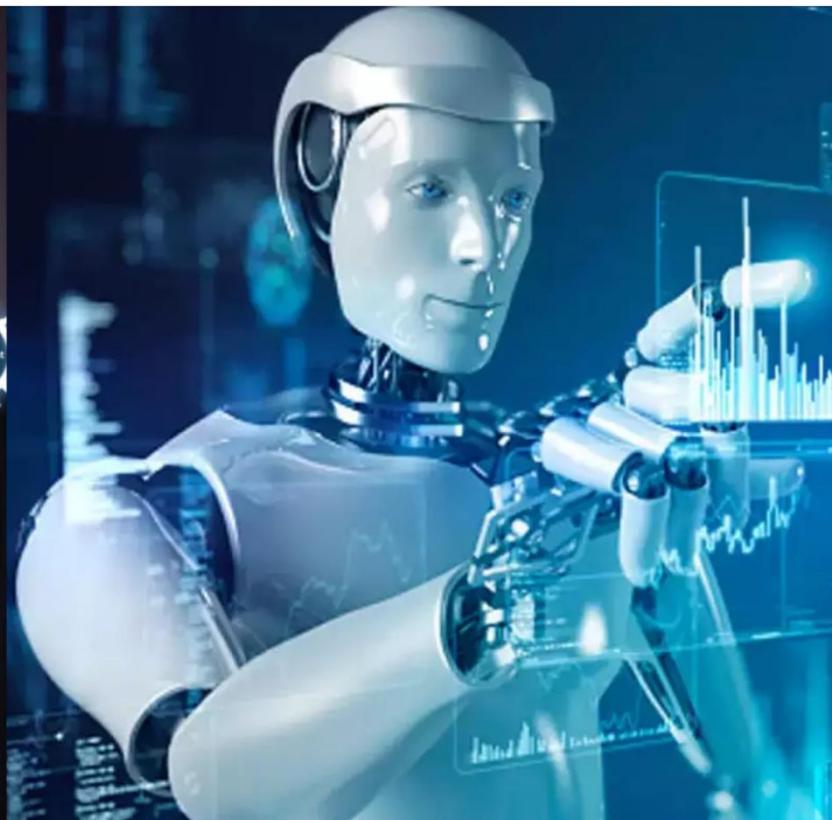




International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

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



Impact Factor: 8.699

Volume 15, Issue 3, March 2026

Vehicle Maintenance System using Digital Twin and Machine Learning

Dr. R. Suresh¹, K. Gayathri², K. Nagender Reddy³, A. Venkat⁴, B. Jagadish⁵

Associate Professor and Head of the Department, Department of CSE (Data Science), ACE Engineering College,
Hyderabad, India¹

IV B.Tech, Department of CSE (Data Science), ACE Engineering College, Hyderabad, India²⁻⁵

ABSTRACT: Modern intelligent transportation systems require reliable methods for vehicle safety and predictive fault detection. This project uses Digital Twin technology and Machine Learning to continuously monitor vehicle behavior and detect potential faults early. A virtual replica of the vehicle predicts normal sensor values such as RPM, engine load, coolant temperature, battery voltage, and CAN communication patterns, which are compared with real-time data to identify deviations. The system classifies these deviations into five categories: Normal, Early Warning, Engine Fault, Battery Fault, and Sensor/CAN Fault, using machine learning models like XGBoost and Random Forest for accurate multi-class fault detection in both fuel-based and electric vehicles. A Streamlit interface enables users to upload vehicle data and instantly view predicted fault categories along with maintenance recommendations, demonstrating how integrating Digital Twins with AI can improve vehicle health monitoring, enhance safety, and reduce maintenance downtime.

