International Journal of Engineering Innovation, Research and Development (IJEIRD)

A Novel Generative AI-Based Framework for Resilient False Data Injection Attack Mitigation in Autonomous Vehicles

Memoona Sadaf*
Murdoch University, Murdoch WA 6150, Australia
*Corresponding Author: memoonasadaf29@gmail.com

HIGHLIGHTS

- A generative-AI framework is proposed for detecting and mitigating false data injection attacks in autonomous vehicles.
- Incoming sensor data are classified as legitimate, partially compromised, or fully compromised.
- The framework targets low-latency operation and reduced computational overhead for intelligent driving environment.
- Experimental evaluation examines attack detection, prevention levels, latency, data loss, and error rate.

ABSTRACT

Every year, millions of people die or are injured due to road traffic accidents; the most common reasons for these accidents are overspeeding, fatigued driving, distracted driving, and drugs or narcotics. To reduce these risks, several research studies are taking place in vehicle automation. The impacts of Autonomous vehicles (AVs) on contemporary transportation depend on how they can change the status of transportation, including human errors, inefficiencies, and traffic congestion. However, like all sophisticated IT systems, AVs remain vulnerable to remote compromise, which can pose significant risks to their safety, security, and users' personal privacy. The most severe threats are disc-shaped False Data Injection (FDI) attacks. These are false or intentionally submitted data designed to influence critical decisions or the operation of an AV. Such a data grab could compromise AV users in a very significant way. Thus, efficient schemes to prevent and detect false data injection while maintaining minimal intervention in the vehicle dynamics are desirable. In the case of FDI attacks, the data loss rate is reduced. Despite the challenges, Generative Artificial Intelligence (Generative AI) presents a novel approach to address the problem of malicious data filtration while processing only accurate and clean data in real time. The existing FDI detection techniques have shortcomings such as output delay and high algorithmic cost, with large demand for machine resources and high latency, which contradict the requirements of intelligent driving. This paper focuses on the crucial role played by Generative AI in assisting detection and fraud data injection defense in AVs. The method adopted depends on the use of AI

Available online 24 August 2026; 1(1):1-29/

algorithms that can consider interference (resource restrictions). In this paper, a generative-AI-based approach was incorporated for anomaly detection and FDI attack mitigation in AVs. AI techniques were applied to raw inputata to avoid high latency and keep a low rate of error. With the proposed classes of incoming data being legitimate, partially compromised, or fully compromised, it is possible to delineate good data, slightly compromised data, and totally bad data. This classification improves AV safety while maintaining its performance level. In general, on comparing the presented solution with existing methodologies, it can be seen that more accurate results are obtained by the presented solution, and it also provides higher reliability, with less computation. Thus, substantial testing and validation verify that the proposed system does not diminish data coherence, with an acceptable compromise of system performance. In addition, this approach can be generalized for detection of other threats in AVs by adding more input variables and adjusting the Generative AI model for counteracting various types of malicious data injection. Thus, the proposed framework can be regarded as significant progress towards ensuring the sustainability of autonomous transportation systems against modern cyber threats.

Keywords: Automotive Security; Autonomous Vehicles; False Data Injection Attack; Generative AI; Intrusion Detection Systems

1. Introduction

Autonomous vehicles (AVs) represent a major advancement in intelligent transportation by integrating sensing technologies, communication protocols, artificial intelligence, and onboard decision-making systems. Sensors such as LiDAR, radar, cameras, Global Positioning System (GPS), and ultrasonic devices continuously collect environmental and operational information that enables an AV to perform navigation, obstacle detection, route planning, lane control, and collision avoidance. The reliability of these interconnected components is therefore essential for ensuring safe and efficient autonomous driving. However, the increasing dependence of AVs on digital data, external communication channels, and intelligent control systems also exposes them to a broad range of cybersecurity threats [1][2].

Among these threats, attacks targeting sensing and communication systems are particularly dangerous because autonomous driving decisions are directly influenced by the integrity and accuracy of the received data. Sensors are responsible for perceiving nearby objects, road conditions, pedestrians, traffic signs, vehicle locations, and environmental hazards. Nevertheless, these components may be compromised through False Data Injection (FDI), spoofing, signal jamming, or communication manipulation attacks [3]. An attacker may inject carefully manipulated information into a sensor stream without causing an obvious system failure, making the malicious data appear consistent with legitimate observations. Consequently, the AV may continue operating while making unsafe decisions based on corrupted information. FDI attacks are especially critical because they deliberately alter the data used by autonomous decision-making systems. Such attacks may affect navigation, obstacle avoidance, braking, acceleration, route planning, and lane-changing operations [4]. For example, manipulating GPS information may cause an AV to deviate from its intended route, overlook an intersection, or incorrectly estimate its geographical position. Similarly, injecting false information into LiDAR or camera streams may cause the vehicle to overlook pedestrians, misidentify obstacles, or fail to recognise traffic signs. These attacks can consequently result in unsafe braking, unnecessary lane changes, system disruption, collisions, or

other potentially life-threatening consequences [5][6][7]. Therefore, protecting the integrity of sensor and communication data is essential for maintaining the operational safety and reliability of AVs.

