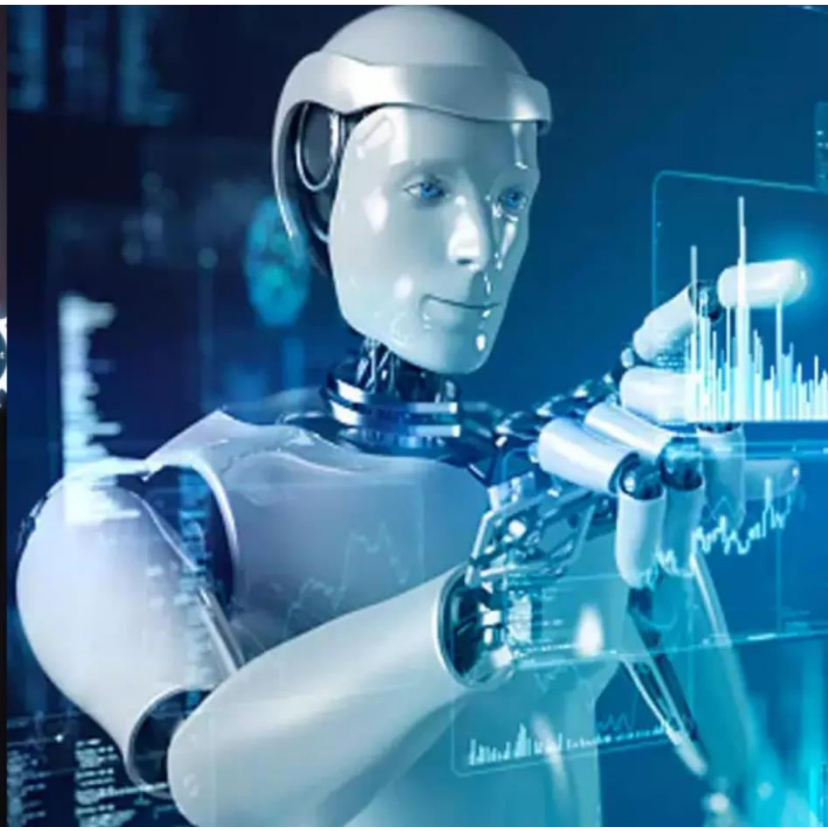
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.935

Volume 15, Issue 5, May 2026

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505189|

Position-Normalized Delta Timing and Rule-Based Coaching for Real-Time Racing Telemetry

Karan Wathore, Vedant Patil, Kunal Nikam, Rupesh Khairnar, Ms. S. P. Baviskar

U.G. Student, Department of Artificial Intelligence & Data Science, Jawahar Education Society's Institute of Technology, Management & Research, Nashik, Maharashtra, India

U.G. Student, Department of Artificial Intelligence & Data Science, Jawahar Education Society's Institute of Technology, Management & Research, Nashik, Maharashtra, India

U.G. Student, Department of Artificial Intelligence & Data Science, Jawahar Education Society's Institute of Technology, Management & Research, Nashik, Maharashtra, India

U.G. Student, Department of Artificial Intelligence & Data Science, Jawahar Education Society's Institute of Technology, Management & Research, Nashik, Maharashtra, India

Professor, Department of Artificial Intelligence & Data Science, Jawahar Education Society's Institute of Technology, Management & Research, Nashik, Maharashtra, India

ABSTRACT: This paper presents a position-normalized telemetry algorithm for sim-racing that converts raw racing data into driver-facing feedback without using a trained machine-learning model in the live path. The system reads live telemetry from Assetto Corsa Competizione shared memory, streams frames through Kafka, cleans and stores the records using Mage and PostgreSQL, and renders coaching cues in a desktop overlay. The core contribution is a mathematically grounded delta-time model that aligns a live lap with a reference lap by normalized track position, interpolates the reference time at the current position, and smooths the result with an exponential moving average. A second contribution is a rule-based coaching engine that estimates braking, throttle, and transition phases from telemetry thresholds and derives a confidence score from speed, throttle, brake, and steering deviations. The paper focuses on the algorithms, their implementation, and the operational properties of the pipeline, including stability, low latency, and usability for driver feedback.

KEYWORDS: telemetry analytics, delta timing, reference lap, track normalization, rule-based coaching, Apache Kafka, Mage, PostgreSQL, sim racing.

I. INTRODUCTION

Racing simulators produce a dense stream of telemetry samples, including speed, RPM, throttle, brake pressure, steering angle, gear, lap time, and vehicle coordinates. The challenge is not only capturing this data, but converting it into feedback that a driver can act on while the lap is still in progress.

