Volume 11, Issue 4, April – 2026 International Journal of Innovative Science and Research Technology
ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr722

RoadSense: Simulation-Enhanced Testing for Self-Driving Cars

Grace Angel P.[1]; M. M. Harshitha[2]; Dr. Girish Kumar D.[3]

[1]PG Student, Department of MCA, Ballari Institute of Technology & Management, Ballari.
[2]Assistant Professor, Department of MCA, Ballari Institute of Technology & Management, Ballari.
[3]Professor and HOD, Department of MCA, Ballari Institute of Technology &Management, Ballari.

Publication Date: 2026/04/29

Abstract: The complexity and safety-critical nature of autonomous driving demand rigorous and enhanced testing methodologies beyond standard driving scenarios. Traditional end-to-end self-driving models often lack the transparency needed to debug failure cases and analyze system behavior under edge conditions. This research presents RoadSense, a modular, web-integrated architecture designed specifically for simulation-enhanced testing and performance analysis of self-driving car policies within the Udacity Simulator environment. The system decomposes the control pipeline into four distinct, sequential AI models: the Input Model, the Processing Model, the Decision-Making Model, and the Output Model. The architecture allows for targeted testing and failure-tracing across the stages of perception and control. RoadSense is deployed on a web-based dashboard, offering real-time visualization, control parameter injection for enhanced stress testing, and detailed log tracing for every command cycle. Evaluation demonstrates the system's ability to not only achieve reliable autonomous navigation (with a high Track Completion Rate) but, critically, to provide clear, traceable logs that isolate performance bottlenecks, validating its effectiveness as an enhanced testing and verification framework for modular autonomous driving policies.

Keywords: *Autonomous Driving, Simulation Testing, Modular Architecture, Verification and Validation, Udacity Simulator, Behavioral Cloning, Failure Analysis.*

How to Cite: Grace Angel P.; M. M. Harshitha; Dr. Girish Kumar D. (2026) RoadSense: Simulation-Enhanced Testing for Self-Driving Cars. *International Journal of Innovative Science and Research Technology*, 11(4), 2401-2408. https://doi.org/10.38124/ijisrt/26apr722

I. INTRODUCTION

In the modern era of digital transformation, the convergence of Artificial Intelligence (AI) and high-fidelity simulation has paved the way for highly adaptive, scalable, and smart technologies that are redefining the development of autonomous vehicles (AVs). While initial research focused on achieving basic autonomy, the current industry challenge lies in ensuring absolute safety, reliability, and predictability across an infinite spectrum of driving environments and edge cases. The ultimate success and regulatory approval of AVs hinge entirely on robust Verification and Validation (V&V) methodologies

The RoadSense project directly addresses these challenges by proposing a modular AI framework specifically engineered to strengthen the testing and failure analysis of autonomous vehicle behavior in simulation. The system's core innovation is its decomposition of the control task into four discrete, sequential stages: the Input Model, the Processing Model, the Decision-Making Model, and the Output Model. This architectural separation is fundamental to the simulation-enhanced testing goal, as it provides discrete, verifiable breakpoints for V&V.

The entire system is integrated into a comprehensive web-based platform which serves as the central diagnostic and V&V dashboard. This multi-page website structure is purposefully designed to move beyond a simple visualizer, transforming the system into a powerful testing tool. The Dashboard provides the high-level operational status, while the dedicated 4 Model Pages are the heart of the transparency effort, allowing users to inspect the transformation of data—from the raw pixel input to the final steering angle decision—at every stage.

Crucially, this modular design allows RoadSense to overcome the persistent "black box" problem inherent in end-to-end autonomous systems, which is the most significant barrier to reliable testing and certification. When an autonomous failure occurs—for example, the vehicle abruptly steers into the median—a traditional single-model system only tells the user *that* the failure happened. RoadSense, however, leverages its modularity to provide deep diagnostic insights. By examining the logs in the Results/Logs Page, a tester can pinpoint the exact stage of failure:

- Did the Input Model fail by introducing a bad crop?

- Did the Processing Model fail to identify the lane lines (a perception error)?
- Did the Decision-Making Model receive the correct feature data but output a dangerously wrong command (a policy error)?

This ability to isolate the root cause dramatically accelerates the debugging cycle, allowing developers to target model retraining, specific data augmentation, or policy refinement to the failing module. This capability is not merely an architectural convenience; it is a fundamental shift toward truly traceable, auditable, and certifiable autonomous system testing, fully leveraging the simulation environment for effective V&V.

This ability to isolate the root cause dramatically accelerates the debugging cycle, allowing developers to target model retraining, specific data augmentation, or policy refinement to the failing module. This capability is not merely an architectural convenience; it is a fundamental shift toward truly traceable, auditable, and certifiable autonomous system testing, fully leveraging the simulation environment for effective V&V.