AVs also rely on continuous communication among sensors, electronic control units, decision-making components, nearby vehicles, roadside infrastructure, and cloud-based platforms. Although such connectivity improves coordination and situational awareness, it creates additional attack surfaces. A compromised communication channel can delay, manipulate, or suppress critical information before it reaches the vehicle controller. Since multiple autonomous functions depend on shared information, an attack against one component may create a cascading effect across navigation, acceleration, braking, and collision-avoidance functions [2][4]. These security breaches may endanger not only the vehicle occupants but also pedestrians, nearby drivers, and other road users [1]. Traditional intrusion and anomaly detection mechanisms commonly rely on predefined signatures, fixed thresholds, manually designed rules, or supervised classifiers. Although these approaches may detect previously observed attacks, their static nature limits their effectiveness against evolving and previously unseen FDI patterns. Attackers can introduce subtle modifications that remain within predefined operational boundaries while still influencing the behaviour of the vehicle. Moreover, conventional detection systems may generate high false-positive rates, require substantial computational resources, or fail to provide an appropriate response after identifying malicious data. These limitations are particularly problematic in AV environments, where cybersecurity mechanisms must operate continuously and make timely decisions without compromising driving performance.

Generative Artificial Intelligence, particularly Generative Adversarial Networks (GANs), provides a promising mechanism for learning complex sensor-data distributions and identifying deviations associated with malicious manipulation. A GAN typically consists of two competing neural networks: a generator and a discriminator. The generator produces synthetic samples that resemble legitimate data, whereas the discriminator attempts to differentiate between authentic and generated samples. Through adversarial training, both networks progressively improve their capabilities. In the context of AV cybersecurity, the generator can simulate realistic FDI attack patterns, while the discriminator can learn to distinguish normal operational data from manipulated or anomalous observations. This process can improve the capability of the system to recognise subtle attack patterns that may not be detected through static rules or conventional signature-based mechanisms. Existing GAN-based FDI and anomaly detection approaches generally concentrate on classifying sensor observations as normal or malicious. Although attack identification is an important security function, detection alone does not provide complete resilience. A practical AV cybersecurity solution must also determine how suspicious data should be isolated, how trustworthy information should be preserved, how vehicle functions should respond to the detected threat, and how safe operation should continue while an attack is being managed. Furthermore, many existing anomaly detection approaches operate as standalone models and do not explicitly integrate protection across the physical, sensor, communication, network, cognitive, decision, software, firmware, and Vehicle-to-Everything (V2X) layers of an AV.

This limitation establishes a significant research gap. There is a need for an integrated cybersecurity framework that extends beyond standalone GAN-based anomaly classification and combines intelligent attack detection with mitigation, secure information handling, resilient decision-making, and continued vehicle operation. Such a framework should support the identification of malicious inputs, prevent compromised data from propagating through the vehicle architecture, and guide the autonomous system towards a safe operational response. It

should also be sufficiently adaptive to address evolving FDI techniques and sufficiently efficient for deployment in time-sensitive AV environments. To address this gap, this study proposes a novel Generative AI-based framework for resilient FDI attack mitigation in autonomous vehicles. The proposed framework integrates GAN-driven anomaly analysis with a layered security architecture that protects the major operational components of an AV. Unlike conventional GAN-based approaches that primarily terminate their operation after detecting an anomalous sample, the proposed framework conceptualises cybersecurity as a continuous process involving data monitoring, attack identification, malicious-input isolation, mitigation, and resilient decision support. The generator is used to produce realistic FDI scenarios that strengthen the learning process, whereas the discriminator evaluates incoming data and identifies observations that deviate from legitimate operational patterns.

The proposed framework protects multiple interconnected layers of the AV ecosystem. At the physical layer, sensor redundancy and backup mechanisms support continued operation when a component is compromised or becomes unreliable [7]. At the communication layer, secure communication, encryption, latency monitoring, data-loss analysis, and transmission-error detection are used to identify and restrict manipulated information before it affects safety-critical functions [3][5]. At the network layer, intrusion detection mechanisms monitor unauthorised access and malicious traffic to protect the confidentiality and integrity of exchanged information [8]. At the sensor and cognitive layer, Generative AI models evaluate the reliability of sensor inputs and distinguish legitimate observations from potentially injected data [4]. At the decision layer, the framework evaluates the consequences of detected anomalies and supports the selection of a lower-risk operational response [6]. The V2X security layer restricts malicious or unverified information received from external vehicles, roadside infrastructure, pedestrians, and cloud services [1]. The software and firmware security layer supports the regular updating of security policies, model parameters, and detection rules to address evolving attack techniques [2]. Finally, the user-interface and alerting mechanisms communicate to identify threats and support timely intervention when required [7]. By coordinating these layers, the framework is intended to provide end-to-end resilience instead of relying on a single anomaly detection model.

The novelty of the proposed framework lies in the integration of Generative AI-based attack modelling with a comprehensive and resilient mitigation architecture. Existing GAN-based FDI detection approaches mainly use generative models to improve the distinction between normal and anomalous data. In contrast, the proposed framework places GAN-based detection within a broader security workflow that connects attack simulation, anomaly identification, data validation, mitigation, secure communication, and safety-oriented decision support. Therefore, the framework does not treat the detection output as the final objective; rather, it uses the detection outcome to support appropriate mitigation actions and maintain safe vehicle operation under adversarial conditions.

The major contributions of this study are summarised as follows:

- A novel Generative AI-based cybersecurity framework is proposed for the detection and resilient mitigation of FDI attacks in autonomous vehicles.
- The proposed approach extends beyond conventional GAN-based anomaly detection by combining attack identification with malicious-data isolation, mitigation, resilient decision support, and continued safe operation.
- A layered security architecture is introduced that integrates physical, communication, network, sensor, cognitive, decision-making,

V2X, software, firmware, and user-interface protection within a unified framework.