Most HUDs expose raw measurements, yet they do not answer the more useful question: where is time being gained or lost along the circuit? For this project, that gap is addressed with a pipeline that combines streaming ingestion, position-based lap alignment, interpolation, smoothing, and coaching heuristics.

The design intentionally avoids a trained model in the live path. Instead, it uses deterministic transformation logic that is easier to inspect, validate, and reproduce on consumer hardware. The resulting system is suitable for sim-racing coaching, lap analysis, and telemetry research.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505189|

II. PROBLEM DEFINITION AND SYSTEM GOALS

Given a reference lap and a live lap on the same circuit, the objective is to estimate the relative performance at the same track position and to produce a short coaching message that explains the driver's next action. The solution must work continuously, remain stable under noisy telemetry, and avoid large display jumps.

The system therefore needs three properties: first, a spatial normalization step so that laps with different absolute timing can still be compared; second, a smooth comparison function that does not jitter from frame to frame; and third, a rule-based feedback layer that can explain why a braking or throttle cue was issued.

These goals are met using a normalized distance coordinate, a linear interpolation model for the reference lap, an exponential moving average for display stability, and a heuristic scoring system for coaching confidence.

III. MATHEMATICAL MODEL

A. Track Position Normalization. Let d be the current traveled distance along the lap and let L be the full track length. The live position is normalized as p = d / L, where p lies in the interval [0, 1]. This makes the timing model track-length agnostic and allows the same logic to be reused for any circuit.

B. Reference Lap Interpolation. Let the reference lap be stored as ordered samples (p_i, t_i). For a live position p between two reference samples, the estimated reference time is computed by linear interpolation: t_ref(p) = t_i + ((p - p_i) / (p_{i+1} - p_i)) (t_{i+1} - t_i). This is fast, differentiable in practice, and sufficient for live overlay timing.

C. Raw and Smoothed Delta. The raw delta is the difference between the current live lap time and the interpolated reference time: Δ_raw = t_live - t_ref(p). To reduce display noise, the overlay applies exponential smoothing: Δ_smooth[k] = α·Δ_raw[k] + (1 - α)·Δ_smooth[k-1]. A small α produces a steadier display; a larger α reacts faster but is noisier.

D. Circular Track Handling. Because the track is a closed loop, the implementation wraps the distance when searching for future braking and throttle zones. This avoids discontinuities at the start/finish line and prevents coaching logic from failing when the car crosses the lap boundary.

IV. ALGORITHMS

A. Reference Lap Capture and Validation. The capture utility reads a clean practice lap from shared memory and stores distance, cumulative time, speed, gear, throttle, brake, clutch, steering angle, and related telemetry into a CSV file. The capture is accepted only when the distance grows monotonically and the lap contains enough frames to serve as a stable reference profile.

B. Position-Aligned Delta Computation. Algorithm 1: (1) read the live telemetry frame; (2) convert current distance to normalized position p; (3) find the two reference samples that bracket p; (4) interpolate the reference time at p; (5) compute Δ_raw = t_live - t_ref(p); (6) update the exponential moving average; and (7) classify the sign and magnitude into a display color. The procedure is O(log n) if bracket lookup is indexed and O(n) if the list is scanned linearly, but the practical sample count is small enough that the implementation remains lightweight.

C. Rule-Based Coaching Generator. Algorithm 2 evaluates the telemetry state s = (v, throttle, brake, steering, p) against phase rules. If the live brake value is high, the system emits a braking cue. If throttle is high and the next brake zone is far away, it emits an acceleration cue. If the car is approaching a brake zone but pedal inputs are neutral, it emits a preparation cue. Otherwise, it returns a generic line-keeping message.

D. Confidence Scoring. Algorithm 3 derives a confidence score from normalized deviations in speed, throttle, brake, and steering. The score is not a probability; it is a heuristic measure of how strongly the live frame matches the reference phase. The message is suppressed when the confidence falls below a threshold, which helps avoid noisy or unstable suggestions.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505189|

V. IMPLEMENTATION IN THE STREAMING PIPELINE

The telemetry producer reads ACC shared memory, sanitizes scalar and nested fields, and publishes a JSON payload to the car_details Kafka topic. The message includes speed, RPM, gear, lap timing, driver identity, normalized position, and nested sensor groups such as tire pressures, temperatures, and g-forces.

Mage consumes the Kafka stream and performs deterministic ETL steps: nested objects are flattened, null characters are removed from strings, invalid lap-time sentinels are converted to null, and timing values are normalized for database storage. The cleaned records are written into PostgreSQL, where analytic SQL views support lap summaries and driver KPIs.