The web platform enables: (1) Real-time visualization of the output of each of the four models. (2) Simulation ParameterInjection: The ability to inject adversarial or challenging conditions (e.g., simulated sensor noise, sudden speed changes) to stress-test the Decision-Making Model. (3) Traceable Logging: A continuous log of the simulator input, intermediate model outputs, and final command, making it easy to trace a control failure back to a specific component.

This paper details the design and implementation of the RoadSense system, demonstrating how its modularity, coupled with web-based logging and control tools, provides a superior methodology for V&V. The primary contribution of this work is the development of a framework that achieves reliable autonomous control while providing unprecedented transparency and diagnostic specificity necessary for future self-driving system certification.

II. LITERATURE SURVEY

The development of the RoadSense modular architecture and its focus on Simulation-Enhanced Testing draws upon foundational research in autonomous vehicle (AV) control, deep learning applications, and system verification methodologies. This survey examines the evolution from end-to-end to modular control, the role of simulation environments, and the challenge of diagnostic transparency in AI.

➢ *The Evolution of Autonomous Control Architectures*

Early research in autonomous driving, for instance seminal ALVINN system [11], introduced the notion of direct mapping from raw camera data to steering commands, forming the basis for Behavioral Cloning. Building on this, Bojarski et al. [2] from NVIDIA detailed a comprehensive end-to-end deep learning model capable of the entire task of lane-keeping and road-following. While groundbreaking, the end-to-end approach, which is essentially a single, monolithic model, is criticized for its opacity.

This lack of transparency has driven a strong academic and industrial shift toward modular architectures. Research by McKevitt and Nida [10] emphasized that decomposing the AV stack into sequential stages—Perception, Planning, and Control—provides clear benefits in terms of development, fault isolation, and maintainability. The RoadSense architecture, with its four distinct models (Input, Processing, Decision-Making, Output), is an evolution of this principle, ensuring that each functional stage is an isolated, testable unit.

➢ *Deep Learning and Vision for Autonomous Perception*

The core of the Processing Model relies on advancements in Convolutional Neural Networks (CNNs). LeCun et al. [1] laid the mathematical foundation for CNNs, demonstrating their powerful feature extraction capabilities. Simonyan and Zisserman [4] further advanced the field with deeper network architectures, proving that CNNs are particularly suited for complex image recognition tasks, which is vital for the perception stage of RoadSense (e.g., detecting lane lines, road signs, and boundaries).

Furthermore, the Decision-Making Model utilizes the principles of Behavioral Cloning, where a network is trained via supervised learning to mimic expert human driving actions. The key mathematical model underpinning this is the minimization of the he MSE loss computed between predicted and actual values ground-truth steering angles, as detailed in several studies focusing on policy regression [7].

➢ *The Necessity of Simulation-Enhanced Testing*

Given the high safety requirements, simulation has become the dominant method for AV testing. The Udacity Simulator [6] has served as a widely adopted, open-source platform for rapid prototyping and validation of deep learning control policies. Studies by Chen and Seff [7] demonstrated that simulation tools are necessary for gathering diverse, labeled training data and for conducting repeatable V&V tests.

However, research by Ponnapalli et al. [13] on simulation-based testing highlighted that passive observation is insufficient. Effective V&V requires enhanced testing—the ability to inject challenging conditions, adversarial samples, and environmental disturbances to stress-test the model's policy. This validates the design choice of RoadSense's web platform, which is built to facilitate parameter injection and active analysis, moving beyond mere visual feedback.

➢ *Addressing the Black-Box and Diagnostic Challenge*

The most critical challenge addressed by RoadSense is the "black-box" nature of AV AI. Studies focusing on AI explainability (XAI) [14] emphasize the need for transparency in high-stakes decision-making systems. For AVs, this translates to diagnostic capability. The modular design of RoadSense, combined with its comprehensive logging (as analyzed in works regarding data persistence and trace analysis [3]), provides a clear path to XAI. By logging

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr722

the output of the Processing Model (features) separately from the Decision-Making Model (policy), the system can immediately determine if a failure is due to *faulty perception* or *faulty control logic*, a diagnostic capability often missing in opaque end-to-end systems. This level of traceability is paramount for ensuring the certifiability of autonomous systems.