- GAN-based adversarial learning is incorporated to generate realistic attack patterns and improve the ability of the detection mechanism to distinguish legitimate sensor behaviour from injected or manipulated observations.
- The framework emphasises adaptive threat handling by supporting the analysis of evolving FDI patterns rather than relying exclusively on fixed attack signatures, static thresholds, or predefined rules.
- The proposed methodology incorporates cybersecurity and operational resilience into the same architecture, thereby distinguishing the framework from approaches that focus only on reporting whether an observed data sample is normal or malicious.
- The framework is designed with real-time AV requirements in mind, including timely anomaly analysis, efficient threat handling, secure inter-component communication, and scalable protection across multiple vehicle layers.

By combining Generative AI, layered cybersecurity, and resilience-oriented mitigation, the proposed framework aims to strengthen the ability of AVs to operate safely in the presence of FDI attacks. The study contributes towards bridging the gap between conventional anomaly detection and comprehensive cyber-resilience by providing a unified framework that not only identifies manipulated information but also supports appropriate protective and operational responses. This approach may facilitate the development of more secure, adaptive, and trustworthy autonomous transportation systems.

2. Literature Review

The rapid adoption of AVs has transformed modern transportation by enabling intelligent perception, autonomous decision-making, and Vehicle-to-Everything (V2X) communication. Despite these technological advancements, the increasing dependence on interconnected sensors, communication networks, and artificial intelligence has significantly expanded the cybersecurity attack surface of AVs. Among various cyber threats, FDI attacks have emerged as one of the most challenging attacks because they manipulate sensor observations or communication messages while preserving the normal operational behaviour of the system. Such attacks may mislead perception modules, influence decision-making algorithms, and ultimately compromise vehicle safety. Consequently, extensive research has been devoted to developing robust mechanisms for detecting, preventing, and mitigating FDI attacks. Existing solutions can generally be categorized into traditional security mechanisms, machine learning approaches, deep learning techniques, and more recently, Generative AI-based methods.

a. Traditional Security Mechanisms for FDI Protection

Early cybersecurity solutions for autonomous vehicles primarily relied on conventional security mechanisms, including firewalls, cryptographic protocols, authentication schemes, Intrusion Detection Systems (IDS), and trust management frameworks. These approaches were originally designed to protect computer networks from external cyber threats by restricting unauthorized access, authenticating communicating entities, and monitoring abnormal network traffic [9][10]. Firewalls remain an important first line of defence by filtering incoming and outgoing network traffic according to predefined security rules. Similarly, encryption and authentication protocols protect communication channels against message tampering and impersonation attacks. Intrusion Detection Systems monitor network behaviour to identify suspicious communication patterns, whereas trust-based mechanisms estimate the reliability of communicating entities using behavioural observations and historical interactions [11][12].

Although these techniques improve communication security, they exhibit several

limitations when applied to autonomous vehicle environments. First, conventional security mechanisms mainly monitor network traffic rather than evaluating the semantic correctness of sensor observations. Consequently, an attacker can inject carefully manipulated sensor measurements that appear statistically normal while still influencing the vehicle's perception and control systems. Second, rule-based IDS solutions generally depend on predefined attack signatures and therefore struggle to identify previously unseen attack strategies. Third, static threshold-based methods require extensive manual tuning and often generate high false-positive rates under dynamic driving conditions. Finally, these approaches typically operate independently without considering the strong dependency among perception, communication, decision-making, and vehicle control modules. Because FDI attacks directly manipulate sensor measurements instead of violating network protocols, conventional cybersecurity mechanisms alone cannot provide sufficient protection for modern AV systems. These limitations have motivated researchers to investigate intelligent data-driven approaches capable of learning complex behavioural patterns and identifying subtle anomalies beyond manually designed security rules.

b. Machine Learning-Based FDI Detection

Machine learning (ML) has emerged as an effective alternative to conventional rule-based cybersecurity solutions by enabling automatic learning from historical sensor observations. Instead of relying on manually defined attack signatures, ML algorithms construct decision boundaries using statistical relationships within the training data. Consequently, they demonstrate improved capability for detecting previously unseen anomalies while reducing dependence on handcrafted security rules. Several supervised learning algorithms, including Decision Trees, Random Forests, Support Vector Machines (SVMs), k-Nearest Neighbours (k-NN), Logistic Regression, Gradient Boosting, and XGBoost, have been investigated for FDI detection in cyber-physical systems and autonomous vehicles [13][14]. These methods generally extract handcrafted features from sensor measurements, communication traffic, or Controller Area Network (CAN) messages before performing binary or multi-class classification.

Compared with traditional IDS solutions, ML-based approaches provide higher detection accuracy and improved adaptability because they continuously learn statistical relationships between normal and malicious behaviours. Random Forest and ensemble-based classifiers have demonstrated strong robustness against noisy sensor observations, while Support Vector Machines remain effective for small and moderately sized datasets due to their strong generalization capability. Despite these improvements, conventional ML techniques exhibit several important limitations. Their performance heavily depends on carefully engineered input features, making them sensitive to feature selection and domain expertise. Furthermore, most supervised learning models require large quantities of accurately labelled attack data, which are often unavailable in real-world autonomous vehicle environments. Another challenge arises from their limited capability to model highly nonlinear relationships among heterogeneous sensor modalities such as LiDAR, radar, cameras, GPS, and inertial sensors. Consequently, traditional ML algorithms often experience degraded performance under complex driving environments and sophisticated attack scenarios. Moreover, many ML-based approaches perform attack detection as an isolated classification task without considering how detected anomalies should be mitigated or how safe vehicle operation should continue following attack identification. These shortcomings have encouraged researchers to investigate deep learning architectures capable of learning hierarchical feature representations directly from raw sensor observations.