I. INTRODUCTION

Vehicle maintenance is essential for ensuring the safety, reliability, and efficiency of modern transportation systems. Traditional maintenance approaches and onboard diagnostic systems often identify problems only after faults become serious, leading to unexpected breakdowns, higher repair costs, and safety risks. With the increasing availability of vehicle sensor data, advanced technologies such as Digital Twin and Machine Learning provide new opportunities for predictive and intelligent vehicle monitoring. A digital twin creates a virtual representation of a vehicle that simulates normal operating conditions, while machine learning models analyze sensor data to detect deviations and potential faults. By comparing predicted sensor values with real-time data, the system can identify abnormal patterns related to the engine, battery, or communication systems. This approach enables early fault detection, supports timely maintenance decisions, and improves overall vehicle performance. The system focuses on continuous monitoring, anomaly detection, and fault classification using vehicle sensor data, while it does not include hardware-level vehicle control or full mechanical diagnostics.

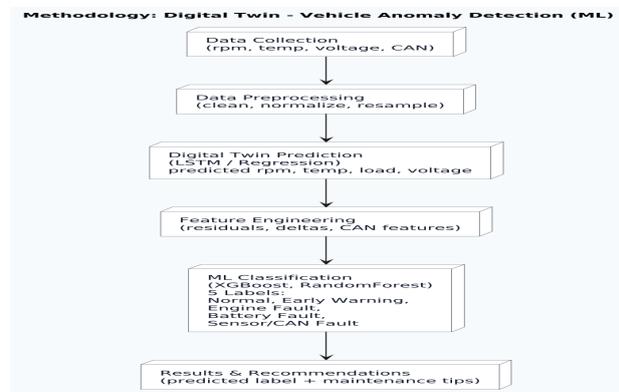
II. EXISTING SYSTEM

Current vehicle fault detection solutions are often fragmented and rely heavily on traditional onboard diagnostic (OBD) systems that operate in a reactive manner rather than a predictive one. These systems typically identify faults only after they become severe, such as engine overheating, battery degradation, or communication failures, which can lead to unexpected breakdowns, increased maintenance costs, and potential safety risks. Many existing diagnostic tools focus on specific areas such as mechanical faults or CAN network security, but rarely address both simultaneously. Additionally, most systems depend on manual inspections, rule-based thresholds, or isolated monitoring tools, which limit their ability to detect early deviations in vehicle behavior. As a result, current approaches lack integrated monitoring, predictive analysis, and comprehensive multi-class fault detection across different vehicle subsystems, making them less effective for modern fuel-based and electric vehicles.

III. PROPOSED SYSTEM

The proposed system uses a Digital Twin model integrated with Machine Learning to detect and classify vehicle anomalies. A virtual replica of the vehicle is created using historical and real-time sensor data, enabling continuous monitoring of critical parameters for both fuel-based and electric vehicles. The Digital Twin predicts normal sensor values, and the difference between predicted and actual readings is analyzed as residuals. These features are processed by XGBoost and Random Forest models, which classify the vehicle state into five labels: Normal, Early Warning, Engine Fault, Battery Fault, and Sensor/CAN Fault. A Streamlit-based web interface allows users to upload driving data and instantly view the predicted condition of the vehicle along with suggestions for maintenance or corrective action.

IV. METHODOLOGY



1. Data Pre-Processing

In this stage, raw vehicle sensor data such as RPM, engine temperature, battery voltage, engine load, and CAN communication signals are collected and prepared for analysis. The dataset is cleaned by handling missing values, removing invalid or duplicate records, and correcting inconsistent units. Sensor readings are normalized to maintain uniformity across different parameters, and noise reduction techniques are applied to eliminate abnormal spikes or fluctuations. Multi-source signals operating at different sampling rates are resampled and aligned to a common time scale. This preprocessing step ensures that the data entering the Digital Twin model is clean, consistent, and suitable for accurate prediction and analysis.

2. Digital Twin Prediction

The system then generates a **Digital Twin**, which is a virtual representation of the vehicle's normal operating behavior. Using machine learning models such as **LSTM or regression models**, the Digital Twin predicts expected sensor values including RPM, engine load, temperature, and battery voltage. These predicted values represent how the vehicle should behave under normal conditions. The predicted outputs are aligned with real-time sensor readings to maintain time-series consistency, enabling the system to simulate realistic vehicle performance patterns.