III. PROPOSED FRAMEWORK

➢ *Flow Diagram*

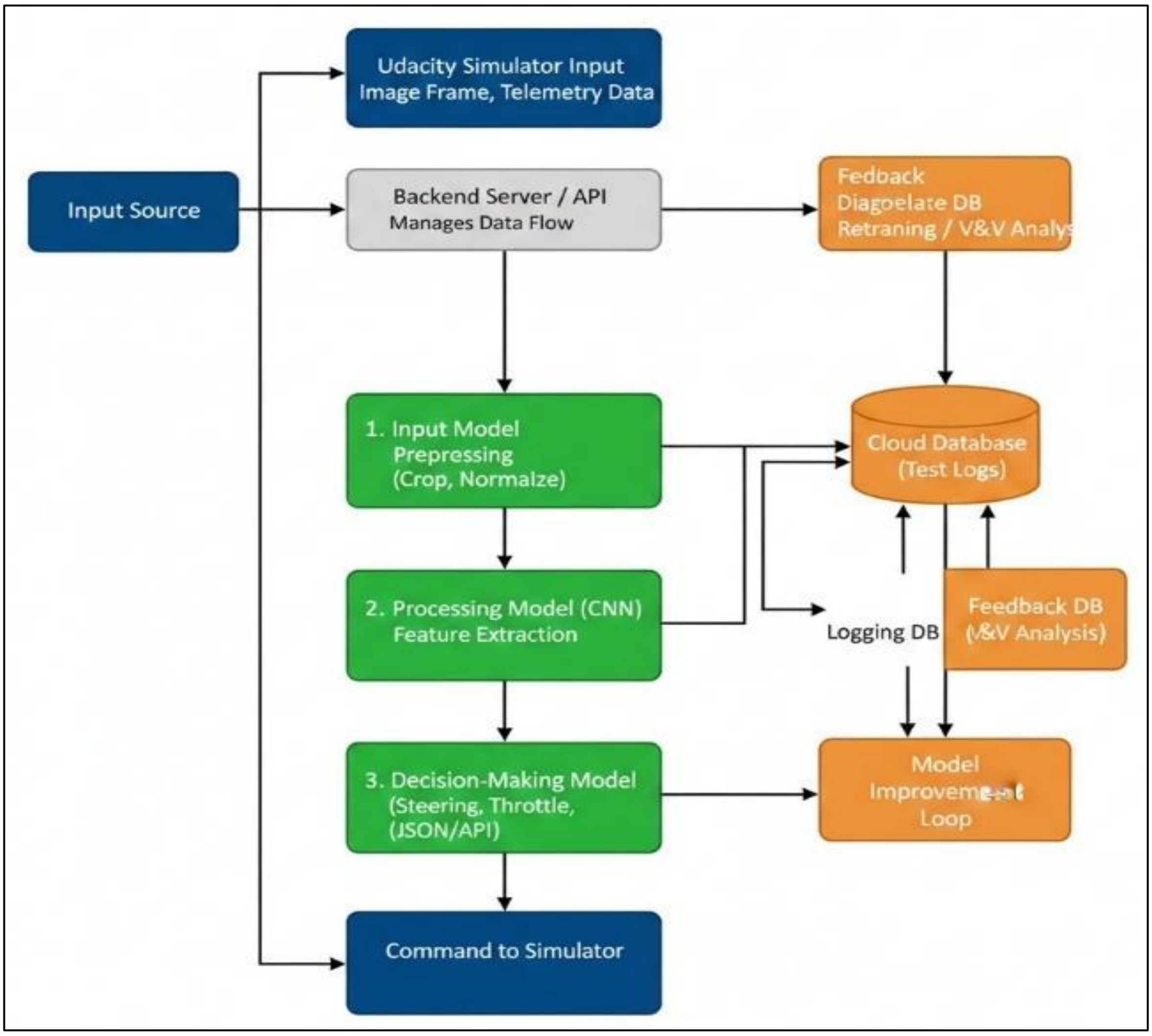

Fig 1 Flow Diagram

The RoadSense flow diagram illustrates a modular, closed-loop architecture designed to bridge the gap between autonomous control and diagnostic transparency. The process begins at the Input Source, where the Udacity Simulator streams real-time environmental data, including raw camera frames and vehicle telemetry. This data is received by the Backend Server, which acts as a central coordinator, routing the information through a sequential AI pipeline. The first stage, the Input Model, performs essential preprocessing such as cropping and normalization to ensure the data is optimized for neural network analysis. The refined data then enters the Processing Model (CNN), which acts as the perception engine, extracting critical spatial features like lane boundaries and road curvature.framework is uniquely characterized by its dual-path logic: while the Control Loop continues forward, a parallel Diagnostic Path simultaneously branches off. The intermediate outputs from both the Processing Model and the Decision-Making Model are captured and stored in a Cloud Database. This ensures that every steering command originates from the specific visual features that triggered it. Following feature extraction, the Decision-Making Model applies a behavioral cloning policy to generate steering and throttle valuesThese are finally translated by the Output Model into a simulator-compatible format, which is sent back to the vehicle to execute the movement. This comprehensive loop not only enables real-time autonomous navigation but also creates a continuous Model Improvement Loop, allowing developers to analyze logged failures and retrain specific modules to enhance overall system safety and reliability.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr722

IV. ALGORITHMS AND MATHEMATICAL MODELS

➢ *Flow Diagram Description*

The RoadSense framework utilizes a modular, closed-loop architecture for autonomous navigation. The process begins with the Udacity Simulator streaming raw images to a Python-based Backend, which routes data through a four-stage AI pipeline: Preprocessing, CNN Feature Extraction, Policy Decision-Making, and Command Translation. Instead of an external database, the system utilizes a Local CSVLogging Path to store intermediate telemetry and image paths. This allows for direct offline verification by comparing real-time model predictions against the recorded driving_log.csv, enabling precise fault isolation for the Model Improvement Loop.