c. Deep Learning-Based FDI Detection

The recent advancement of deep learning has substantially improved cybersecurity solutions for autonomous vehicles by enabling automatic feature extraction from high-dimensional sensor data. Unlike conventional machine learning algorithms, deep neural networks learn hierarchical representations directly from raw observations, thereby reducing the dependence on handcrafted features and improving generalization performance. Convolutional Neural Networks (CNNs) have been widely adopted for processing camera images, LiDAR point clouds, and spatial sensor representations because of their ability to capture local spatial correlations. Similarly, Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) architectures have demonstrated promising performance in analysing sequential sensor measurements and CAN bus messages by modelling temporal dependencies within vehicle operations [15][16]. Autoencoders and Variational Autoencoders (VAEs) have also attracted considerable attention for anomaly detection because they learn compact latent representations of normal system behaviour. During inference, anomalous observations typically generate larger reconstruction errors than legitimate samples, enabling unsupervised identification of abnormal sensor measurements without requiring extensive labelled datasets. More recently, Transformer architectures have further improved anomaly detection through self-attention mechanisms capable of modelling long-range dependencies across heterogeneous sensor streams.

Although deep learning methods significantly outperform conventional machine learning techniques in many benchmark datasets, several challenges remain unresolved. First, most existing models are optimized primarily for anomaly detection accuracy rather than overall cyber resilience. Their outputs are generally limited to binary attack predictions without providing mechanisms for attack mitigation, recovery, or adaptive decision support. Second, deep neural networks require large computational resources for training and inference, making deployment on resource-constrained edge devices challenging. Third, their performance often deteriorates when encountering previously unseen attack strategies, distribution shifts, sensor failures, or adversarial manipulations. Furthermore, many deep learning models operate as standalone detection systems and rarely consider coordinated protection across sensing, communication, networking, decision-making, and vehicle control layers. These limitations indicate that although deep learning has substantially improved FDI detection capabilities, further research is required to develop intelligent cybersecurity frameworks that combine accurate anomaly detection with adaptive mitigation, resilient system operation, and real-time deployment requirements. This need has recently motivated the investigation of Generative AI models, particularly GANs, which are capable of modelling complex sensor-data distributions and generating realistic attack scenarios for improving detection robustness.

d. Generative Adversarial Networks for FDI Detection

The limitations of conventional machine learning and deep learning approaches have motivated researchers to investigate Generative AI models for cybersecurity applications. Among these models, GANs have received significant attention because of their ability to learn complex data distributions, generate realistic synthetic samples, and improve anomaly detection performance. A GAN consists of two competing neural networks: a generator and a discriminator. The generator attempts to produce synthetic samples that resemble legitimate sensor observations, whereas the discriminator learns to distinguish genuine data from artificially generated samples. Through adversarial learning, both networks continuously improve, enabling the discriminator to recognize subtle anomalies that may remain undetected by conventional classifiers.

Recent studies have successfully applied GANs to FDI detection in cyber-physical systems, smart grids, Internet of Things (IoT), and autonomous vehicle environments. Instead of relying solely on predefined attack signatures, GANs learn the underlying distribution of legitimate sensor data and identify deviations introduced by malicious manipulation. Furthermore, synthetic attack samples generated by GANs help alleviate data imbalance problems and improve model robustness against limited training data. Consequently, GAN-based methods generally achieve higher detection accuracy than conventional machine learning approaches, particularly when sophisticated attack patterns closely resemble legitimate observations. Several researchers have combined GANs with CNNs, recurrent neural networks, and autoencoder architectures to improve anomaly detection performance. CNN-GAN hybrid models have demonstrated improved capability for analysing spatial sensor information obtained from cameras and LiDAR systems, whereas LSTM-GAN architectures effectively capture temporal dependencies within sequential CAN messages and communication traffic. Similarly, adversarial autoencoders have been employed to improve latent feature representation and reduce reconstruction errors associated with anomalous observations. These hybrid architectures demonstrate that adversarial learning can significantly enhance feature discrimination while improving generalization capability under complex driving environments. Despite these promising developments, current GAN-based FDI detection methods still exhibit several important limitations. First, the majority of existing studies primarily formulate cybersecurity as a binary anomaly detection problem. Once malicious sensor measurements are identified, little attention is given to how compromised information should be isolated, how vehicle behaviour should adapt, or how safe operation should continue during an ongoing attack. Consequently, attack detection is often treated as the final objective rather than one component of a comprehensive cyber-resilience strategy.

Second, most GAN-based detection models focus exclusively on individual sensor modalities or specific communication channels. For example, many studies independently analyse CAN traffic, LiDAR observations, GPS measurements, or camera images without considering the interdependency among multiple sensing and communication layers. Such isolated analysis may reduce robustness when attackers simultaneously target multiple components within the autonomous vehicle architecture. Third, many existing GAN-based approaches are evaluated under relatively constrained experimental conditions using limited attack scenarios. Their performance under unseen attacks, adversarial perturbations, sensor failures, noisy environments, communication delays, and dynamic traffic conditions remains insufficiently investigated. Consequently, the practical generalization capability of these methods in real-world autonomous driving environments remains uncertain. Furthermore, existing GAN-based methods generally prioritize detection accuracy while providing limited discussion regarding computational efficiency, inference latency, scalability, memory consumption, and deployment feasibility on resource-constrained embedded vehicle platforms. Since autonomous vehicles require strict real-time operation, computational performance is equally important as detection accuracy for practical deployment.

e. Generative AI for Autonomous Vehicle Cybersecurity

The emergence of Generative AI has considerably expanded the role of artificial intelligence in cybersecurity beyond traditional anomaly detection. Unlike discriminative learning algorithms that simply classify observations as normal or malicious, Generative AI models are capable of modelling complex probability distributions, generating realistic attack scenarios, simulating adversarial behaviours, and supporting adaptive learning under evolving threat

environments. These characteristics make Generative AI particularly attractive for autonomous vehicles, where attackers continuously develop increasingly sophisticated FDI strategies. Recent research has explored various Generative AI models, including GANs, VAEs, diffusion models, and foundation models, for enhancing cyber defence. These approaches enable the generation of realistic synthetic attack samples that improve model training, reduce class imbalance, and expose detection systems to a wider range of potential attack patterns before real deployment. Such capability is particularly valuable because collecting comprehensive real-world FDI datasets remains expensive, time-consuming, and often impractical.