3. Feature Engineering

Once the predicted values are obtained, the system analyzes the differences between the **actual sensor readings and Digital Twin predictions**. These differences, known as **residuals**, help identify unusual patterns or deviations in vehicle behavior. Additional features such as delta changes, rate-of-change values, statistical indicators (mean, variance, and moving averages), and CAN communication features are extracted. These engineered features provide deeper insights into system performance and help the machine learning models better distinguish between normal operation and abnormal conditions.

4. Machine Learning Classification

The engineered feature set is then processed using **machine learning classification models such as Random Forest and XGBoost**. These models analyze the patterns within the data and classify the vehicle condition into one of five categories: **Normal, Early Warning, Engine Fault, Battery Fault, and Sensor/CAN Fault**. Each prediction is associated with a confidence score that indicates the reliability of the classification. This stage enables accurate identification of multiple types of vehicle anomalies based on data-driven learning.

5. Results and Recommendation Integration

In the final stage, the predicted vehicle condition is presented through a **Streamlit-based web interface**. Users can upload vehicle datasets and instantly receive diagnostic results along with suggested maintenance actions. The system displays fault categories, confidence scores, and visual summaries to help users understand the vehicle’s health status. This integration connects backend Digital Twin analytics and machine learning models with an accessible user interface, enabling drivers, technicians, and fleet managers to make informed maintenance decisions.

V. SYSTEM ARCHITECTURE

The system architecture of the Digital Twin–based Vehicle Maintenance System follows a modular processing pipeline that analyzes vehicle sensor data through multiple stages. The system begins with users uploading a vehicle dataset containing parameters such as RPM, temperature, voltage, load, and CAN signals through a web interface. The data first passes through a preprocessing module, where it is cleaned, normalized, and structured for analysis. The processed data is then used in the model building stage, where machine learning algorithms such as Random Forest and XGBoost are trained and saved as a model file for fault prediction. During system execution, the backend sends the input data to the trained model, which analyzes the sensor patterns and predicts potential vehicle faults. Finally, the predicted fault category along with maintenance recommendations is sent back to the frontend interface, allowing users to easily view diagnostic results. The modular architecture ensures efficient data flow, scalability, and easy integration for both fuel vehicles and electric vehicles.

VI. RESULTS AND OUTPUT

Step 1: Start the Application

Run the Flask application and open a web browser. Navigate to **http://127.0.0.1:5000** to access the vehicle anomaly detection system interface.

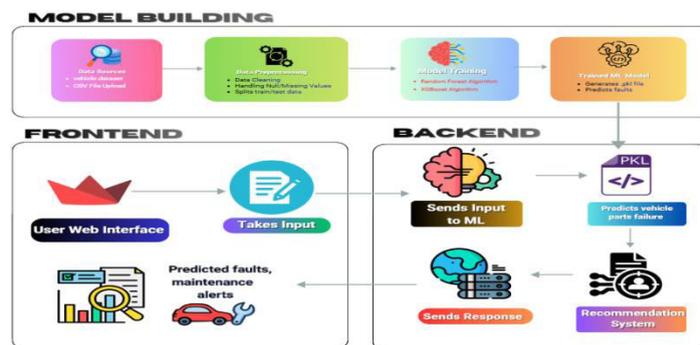




Fig : 6.1 Application

Step 2: Upload Dataset

Click on “Choose File” and select the vehicle dataset in CSV format (e.g., latest_vehicle_dataset.csv). The dataset should include sensor parameters such as Speed, RPM Temperature, Engine Load, Battery Voltage, and CAN bus signals.

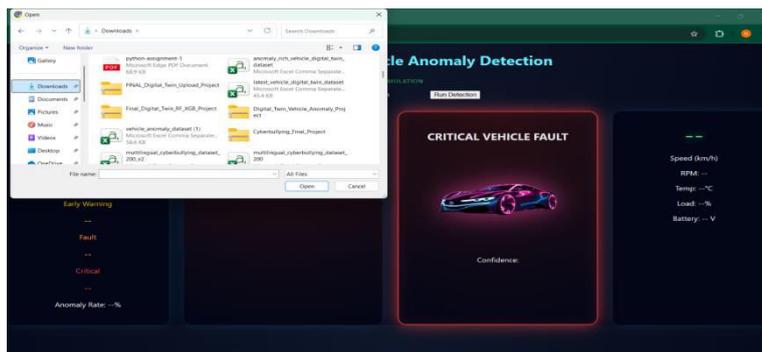


Fig : 6.2 Upload Dataset

Step 3: Confirm File Selection

After selecting the file, the file name appears next to the Choose File option, confirming that the dataset has been successfully uploaded and is ready for analysis.