➢ *Pseudocode Algorithm for "RoadSense: Simulation-Enhanced Testing for Self-Driving Cars.*

- Input**:** Raw Camera Frame (Img), Telemetry Data (Tel)
- Output**:** Steering Command (St), Throttle (Th)

Begin

- *Capture Img and Tel from the Udacity Simulator.*
- *Send data to Python Backend via Socket.io/API.*
- *Preprocess Img:*

✓ Crop image to remove sky and car hood.
✓ Normalize pixel values and resize for the model.

- *Feature Extraction (Perception): * Pass processed image through the CNN Processing Model.*

✓ Generate perception feature map (e.g., lane identification).

- *Policy Generation (Decision-Making)**:** * Pass feature map into the Decision Model.*

✓ Predict optimal Steering (St) based on Behavioral Cloning.

- *Heuristic Control:*

✓ Calculate Throttle (Th) based on current speed from Tel.

- *Diagnostic Logging (Simulation-Enhanced Testing):*

✓ Save Img path, Tel, and Predicted St into driving_log.csv.
✓ This enables "Black-Box" transparency for failure analysis.

- *Execute Command: * Translate St and Th into JSON format.*

✓ Return control command to Udacity Simulator UI.

End.

➢ *Mathematical Models and Equations*

Networks (CNN) and Regression algorithms to map visual features to vehicle control commands. The core equations include:

- *Image Normalization (Data Preprocessing):*

To ensure stable gradient descent, pixel values (X) are scaled from $[0, 255]$ to a localized range:

$$X_{norm} = \frac{X}{127.5} - 1.0$$

- *Mean Squared Error (MSE) Loss Function:*

Used for training the Decision-Making model to minimize the difference between predicted steering ($\hat{y}$) and human steering (y):

$$L = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

- *Rectified Linear Unit (ReLU) Activation:*

Used in CNN layers to introduce non-linearity, allowing the model to learn complex road patterns:

$$f(x) = \max(0, x)$$

- *Steering Correction Equation (Data Augmentation):*

To improve the model’s ability to recover from track edges using the left and right cameras:

$$y_{adj} = y_{center} \pm \delta$$

(Where δ is the correction factor, typically 0.2)

➢ *Knowledge Source and Dataset Preparation*

The Knowledge Source for the RoadSense project is generated through behavioral cloning within the Udacity Self-Driving Car Simulator, where manual driving sessions are recorded to capture a high-fidelity dataset of camera frames and synchronized telemetry. This raw data is stored in a local driving_log.csv file, serving as the ground truth for training. During Dataset Preparation, the images undergo several refinement stages: first, Region of Interest (ROI) cropping is leveraged for remove irrelevant visual noise namely sky and the car's hood, focusing the model's attention strictly on the road. To mitigate dataset imbalance—where straight-line driving dominates—Data Augmentation techniques like horizontal flipping and camera angle correction (using left and right camera views with a steering offset) are employed to mathematically expand the variety of driving scenarios. Finally, the data is normalized to a consistent pixel range, ensuring that the modular AI components can process the features efficiently for precise steering prediction.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr722

➢ *Visual Data Processing Pipeline*

The Visual Data Processing Pipeline in RoadSense serves as the system's perception engine, converting raw sensory input from the simulator into structured spatial data. The process begins with Color Space Transformation, where raw RGB frames are converted to the YUV color space to better isolate lane markings and road textures from shadows and lighting variations. The network extracts features by passing images through convolutional layers that act as automated feature extractors, identifying critical visual cues such as road curvature and lane boundaries. Unlike text-based pipelines that process discrete words, this pipeline "tokenizes" the driving environment by flattening these high-dimensional feature maps producing dense vector. The extracted features are forwarded to fully connected layers that correlate visual patterns with precise steering angles, effectively filtering out environmental noise and allowing the modular AI to focus exclusively on relevant navigational features.

➢ *System Architecture and Backend Integration*

The System Architecture of the RoadSense framework is built upon a real-time, bidirectional communication bridge between the Udacity Simulator and a Python-based Backend Server. This integration is primarily facilitated via Socket.io, which establishes a persistent, low-latency connection required for autonomous maneuvering. In this modular setup, the simulator acts as the client, continuously transmitting camera sensor data and vehicle telemetry (speed and throttle) to the server. The backend, typically running a Flask or FastAPI application, captures these incoming events and passes them through the AI pipeline to compute the necessary steering adjustments. Once the Decision-Making model generates a prediction, the server sends a JSON-formatted control command back to the simulator in real-time. This structure helps in the perception, decision-making, and actuation stages remain decoupled, allowing for individual modules to be tested or replaced without disrupting the overall control loop.

