome and validation-script self-test on synthetic logs; workstation CPU; reproduced by the author. “Reproduced” rows were run by the author; “Reported” rows are quoted from the repository at the commit shown.

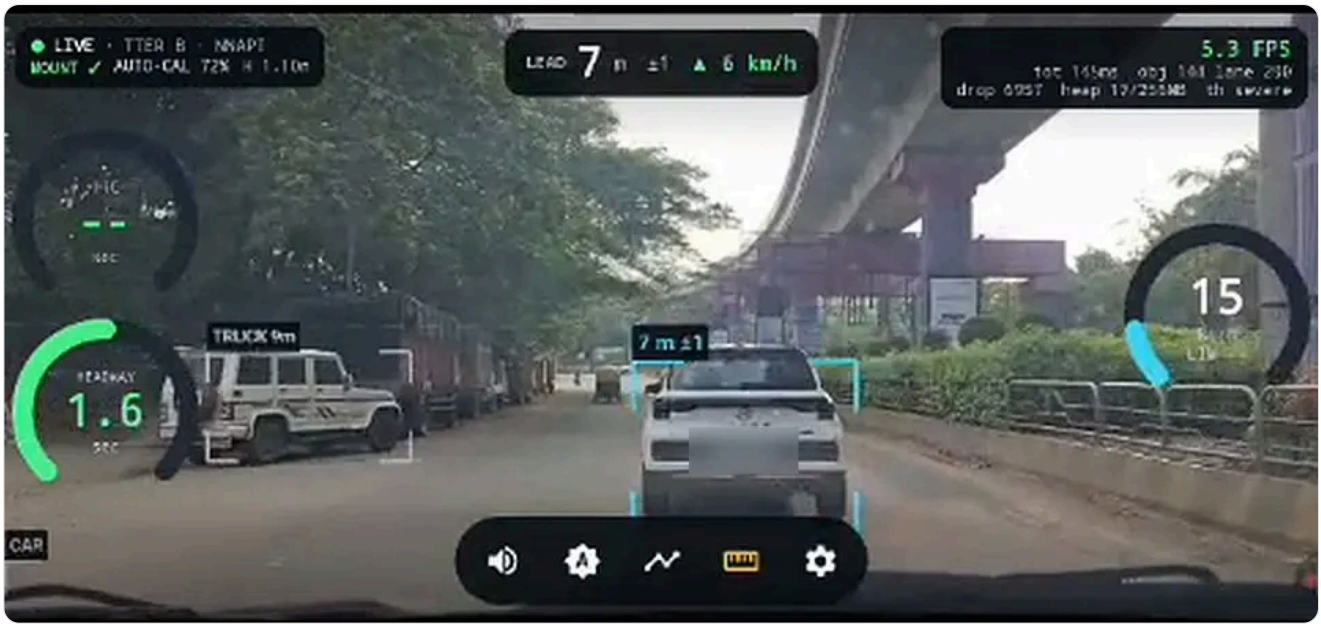


Figure 5. The app's display on a city drive: the lead vehicle's range with its uncertainty, the headway, and no warning issued. Still from the repository (docs/img/drive-stop-go.jpg), number plates blurred there. Illustrative only; no on-road measurement is reported here (Limitations).

5 Limitations

- **Everything is simulation.** Detections are synthetic boxes with Gaussian edge noise; the neural detector, real optics, road slope, glare, night and weather are not in the loop, and class confusion is modelled only in the duplicate truck/bus case. The README itself states that these are simulations, not road certification.
- **No on-road accuracy result is published.** The release notes say v2 was road-tested in Bengaluru traffic, and the README shows screen recordings from a city drive there. `VALIDATION.md` is a protocol with no recorded results, and a radar-referenced real-world dataset is an open roadmap item. The author has no evidence of range error, TTC error or false-warning rate on a real road.