Generative AI also supports proactive cybersecurity by enabling the simulation of future attack behaviours before they occur in operational environments. Instead of merely reacting to previously observed attacks, intelligent systems can continuously improve their understanding of emerging attack strategies through adversarial learning and synthetic data generation. Consequently, Generative AI provides greater adaptability than conventional supervised learning techniques that rely exclusively on historical attack samples. Nevertheless, despite these advantages, existing applications of Generative AI remain largely detection-oriented. Most studies investigate how synthetic data can improve classifier performance but pay comparatively little attention to integrating Generative AI within a complete cybersecurity architecture capable of supporting attack mitigation, resilient decision-making, secure communication, recovery planning, and continuous vehicle operation. Therefore, although Generative AI significantly improves anomaly detection capability, its full potential for developing resilient autonomous vehicle cybersecurity frameworks remains largely unexplored.

f. Critical Analysis and Research Gap

The existing literature demonstrates substantial progress in applying machine learning, deep learning, and Generative AI for FDI detection. Traditional cybersecurity mechanisms provide essential communication security but struggle to identify sophisticated sensor manipulation attacks. Machine learning methods improve adaptability but depend heavily on handcrafted features and labelled datasets. Deep learning techniques automatically learn complex feature representations; however, they primarily focus on maximizing detection accuracy while requiring significant computational resources. More recently, GAN-based approaches have demonstrated improved capability for modelling complex sensor distributions and generating realistic attack scenarios.

Despite these advances, several important research gaps remain unresolved. First, most existing studies formulate FDI protection primarily as an anomaly detection problem rather than a comprehensive cyber-resilience challenge. Detection alone cannot guarantee safe autonomous driving because additional mechanisms are required to isolate compromised data, support resilient decision-making, maintain secure communication, and ensure continued vehicle operation after attack identification. Second, the majority of existing approaches operate independently at individual system components, including sensor data, communication messages, or network traffic, without providing coordinated protection across the multiple interconnected layers of autonomous vehicle architecture. Such isolated solutions may become less effective when sophisticated attackers simultaneously compromise multiple subsystems. Third, relatively few studies investigate the integration of Generative AI with layered cybersecurity architectures capable of supporting end-to-end attack detection, mitigation, recovery, adaptive learning, and operational resilience. Furthermore, computational efficiency, scalability, real-time deployment, and practical implementation

challenges remain insufficiently addressed within the existing literature. These limitations motivate the development of the proposed Generative AI-based resilient cybersecurity framework presented in this study. Unlike existing GAN-based anomaly detection methods, the proposed framework integrates adversarial learning with a comprehensive multi-layer defence architecture that supports intelligent attack detection, malicious-data isolation, adaptive mitigation, resilient decision support, and secure vehicle operation. Consequently, the proposed framework extends the role of Generative AI beyond anomaly classification and establishes a unified cybersecurity strategy designed specifically for the operational requirements of autonomous vehicles.

3. Methodology

This study proposes a hybrid Generative Artificial Intelligence and fuzzy inference framework for detecting, classifying, and mitigating FDI attacks in autonomous vehicles. The framework evaluates communication reliability through three input indicators: latency, Data Loss Rate (DLR), and error rate. These indicators are processed by a fuzzy inference system to estimate the operational risk associated with each incoming observation. In parallel, a GAN learns the distribution of legitimate autonomous-vehicle communication data and generates realistic synthetic observations to improve anomaly detection. The outputs of the fuzzy inference and GAN modules are subsequently fused to determine whether the incoming data should be classified as Legitimate, Partial Drop, or Full Drop. The complete operational workflow is expressed as

$$x_t \rightarrow \text{Preprocessing} \rightarrow \begin{bmatrix}\text{Fuzzy Risk Evaluation, GAN-Based} \\ \text{Anomaly Detection}\end{bmatrix} \rightarrow \text{Decision Fusion} \rightarrow \text{Mitigation Action.}$$

where

$$x_t = [L_t, D_t, E_t]$$

represents the input observation at time t, L_t denotes communication latency in milliseconds, D_t represents the percentage of lost packets, and E_t represents the percentage of corrupted or incorrect observations.

Input Variables and Data Preprocessing

The proposed framework uses the following three input variables:

- L: communication latency measured in milliseconds;
- D: Data Loss Rate measured as the percentage of lost packets;
- E: error rate measured as the percentage of corrupted or incorrect data.

The false injection attack dataset is initially inspected for missing values, duplicate observations, inconsistent feature formats, and extreme outliers. Missing numerical values are replaced using median imputation because the median is less sensitive to extreme values than the arithmetic mean. Duplicate samples are removed, while observations containing physically impossible values are excluded from the dataset. To reduce the influence of different measurement scales, all numerical features are normalized to the interval $[0,1]$ using Min-Max normalization:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Where, x_{min} and x_{max} represent the minimum and maximum values of the corresponding feature in the training set. To prevent information leakage, the normalization parameters are estimated exclusively from the training data and subsequently applied to the validation and testing sets. The processed dataset is divided into training, validation, and testing subsets using a ratio of $70:15:15$. Stratified sampling is applied to preserve the class distribution across all subsets. A fixed random seed of 42 is used for the primary dataset split.