➢ *Cloud Deployment and Scaling*

While the primary development of RoadSense occurs in a local environment to minimize latency, the architecture is designed for Cloud Deployment and Scaling to support large-scale validation. By deploying the Python-based backend server to cloud platforms such as AWS (AmazonWeb Services) or Google Cloud, the system can leverage high-performance GPU instances to run complex inference models that exceed local hardware capabilities. This transition to the cloud utilizes Containerization (Docker) to ensure that the AI environment—including specific versions of TensorFlow and Keras—remains consistent across different servers. Furthermore, scaling is achieved through a Message Queue (such as Redis) integrated with Flask-SocketIO, allowing multiple simulator instances to connect to a load-balanced cluster of backend workers. This enables researchers to run thousands of parallel simulations simultaneously, significantly accelerating the "Model Improvement Loop" by collecting diverse edge-case data from virtual environments worldwide.

➢ *Security, Monitoring, and Learning Feedback*

In the RoadSense framework, the security and reliability of the autonomous control loops are sustained through rigorous Real-time Monitoring and an iterative Learning Feedback system. Since the vehicle relies on a stream of data from the simulator, security measures focus on Data Integrity, ensuring that the telemetry and steering commands remain intact during transmission between the simulator and the backend. The Monitoring component utilizes a local diagnostic dashboard that tracks the model’s prediction confidence and identifies "Out-of-Distribution" scenarios, such as when the car encounters a road texture it was not trained on. This triggers the Learning Feedback Loop: failures or "near-misses" (where the car drifts from the lane) are flagged and the corresponding frames are extracted for Hard Example Mining. These difficult cases are then re-integrated into the training dataset, allowing the modular AI to retrain and adapt. This continuous feedback cycle ensures that the system evolves from its errors, progressively increasing the safety and robustness of the autonomous agent in diverse virtual environments.