- **Field validation needs hardware.** `validate_accuracy.py` in drive and warning-audit modes needs phone drive logs and a reference radar or lidar log; only its synthetic self-test was run.
- **Three README figures conflict with the run.** v2 range MAE (`2 m` stated, `2.07 m` measured), v2 jitter (`1 m` stated, `1.09 m` measured), and the hard-brake median critical alarm. The first two README figures are the more favourable; for the third, the recorded values give more warning time.
- **The v1 baseline is a replica**, with a field of view that varies between tests and different counting conventions in the soak and steady-following tests.
- **Narrow scenarios.** Apart from hard braking, each scenario runs one seed; the range test covers three car widths and no two-wheelers, trucks or auto-rickshaws.
- **Indian-road adaptation is not in this build.** `models/README.md` states that drivable-area fine-tuning on the India Driving Dataset was not run in this build. The shipped object labels are COCO, without an auto-rickshaw class, so the auto-rickshaw width prior applies only with a fine-tuned label file. `models/README.md` also says the app ships with no real weights, while the tag contains bundled `.tflite` files that `ARCHITECTURE.md` describes as real.
- **No device performance is established.** The app is described as running on budget phones; tier frame rates are targets, and the release notes' claim that tracking holds at budget-phone frame rates is supported here only by one simulated crossing at 7 fps. No claim is made about frame rate, latency, thermals or warning quality on any phone.

6 Reproducibility

- **Version.** Lyne ADAS tag v2.0.0, commit a68060b.
- **Simulation suite.** `JAVA_HOME=<JDK 17> ANDROID_HOME=<Android SDK with platform 35> ./gradlew --no-daemon :app:testDebugUnitTest`; figures are in `app/build/test-results/testDebugUnitTest/TEST-com.lyne.adas.l1.fusion.AccuracyTest.xml` under `<system-out>`. Runtime: JDK 17.0.19 (Temurin), Gradle 8.11.1, AGP 8.7.3, Kotlin 2.1.0; AMD Ryzen 9 9900X, 30 GB RAM, Linux 7.0.0-42 (Ubuntu), CPU only. The suite took `0.369 s`; the whole command, including Gradle start-up and compilation, `104.236666209 s`. A first attempt with a JDK 21 installation lacking a Java compiler failed at task configuration; the build targets Java 17, as the README states.
- **Validation-script self-test.** `python -I tools/validate_accuracy.py selftest`, Python 3.12, standard library only, same machine; `0.190178353 s`.
- **Not run by the author.** APK build and install; any on-device run; the in-app parked range check; `validate_accuracy.py` in `static`, `drive` and `warnings` modes, which need phone logs and a reference radar, lidar or rangefinder log; IDD fine-tuning (`models/convert/finetune_idd.py`), which needs a GPU and the dataset.

References

1. Sherin Joseph Roy. Lyne ADAS, release v2.0.0 (commit a68060b). Source code, MIT. github.com/Sherin-SEF-AI/Lyne-ADAS/tree/v2.0.0
2. G. Varma, A. Subramanian, A. Namboodiri, M. Chandraker, C. V. Jawahar. IDD: A dataset for exploring problems of autonomous navigation in unconstrained environments. WACV 2019. doi:10.1109/WACV.2019.00190
3. J. Civera, A. J. Davison, J. M. M. Montiel. Inverse depth parametrization for monocular SLAM. IEEE Transactions on Robotics 24(5), 2008. doi:10.1109/TRO.2008.2003276
4. R. E. Kalman. A new approach to linear filtering and prediction problems. Journal of Basic Engineering 82(1), 1960. doi:10.1115/1.3662552
5. J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. Neural Networks 32, 2012. doi:10.1016/j.neunet.2012.02.016
6. Q.-H. Che, D.-P. Nguyen, M.-Q. Pham, D.-K. Lam. TwinLiteNet: An efficient and lightweight model for driveable area and lane segmentation in self-driving cars. arXiv:2307.10705, 2023. arxiv.org/abs/2307.10705
7. G. Jocher, A. Chaurasia, J. Qiu. Ultralytics YOLOv8. Source code, 2023. github.com/ultralytics/ultralytics