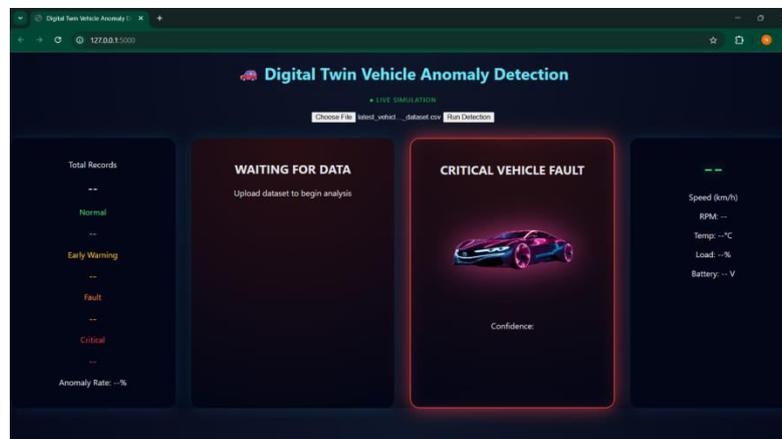


Fig : 6.3 Confirm File Selection

Step 4: Run Detection

Click “Run Detection” to start the analysis. The backend processes the dataset by reading all records, performing pre-processing and normalization, comparing actual sensor values with Digital Twin predictions, and classifying anomalies using Machine Learning models.

Step 5: View Detection Results

Once processing is complete, the dashboard displays the results. It shows the total records analyzed, counts of Normal, Early Warning, Fault, and Critical cases, and the overall anomaly rate. The system also presents the vehicle condition status, confidence score, and reasons for detected anomalies, along with key telemetry values such as Speed, RPM, Temperature, Load, and Battery Voltage.

Step 6: Analyze Output

Significant deviations in sensor values or CAN communication irregularities trigger alerts. Critical faults are highlighted clearly, allowing users to quickly identify potential problems and take preventive maintenance actions to avoid vehicle breakdowns.

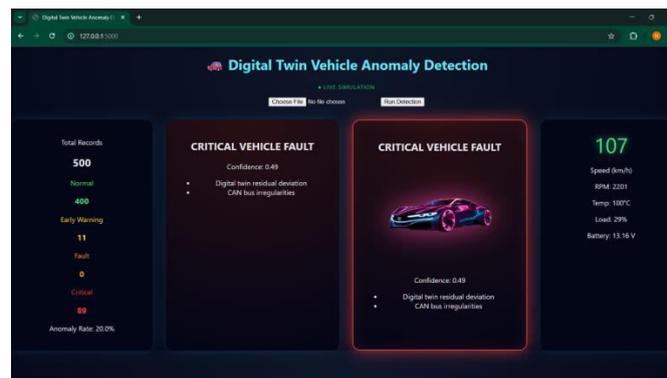


Fig : 6.4 Output

VII. CONCLUSION

The Digital Twin-based Vehicle Anomaly Detection system demonstrates an effective integration of virtual modeling and machine learning to improve vehicle health monitoring. By creating a digital representation of the vehicle and continuously generating predicted sensor values, the system can identify even minor deviations early. This approach offers a more accurate method of detecting anomalies compared to traditional threshold-based systems, enabling predictive maintenance rather than reactive repair.

Machine learning algorithms such as Random Forest and XGBoost further enhance the system by classifying sensor data into five distinct fault categories: Normal, Early Warning, Engine Fault, Battery Fault, and Sensor/CAN Fault. These models analyze processed features, including Digital Twin residuals and CAN communication indicators, making the system capable of detecting both mechanical and communication-related issues. The solution also supports both fuel-based and electric vehicles, showcasing its versatility across modern automotive platforms.