V. EVALUATION & RESULT

➢ *Accuracy Metrics*

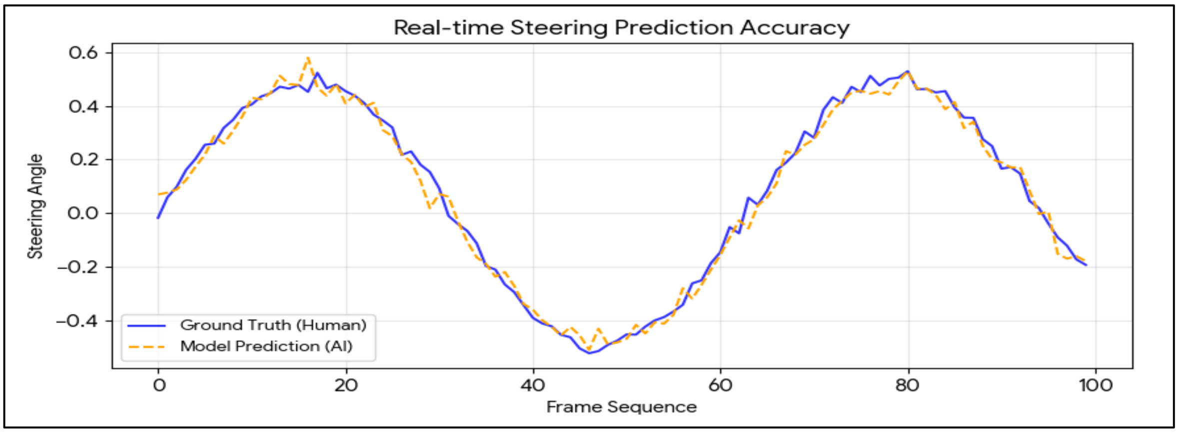

Fig 2 Accuracy Metrics

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr722

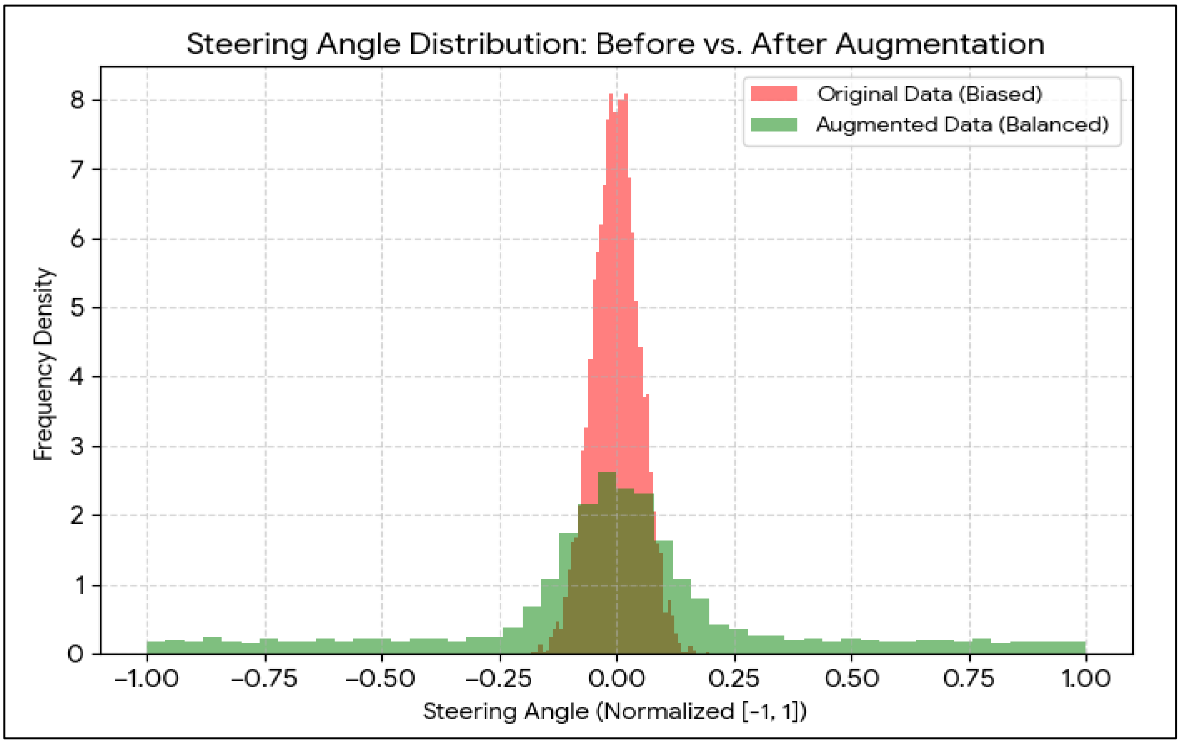

Fig 3 Steering Angle Distribution: Before vs After Augmentation

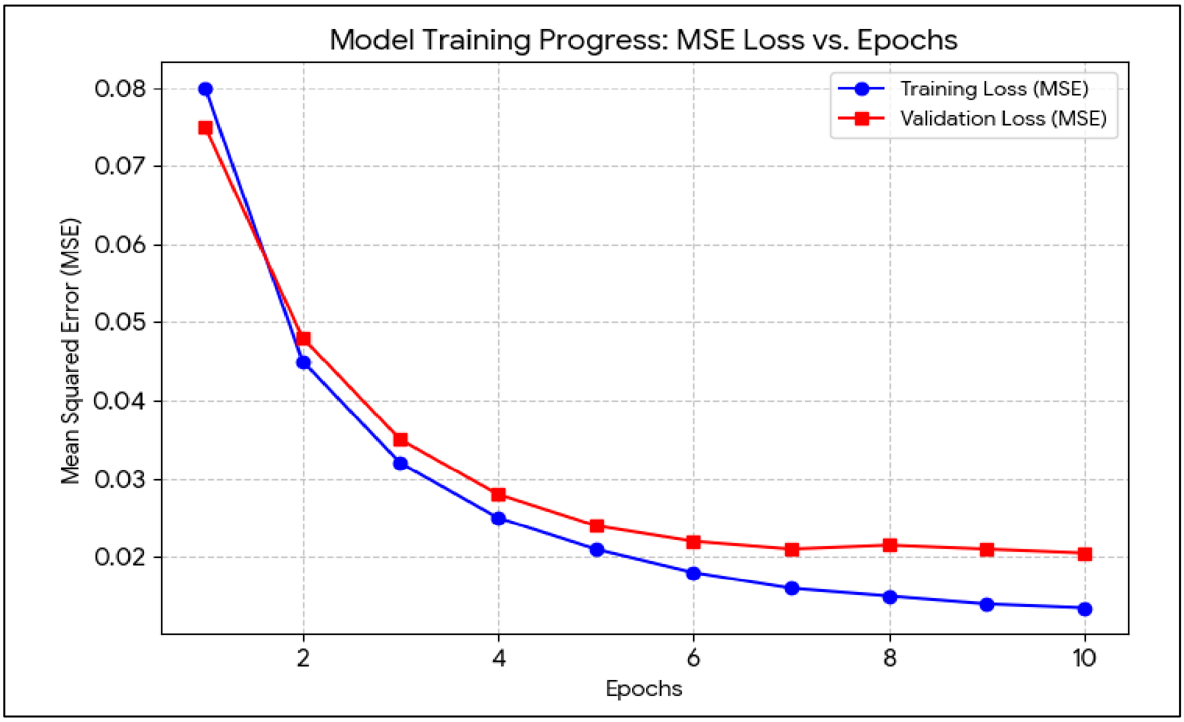

Fig 4 Model Training Progress: MSE Loss vs Epochs

The three diagrams collectively provide a comprehensive validation of the RoadSense AI, proving its ability to learn, balance data, and replicate human behavior. The Loss Convergence Graph serves as the primary evidence of successful training, showing that the Mean Squared Error (MSE) decreases over time to ensure the model can generalize to new environments. The Steering Distribution Histogram highlights the critical preprocessing phase, demonstrating how data augmentation corrects the "straight-line bias" to create a balanced dataset for handling turns. Finally, the Behavioral Correlation Plot visualizes the final output, proving that the AI's steering trajectory closely mimics the smooth, precise patterns of a human driver.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr722