Evidence

Quantity	Value	Conditions
False forward-collision warnings, 20-min random traffic soak (v1 baseline)	626 warnings	same 1200 s scen replayed through width-only FCW (LegacyFcw); cou not asserted
False forward-collision warnings, 20-min random traffic soak (v2)	0 warnings	1200 s simulated fps, ego 22 m/s, l gap varying 15-45 adjacent-lane car; true TTC printed a s; test asserts 0
Minimum true TTC in the 20-min soak scenario	12.4 s	ground-truth TTC simulated lead; te asserts > 4 s
False FCW events, 2 min steady following at 30 m (v1 baseline, 8 fps)	10 events	120 s at 8 fps, lea m, ego 25 m/s, v1 only FCW; counte asserted

Quantity	Value	Conditions
False FCW events, 2 min steady following at 30 m (v1 baseline, 15 fps)	112 events	120 s at 15 fps, lea 30 m, ego 25 m/s width-only FCW; counted, not asse
False FCW events, 2 min steady following at 30 m (v1 baseline, 30 fps)	514 events	120 s at 30 fps, le 30 m, ego 25 m/s width-only FCW; counted, not asse
False FCW events, 2 min steady following at 30 m (v2)	0 events	printed 0 at each and 30 fps; ego 2 test asserts 0 per rate
Mean absolute range error (v1 width-only baseline)	7.9 m	same 21 scenarios width-only with co 69 deg FoV

Quantity	Value	Conditions
Mean absolute range error (v2 fused range)	2.07 m	hatchback/sedan, widths 1.68/1.82/2 at 8-60 m (7 dista 15 fps, app calibra with +5 cm height +0.005 rad pitch e samples after 1.5 s asserts v2 < 0.8 x Differs from anoth source; both are li
Range jitter at 30 m, std dev (v1 baseline)	5 m	same scenario, v1 only range with 66 FoV
Range jitter at 30 m, std dev (v2)	1.09 m	static lead at 30 n at 15 fps, samples s; test asserts v2 sd / 3 Differs from another source; b listed.

Quantity	Value	Conditions
Real TTC at first FCW caution, lead brakes 6 m/s^2 (per seed)	`1.4068645, 1.8833354, 1.7559159, 1.4068645, 1.5181836, 1.8833354, 1.0981997, 1.0981997 s`	8 noise seeds; lea m, both at 22 m/s brakes at 6 m/s^2 t=1.5 s, 15 fps; tes asserts sorted[4]
Real TTC at first FCW critical (brake) alarm, lead brakes 6 m/s^2 (per seed)	`0.8208347, 0.8208347, 1.3000017, 0.8208347, 0.8208347, 1.002633, 1.0981997, 1.002633 s`	same 8 seeds and scenario; test ass every seed fires w TTC > 0.5 s Differs another source; b listed.
Real TTC at FCW caution, gentle approach	`2.0666754 s`	closing 5 m/s fron ego 20 m/s, 15 fp 4; test asserts > 1.
Real TTC at FCW caution with duplicate truck/bus detections	`1.7571429 s`	truck at 30 m rep as both truck and each frame, lead b 5 m/s^2, 10 Hz; te asserts > 0.8 s

Quantity	Value	Conditions
VRU critical events, pedestrian at constant 8 m	`0 events`	10 s at 8 fps, ego s input 14 m/s, pede gap held at 8 m; t asserts 0
Real time-to-reach at VRU critical, pedestrian in path	`1 s`	pedestrian from 3 ego 10 m/s (36 km fps; test asserts >
Track ids used for one car crossing the frame at 7 fps	`1 track ids`	car moving 3.5 m/ laterally from x=-5 12 m, 3 s at 7 fps; asserts <= 1
Camera height learned by auto-calibration	`1.42 m`	configured 1.20 m 1.45 m, 40 random cars x 4 s at 15 fp