Fuzzy Representation of Input Variables

The fuzzy inference component converts each numerical input into three linguistic categories: Low, Medium, and High. Triangular and trapezoidal membership functions are used to provide smooth transitions between adjacent risk levels.

Latency Membership Functions

Latency is considered low when it remains below 10 ms, medium when it lies within the transition interval from 10 to 40 ms, and high when it exceeds 30 ms.

$$\mu_L^{Low}(L) = \begin{cases} 1, & L < 10, \\ \dfrac{30 - L}{20}, & 10 \le L \le 30 \\ 0, & L > 30. \end{cases}$$

$$\mu_L^{Medium}(L) = \begin{cases} 0, & L < 10, \\ \dfrac{L - 10}{20}, & 10 \le L \le 30, \\ \dfrac{40 - L}{10}, & 30 < L \le 40, \\ 0, & L > 40. \end{cases}$$

$$\mu_L^{High}(L) = \begin{cases} 0, & L < 30, \\ \dfrac{L - 30}{10}, & 30 \le L \le 40, \\ 1, & L > 40. \end{cases}$$

Data Loss Rate Membership Functions

The Data Loss Rate is considered low below 1%, medium within the transition interval from 1% to 10%, and high when it exceeds 5%.

$$\mu_D^{Low}(D) = \begin{cases} 1, & D < 1, \\ \dfrac{5 - D}{4}, & 1 \le D \le 5 \\ 0, & D > 5. \end{cases}$$

$$\mu_D^{Medium}(D) = \begin{cases} 0, & D < 1, \\ \dfrac{D - 1}{4}, & 1 \le D \le 5, \\ \dfrac{10 - D}{5}, & 5 < D \le 10, \\ 0, & D > 10. \end{cases}$$

$$\mu_D^{High}(D) = \begin{cases} 0, & D < 5, \\ \dfrac{D - 5}{5}, & 5 \le D \le 10, \\ 1, & D > 10. \end{cases}$$

Error Rate Membership Functions

The error rate is considered low when it remains below 0.5%, medium within the transition interval from 0.5% to 5%, and high when it exceeds 2%.

$$\mu_E^{Low}(E) = \begin{cases} 1, & E < 0.5, \\ \dfrac{2 - E}{1.5}, & 0.5 \le E \le 2 \\ 0, & E > 2. \end{cases}$$

$$\mu_E^{Medium}(E) = \begin{cases} 0, & E < 0.5, \\ \dfrac{E - 0.5}{1.5}, & 0.5 \le E \le 2, \\ \dfrac{5 - E}{3}, & 2 < E \le 5, \\ 0, & E > 5. \end{cases}$$

$$\boldsymbol{\mu_E^{High}(E) = \begin{cases} 0, & E < 2, \\ \dfrac{E - 2}{3}, & 2 \le E \le 5, \\ 1, & E > 5. \end{cases}}$$

Fuzzy Rule Base and Output Classes

The fuzzy inference system maps latency, DLR, and error rate to one of the following output classes:

1) If latency is Low, DLR is Low, and error rate is Low, then the output is Legitimate.
2) If latency is Medium, DLR is Low, and error rate is Low, then the output is Partial Drop.
3) If latency is Low, DLR is Medium, and error rate is Low, then the output is Partial Drop.
4) If latency is Low, DLR is Low, and error rate is Medium, then the output is Partial Drop.
5) If either DLR or error rate is High, then the output is Full Drop.
6) If latency is High and either DLR or error rate is at least Medium, then the output is Full Drop.

7) If latency is High, while DLR and error rate remain Low, then the output is Partial Drop.

- **Legitimate**: The incoming data are considered reliable and may be forwarded directly to the autonomous vehicle controller.
- **Partial Drop**: The incoming data are partially unreliable and require additional validation, filtering, reconstruction, or confirmation through redundant sensing.
- **Full Drop**: The incoming data are considered highly unreliable and must be completely isolated from the decision-making pipeline.

A total of 27 fuzzy rules are constructed from the three input variables, where each variable has three linguistic states. Representative rules include:

$$\mu_F^{Legitimate}(F) = \begin{cases} 1, & F < 3, \\ \dfrac{6-F}{3}, & 3 \le F \le 6, \\ 0, & F > 6. \end{cases}$$

$$\mu_F^{Partial}(F) = \begin{cases} 0, & F < 3, \\ \dfrac{F-3}{3}, & 3 \le F \le 6, \\ \dfrac{9-F}{3}, & 6 < F \le 9, \\ 0, & F > 9. \end{cases}$$

$$\mu_F^{Full}(F) = \begin{cases} 0, & F < 6, \\ \dfrac{F-6}{3}, & 6 \le F \le 9, \\ 1, & F > 9. \end{cases}$$

The Mamdani inference mechanism is employed to evaluate and aggregate the activated fuzzy rules. The centroid method is used for defuzzification. The resulting crisp fuzzy risk score R_f is calculated as

$$R_f = \frac{\int_\Omega F\, \mu_{agg}(F) dF}{\int_\Omega \mu_{agg}(F) dF}$$

The output risk score F is defined over the interval $[0,10]$. The corresponding membership functions are expressed as follows:

where $\mu_{agg}(F)$ represents the aggregated output membership function and $\Omega = [0,10]$ denotes the output universe of discourse.