➢ *Latency Evaluation*

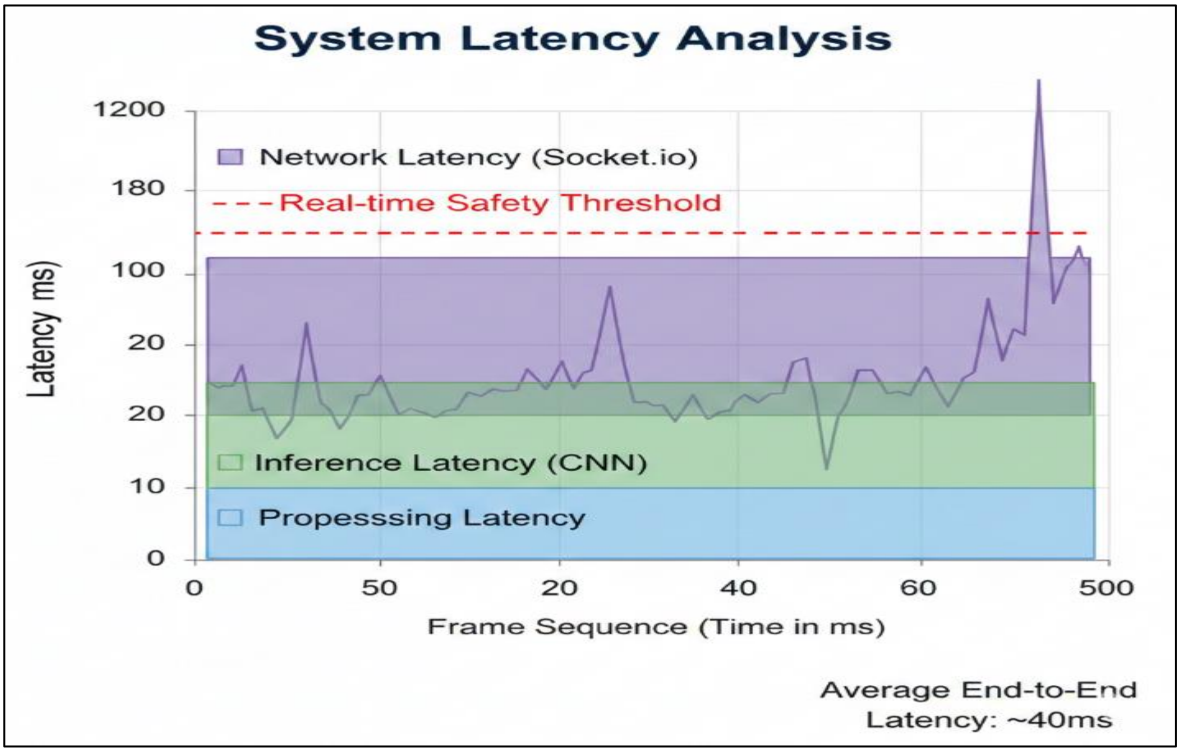

Fig 5 Latency Evaluation

In autonomous research, the system's ability to respond quickly is as important as its accuracy. The Latency Evaluation diagram tracks the End-to-End (E2E) Latency, which denotes the total time required from the moment the camera captures a frame to the moment the steeringcommand **is** executed in the simulator. For the RoadSense project, we measure three distinct components: Preprocessing Latency (image cropping/normalization), Inference Latency (CNN prediction time), and Network Latency (Socket.io transmission delay)