Quantity	Value	Conditions
		asserts |learned - 0.08 m
False collision warnings, 20 min random traffic (v1)	626 warnings	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b true TTC never un s
False collision warnings, 20 min random traffic (v2)	0 warnings	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b true TTC never un s
False warnings, 2 min steady following at 30 m (v1, 8 / 15 / 30 fps)	10 / 112 / 514 warnings	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b
False warnings, 2 min steady following at 30 m (v2, 8 / 15 / 30 fps)	0 / 0 / 0 warnings	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b
Mean range error, hatchback / sedan / pickup at 8 to 60 m (v1)	7.9 m	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b with 5 cm height a deg pitch calibrati error
Mean range error, hatchback / sedan / pickup at 8 to 60 m (v2)	2 m	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b with 5 cm height a deg pitch calibrati

Quantity	Value	Conditions
		error Differs from source; both are li
Distance jitter at 30 m, std dev (v1)	5 m	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b
Distance jitter at 30 m, std dev (v2)	1 m	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b Differs from anoth source; both are li
Real TTC at caution, lead brakes 6 m/s^2 from 35 m at 80 km/h (median of 8 runs)	≈1.5 s	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b
Real TTC at brake alarm, lead brakes 6 m/s^2 from 35 m at 80 km/h (median of 8 runs)	≈0.8 s	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b Differs from anoth source; both are li
Caution at real TTC, gentle approach closing at 18 km/h	2.1 s	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b
Camera height learned by auto-calibration (configured 1.20 m, true 1.45 m)	1.42 m	README 'Measure reproducibly' table synthetic drives, 1. per box edge at 2 input, 5% missed detections, road b

Quantity	Value	Conditions
JVM unit tests passed	21 tests	of 21 run (Accurac 16, AlertGateTest 3 StaleTrackTest 2); failed, 0 skipped
Unit-test command wall time	104.236666209 s	whole ./gradlew -- daemon :app:testDebugUr invocation incl. Gr daemon start, dependency resol and Kotlin compila (BUILD SUCCESS 1m 43s)
AccuracyTest suite execution time	0.369 s	JUnit testsuite tim 16 simulation tests excluding build
Validation-script selftest checks passed	4 checks	of 4 printed check ([PASS] static sca recovery, drive aut offset, drive error recovery, warning SELFTEST PASSE 0

Quantity	Value	Conditions
Selftest: recovered static range scale ratio	1.048 ratio	expected 1.05; ch passes within 0.0
Selftest: recovered clock offset between synthetic logs	1240 ms	expected +1200 m frames; check pas within 60 ms
Selftest: recovered relative MAE on synthetic drive	2.45 %	expected ~2.4% fo injected noise; ch passes in 1.5-3.2%
Validation-script selftest wall time	0.190178353 s	whole selftest cor CPU

Table 4. Every measured quantity in this report, with its source, commit, hardware and the command that reproduces it. “Reproduced” rows were run by the author; “Reported” rows are quoted from the repository at the commit shown.

Cite this report

APA

Roy, S. J. (2026). Monocular range and TTC for a warning-only phone ADAS: v1 false warnings and the v2 chain in simulation (Technical Report IF-TR-2026-03, Version 1.0). IsoFrames. https://isoframes.io/reports/if-tr-2026-03/

BibTeX

```
@techreport{isoframes_if_tr_2026_03,
  author      = {Roy, Sherin Joseph},
  title       = {Monocular range and TTC for a warning-only phone ADAS: v1 false warnings and the v2 chain in simulation},
  institution = {IsoFrames, BluRabbit Mobility},
  number      = {IF-TR-2026-03},
  type        = {Technical Report},
```

```
  year         = {2026},
  month        = nov,
  note         = {Version 1.0},
  url          = {https://isoframes.io/reports/if-tr-2026-03/}
}
```