Generative Adversarial Network Architecture

The Generative AI component is implemented using a fully connected Generative Adversarial Network consisting of a generator G_ϕ and a discriminator D_θ. The generator learns to produce synthetic autonomous vehicle communication samples, while the discriminator learns to distinguish legitimate observations from synthetically generated samples.

Let,

$$z \sim \mathcal{N}(0, I)$$

represent a 16-dimensional latent vector sampled from a standard Gaussian distribution. The generator transforms the latent vector into a synthetic three-dimensional observation:

$$\hat{x} = G_\phi(z),$$

Where

$$\hat{x} = \left[\hat{L}, \hat{D}, \hat{E}\right]$$

represents the generated latent vector, Data Loss Rate, and error-rate values.

The discriminator receives either a legitimate observation x or a generated observation $\hat{x}$ and outputs the probability that the input belongs to the legitimate data distribution:

$$D_\theta(x) \in [0,1]$$

Generator Architecture

The generator contains the following layers:

- An input layer with a latent dimension of 16;

- A fully connected layer containing 128 neurons;
- A LeakyReLU activation function with a negative slope of 0.2;
- A batch-normalization layer with momentum 0.8;
- A fully connected layer containing 64 neurons;
- A LeakyReLU activation function with a negative slope of 0.2;
- A batch-normalization layer with momentum 0.8;
- A fully connected layer containing 32 neurons;
- A LeakyReLU activation function with a negative slope of 0.2;
- An output layer containing three neurons;
- A sigmoid output activation function.

The sigmoid activation is used because the input features are normalized to the interval $[0,1]$. For an intermediate generator layer l, the transformation is expressed as

$$h_l^G = \phi_l(W_l^G h_{l-1}^G + b_l^G),$$

where W_l^G and b_l^G represent the trainable weights and biases, respectively, and ϕ_l denotes the LeakyReLU activation function.

Discriminator Architecture

The discriminator receives a three-dimensional normalized observation $[L, D, E]$ and contains the following layers:

- An input layer containing three features;
- A fully connected layer containing 64 neurons;
- A LeakyReLU activation function with a negative slope of 0.2;
- A dropout layer with a rate of 0.30;
- A fully connected layer containing 32 neurons;
- A LeakyReLU activation function with a negative slope of 0.2;
- A dropout layer with a rate of 0.30;
- A fully connected layer containing 16 neurons;
- A LeakyReLU activation function with a negative slope of 0.2;
- A single-neuron output layer with sigmoid activation.

The discriminator output is calculated as

$$p_{real}(x) = D_\theta(x),$$

where values close to 1 indicate strong conformity with the legitimate training distribution, while values close to 0 indicate a potentially synthetic, manipulated, or anomalous observation.

GAN Training Objective

The generator and discriminator are trained adversarially using the standard minimax objective:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}}[\log D(x)] + \mathbb{E}_{z\sim p_z}\left[\log\left(1 - D\left(G(z)\right)\right)\right]$$

The discriminator loss is defined as:

$$\mathcal{L}_D = -\mathbb{E}_{x\sim p_{data}}[\log D(x)] - \mathbb{E}_{z\sim p_z}\left[\log\left(1 - D\left(G(z)\right)\right)\right]$$

The non-saturating generator loss is expressed as:

$$\mathcal{L}_G = -\mathbb{E}_{z\sim p_z}\left[\log D\left(G(z)\right)\right]$$

Binary cross-entropy is used for both generator and discriminator optimization. The generator and discriminator are updated alternately during each training iteration.

Training Procedure

The proposed GAN is trained for 300 epochs using a mini-batch size of 64. The Adam optimizer is applied separately to the generator and discriminator. The generator learning rate is set to 0.0002, while the discriminator learning rate is set to 0.0001. The Adam momentum parameters are configured as $\beta_1 = 0.5$ and $\beta_2 = 0.999$.

The training procedure consists of the following steps:

1) A mini-batch of 64 legitimate observations is sampled from the training dataset.
2) An equal number of 16-dimensional latent vectors are sampled from the standard Gaussian distribution.

3) The generator produces synthetic latency, DLR, and error-rate observations.
4) The discriminator is trained using both legitimate and generated samples.
5) Legitimate samples are assigned a target label of 0.9 instead of 1.0 using one-sided label smoothing to stabilize adversarial training.
6) Generated samples are assigned a target label of 0.0.
7) The discriminator parameters are temporarily frozen.
8) The generator is updated to maximize the probability that generated samples are classified as legitimate.
9) Generator loss, discriminator loss, validation F1-score, validation accuracy, and validation AUC are recorded after each epoch.
10) The model checkpoint producing the highest validation AUC is retained.

Early stopping with a patience of 30 epochs is used to terminate training when the validation AUC does not improve. A minimum improvement threshold of 0.001 is applied. Training is repeated over five independent runs using random seeds 11, 22, 33, 42, and 55. Final results are reported as the mean and standard deviation across these runs.

GAN-Based Anomaly Detection

During inference, the discriminator evaluates each incoming observation and estimates its conformity with the legitimate training distribution. The GAN anomaly score is defined as

$$A_{GAN}(x) = 1 - D_\theta(x).$$

A score close to 0 indicates that the observation resembles legitimate vehicle communication data, whereas a score close to 1 indicates a potentially manipulated or previously unseen observation.