Overall, the system provides a reliable and user-friendly method for real-time vehicle fault prediction. By visualizing results through a Streamlit web interface, users receive instant feedback and maintenance recommendations. This leads to improved safety, reduced breakdowns, lower maintenance costs, and better vehicle performance. The combination of Digital Twin technology and machine learning makes this approach a strong foundation for future advancements in intelligent automotive diagnostics.

Furthermore, the integration of Digital Twin predictions with machine learning classification demonstrates the potential for scalable, real-time vehicle diagnostic solutions. As more data is collected, the system can continuously improve its prediction accuracy, enabling advanced analytics such as fault severity estimation, remaining useful life (RUL) prediction, and automated maintenance scheduling. This project serves as a foundation for future enhancements like onboard deployment, cloud-based monitoring, and integration with IoT and telematics systems, ultimately contributing to smarter.

VIII. FUTURE SCOPE

i. Real-Time Vehicle Data Integration

The system can be extended to process real-time vehicle data instead of relying only on offline CSV datasets. By connecting directly to the vehicle's CAN bus or onboard sensors, the system can continuously monitor parameters such as RPM, temperature, engine load, and battery voltage. This will enable instant anomaly detection and faster response to potential vehicle faults.

ii. Advanced Digital Twin Modeling

Future improvements can focus on building more accurate Digital Twin models using advanced simulation techniques and data-driven approaches. Combining physics-based vehicle models with deep learning algorithms can help create highly realistic virtual replicas of vehicle behavior, allowing more precise prediction of sensor values and early detection of subtle anomalies.

iii. Enhanced Machine Learning Models

The system can integrate advanced deep learning models such as LSTM or Transformer networks to better analyze time-series vehicle sensor data. These models can capture complex temporal patterns in vehicle operation, improving the accuracy of anomaly detection and fault classification across different vehicle components.

iv. Predictive Maintenance

The system can be extended to support predictive maintenance, where the remaining useful life (RUL) of vehicle components such as engines, batteries, and sensors is estimated. This would allow vehicle owners and fleet managers to schedule maintenance proactively before failures occur, reducing downtime and maintenance costs.

v. Cloud-Based Deployment

Deploying the system on **cloud platforms** can enable remote monitoring and large-scale vehicle analytics. Cloud integration would allow centralized data storage, faster processing, and the ability to monitor multiple vehicles simultaneously, making the system suitable for fleet management applications.

vi. IoT and Edge Computing Integration

Integration with IoT devices and edge computing systems can enable real-time anomaly detection directly within the vehicle environment. Edge processing reduces latency and allows critical fault detection even without continuous internet connectivity, improving reliability and response time.

vii. Automated Alerts and Notifications

Future versions of the system can include automated alert mechanisms such as SMS, email, or mobile app notifications. When critical anomalies are detected, the system can instantly notify drivers, vehicle owners, or service centers, enabling faster corrective actions and improving overall vehicle safety.

REFERENCES

- [1] Digital Twin-Based Predictive Maintenance for Industrial Systems
URL: <https://ieeexplore.ieee.org/document/9428366>
- [2] Machine Learning-Based Intrusion Detection for In-Vehicle CAN Bus Networks
URL: <https://ieeexplore.ieee.org/document/9443586>
- [3] A Review of Machine Learning Methods for Automotive Fault Diagnosis

URL: <https://ieeexplore.ieee.org/document/9264813>

[4] Digital Twin Driven Smart Manufacturing: Framework and Key Technologies

Reference for Digital Twin implementation concepts.

URL: <https://ieeexplore.ieee.org/document/9125313>

[5] LSTM-Based Vehicle Sensor Anomaly Detection Using Multivariate Time-Series Data

URL: <https://ieeexplore.ieee.org/document/9896541>

[6] XGBoost: A Scalable Tree Boosting System

Reference for XGBoost algorithm used in your classifier.

URL: <https://arxiv.org/abs/1603.02754>



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



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