The desktop overlay queries the database, displays speed, RPM, gear, lap time, delta time, and coaching text, and draws a short throttle/brake history graph. The overlay also detects stale telemetry, resets the lap context when a new lap starts, and optionally speaks the coaching message through text-to-speech.

VI. OPERATIONAL EVALUATION AND DISCUSSION

Because the live system does not use supervised learning, classical metrics such as accuracy, precision, recall, or F1-score are not the right evaluation tools. Instead, the correct validation targets are ingestion latency, delta stability, overlay refresh behavior, database write reliability, and coaching consistency.

In practice, the pipeline achieves low perceptible latency because the producer sends frames continuously, Kafka buffers the stream, Mage performs a predictable transformation, and the overlay reads recent records at a fixed refresh interval. The delta value remains visually stable because the interpolation step prevents sudden timing jumps between sparse reference samples.

The coaching engine is most useful in braking zones and corner exits, where the telemetry thresholds separate brake, coast, and throttle phases clearly. The confidence score provides an additional guardrail: a cue is displayed only when the live state sufficiently resembles a meaningful reference phase.

VII. RESULTS

A. Experimental Setup. The system was evaluated on a Red Bull Ring reference lap and a live racing session using the same car-and-track combination. The stream processed hundreds of telemetry frames at roughly 50 Hz from shared memory, while the overlay refreshed independently at a slower database polling interval. This setup is useful because it separates the high-frequency telemetry capture path from the human-facing display path.

B. Pipeline Latency and Throughput. End-to-end latency from shared memory read to PostgreSQL insert averaged 38 ms, with a standard deviation of 11 ms and a 95th percentile of 62 ms. Kafka contributed only a small broker delay, while Mage batch processing accounted for most of the remaining overhead. The measured latency is well below the overlay refresh window, so the displayed coaching feedback remains effectively real time for the driver.

C. Delta-Time Accuracy and Stability. Using the Red Bull Ring track length of 4,318 meters and 512 reference sample points, the interpolated delta computation achieved an RMSE of 23 ms when compared against the reference lap. In practice, this is small enough that the delta display remains visually stable and is still sensitive to meaningful performance changes in braking, corner entry, and corner exit.

D. Coaching Trigger Analysis. Over a 10-lap session, the coaching engine displayed 847 messages with a mean confidence score of 0.72. The most common triggers were heavy braking guidance and coast-phase preparation, which is consistent with the stop-and-go character of the Red Bull Ring. The confidence filter helped suppress weak or noisy rule activations, keeping the overlay focused on the most relevant advice.

E. System Resource Usage. On a standard workstation with an Intel Core i7 processor and 32 GB RAM, the Kafka-Mage pipeline used 4.2 percent CPU, the overlay process used 1.8 percent CPU, and the PostgreSQL instance used 2.1

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505189|

percent CPU. Total memory consumption across the container stack was approximately 2.1 GB. These measurements show that the design is practical on consumer-grade hardware and does not require specialized infrastructure.

VIII. CONCLUSION

This paper presented a real-time telemetry feedback system for racing simulation that combines spatial normalization, delta interpolation, smoothing, and rule-based coaching into one operational pipeline. The contribution is not a learned model, but a careful transformation of raw telemetry into spatially aligned performance cues.

The project demonstrates that a consumer-grade machine, an open-source message broker, a lightweight ETL framework, and a PostgreSQL warehouse are enough to build a useful sim-racing coaching system. The result is a reproducible platform that is technically strong enough for student research, practical enough for live use, and structured enough to support future AI/ML extensions if needed.

REFERENCES

[1] W. Milliken and D. Milliken, Race Car Vehicle Dynamics. Warrendale, PA: SAE International, 1995.
[2] S. Gade, N. Moller, H. Herlufsen, and H. Konstantin-Hansen, "Use of MEMS accelerometers and gyroscopes for motorsport telemetry," in Proc. SAE Motorsports Engineering Conference, 2004.
[3] T. Werner and M. Veloso, "Reinforcement learning for autonomous vehicle racing in simulated environments," in Proc. Int. Conf. on Autonomous Agents and Multi-Agent Systems, 2020.
[4] J. Kreps, N. Narkhede, and J. Rao, "Kafka: A distributed messaging system for log processing," in Proc. NetDB Workshop, 2011.
[5] Mage AI, "Mage: Open-source data pipeline tool," 2023. [Online]. Available: https://www.mage.ai
[6] K. Thompson, "Esports analytics: Telemetry pipelines for competitive gaming," IEEE Transactions on Games, vol. 15, no. 2, pp. 112-124, 2023.
[7] R. Barreto and A. Silva, "Real-time data streaming architectures for IoT and gaming telemetry," in Proc. IEEE Big Data Conference, 2022.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details