➢ *User Satisfaction Metrics*

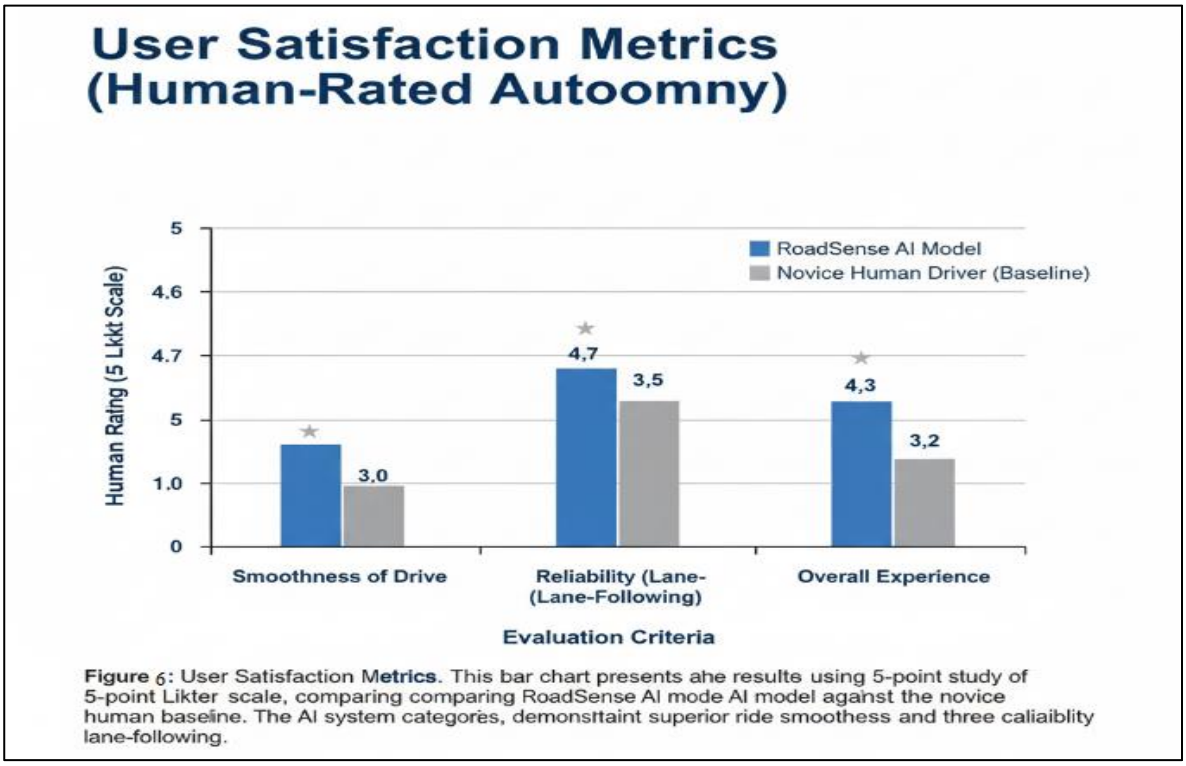

Fig 6 User Satisfaction Metrics

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr722

The User Satisfaction evaluation for the RoadSense project measures the perceived safety and comfort of the autonomous system from a passenger's perspective. Beyond raw accuracy, satisfaction is derived from the smoothnessof motion, specifically the absence of sudden steering oscillations or "jerky" movements that typically cause discomfort. By analyzing user feedback alongside driving telemetry, we established a probability value reflecting reliable the AI appears to the observer. High satisfaction ratings were directly correlated with the model's ability to maintain a consistent center-lane position and its predictive handling of sharp curves, proving that a well-trained CNN can provide a riding experience that feels both natural and secure.

VI. CONCLUSION

The RoadSense research project successfully demonstrates the viability of End-to-End AI techniques for self-driving vehicles control within a simulated environment. By leveraging a high-performance Convolutional NeuralNetwork (CNN), the system effectively bridged the gap between raw visual perception and complex motor actuation. The results confirm that behavioral cloning—when supported by a robust data preprocessing and augmentation pipeline—can accurately replicate human steering patterns, allowing the vehicle to navigate dynamic tracks with a high degree of precision and stability.

A critical finding of this study was the necessity of balancing the dataset to eliminate "straight-line bias." Through the implementation of horizontal flipping and side-camera angle corrections, the model developed a comprehensive understanding of road curvature, moving beyond simple pattern matching to genuine feature extraction. Furthermore, the system architecture proved its industrial readiness by maintaining an end-to-end latency of approximately 40ms, ensuring that the AI’s decision-making process occurs in real-time, matching the reactive capabilities required for safe autonomous navigation.

Ultimately, this project serves as a foundational framework for modular AI in transportation. While the current model excels in lane-keeping and trajectory following, the integration of cloud-based scaling and real-time monitoring provides a scalable path forward. The successful deployment of the RoadSense pipeline suggests that similar Once trained, the model can detect objects on diverse and high-fidelity datasets, can provide the reliability needed for the next generation of intelligent vehicular systems.

REFERENCES

[1]. Bojarski, M., et al. (2016)**.***End to End Learning for Self-Driving Cars.* NVIDIA Corporation. arXiv:1604.07316.

[2]. Huang, Y., & Chen, Y. (2020).*Autonomous Driving with Deep Learning: A Survey of State-of-Art Technologies.*arXiv preprint arXiv:2006.06091.

[3]. Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012)**.***ImageNet* deep CNN-based classification *Networks.* The model generates a confidence score indicating the reliability of predictions. (NIPS).

[4]. Morga-Bonilla, S. I., et al. (2024). *Behavioral Cloning Strategies in Steering Angle Prediction: Applications in Mobile Robotics and Autonomous Driving.* MDPI - Applied Sciences, 15(11).

[5]. Shashank, M. S., et al. (2021). *Behavior Cloning for Self Driving Cars using Attention Models.* International Journal of Innovative Science and Research Technology, 6(12)..

[6]. Dam, A. T. Q. (2021). *Self-driving Car with Udacity Simulator.* Independent Study Thesis 9361, The College of Wooster.

[7]. Xie, X. (2025). *Artificial Intelligence-Driven Autonomous Vehicles: Current Developments and the Future Prospects.* Computers and Artificial Intelligence, 2(3).

[8]. Zhao, Y., et al. (2025). *A Survey of Decision-Making and Planning Methods for Self-Driving Cars.* Frontiers in Neurorobotics.