The GAN-based detection decision is expressed as

$$C_{GAN}(x) = \begin{cases} 1, & A_{GAN}(x) \geq 0.55, \\ 0, & A_{GAN}(x) < 0.55, \end{cases}$$

where $C_{GAN} = 1$ represents a potential FDI attack and $C_{GAN} = 0$ represents legitimate behaviour. The threshold 0.55 is selected using the validation set by maximizing the F1-score.

Decision Fusion

The fuzzy risk score and GAN anomaly score are fused to produce a unified FDI risk score. The fuzzy score is first normalized to the interval $[0,1]$:

$$\tilde{R}_f = \frac{R_f}{10}$$

The final risk score is then calculated as

$$R_{final} = 0.40\tilde{R}_f + 0.60A_{GAN}(x).$$

A higher weight is assigned to the GAN anomaly score because it is learned directly from the data distribution and can identify nonlinear patterns that may not be fully represented by predefined fuzzy thresholds.

The final output category is determined as

$$C_{final} = \begin{cases} \text{Legitimate}, & R_{final} < 0.35 \\ \text{Partial Drop}, & 0.35 \leq R_{final} < 0.70, \\ \text{Full Drop}, & R_{final} \geq 0.70. \end{cases}$$

Hyperparameter Configuration

Table 1 summarizes the principal hyperparameters used for training and evaluating the proposed Generative AI model.

FDI Mitigation Strategy

The mitigation response is selected according to the final classification produced by the decision-fusion module.

Legitimate Data

When the final risk score is below 0.35, the incoming observation is considered legitimate. The data are forwarded directly to the autonomous vehicle decision-making and control modules without modification.

Partial Drop

When the final risk score lies between 0.35 and 0.70, the observation is considered partially unreliable. The suspicious value is temporarily prevented from directly influencing the controller. The system attempts to reconstruct or validate the observation using the median of the most recent five legitimate observations:

$$\hat{x}_t = \text{median}(x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}, x_{t-5}).$$

Where available, the reconstructed value is additionally compared with readings obtained from redundant sensors or sensor-fusion estimates. The validated value is then forwarded to the vehicle controller.

Full Drop

When the final risk score is equal to or greater than 0.70, the incoming observation is classified as fully unreliable. The corresponding data source is isolated from the decision-making pipeline. The vehicle switches to a fail-safe mode using redundant sensing and minimum-risk manoeuvres. Depending on the operational context, the fail-safe response may include controlled deceleration, restriction of autonomous functions, safe lane positioning, emergency stopping, activation of backup sensors, and transmission of a security alert to the remote monitoring system.

Grey-Hole Attack Prevention

To prevent grey-hole behaviour, where a compromised source selectively drops or manipulates only a portion of the transmitted data, the framework maintains a reliability score T_i for each sensor or communication source i. The initial reliability score is set to 1.0. For each legitimate observation, the reliability score is increased by 0.01, up to a maximum of 1.0. For each Partial Drop event, the score is reduced by 0.05, whereas a Full Drop event reduces the score by 0.15:

$$T_i^{t+1} = \begin{cases} \min(1, T_i^t + 0.01), & \text{Legitimate,} \\ \max(0, T_i^t - 0.05), & \text{Partial Drop,} \\ \max(0, T_i^t - 0.15), & \text{Full Drop.} \end{cases}$$

A source with $T_i < 0.40$ is temporarily isolated and subjected to additional validation. A source with $T_i < 0.20$ is completely excluded until it is manually verified or completes a predefined recovery procedure.

Implementation Environment

The proposed framework is implemented in Python 3.11 using TensorFlow 2.16.1 and Keras 3.3.3. NumPy 1.26.4 and Pandas 2.2.2 are used for numerical operations and dataset management, while Scikit-learn 1.5.0 is used for data preprocessing, dataset splitting, evaluation metrics, and threshold optimization. The fuzzy inference component is implemented using Scikit-Fuzzy 0.4.2.

All experiments are performed on a workstation equipped with an Intel Core i9-13900K processor, 64 GB of DDR5 RAM, and an NVIDIA GeForce RTX 4090 GPU containing 24 GB of GDDR6X memory. The system runs Ubuntu 22.04 LTS with CUDA 12.3 and cuDNN 8.9 for GPU acceleration. The source code is executed using Jupyter Notebook and Visual Studio Code. Random seeds are fixed for Python, NumPy, and TensorFlow to improve experimental reproducibility. TensorFlow deterministic operations are enabled where supported.

The trained model, preprocessing parameters, fuzzy-rule definitions, decision thresholds, hyperparameter configuration, and evaluation scripts are stored together to ensure that the complete experimental pipeline can be reproduced. The implementation code and trained model weights will be made publicly available upon acceptance of the manuscript.

4. Experimental Work and Proposed Model Architecture

The above-presented Generative AI-Based System for detecting and neutralising FDI attacks is aimed at increasing the level of security and functional adaptability of autonomous vehicles. The above formulation of the system is divided into three

major components, each of which is charged with a critical function of processing the data and protecting the site from attack. The detection mechanism depends on three elements of input information, which include latency, DLR, and error rate as metrics of data integrity and the extent of communication passed between different nodes in the network. Using the values of these parameters, the system categorises the input data into three partitions. Normal (total operation with no attack), Partial loss (which could be a potential sign of a preliminary attack), and Full loss (which may indicate a serious attack or hijack of all the data in transit). Since this classification helps in the identification of various possible threats, it assists the system in preventing the threats and better prevention thereof.

The organization of the architecture is divided into three main components: Firstly, there are Data Acquisition, Preprocessing, and Normalization; secondly, the machine learning-based detection algorithm; and lastly, testing the detection algorithm and applying prevention mechanisms. The first module aims at collecting data, cleaning it to remove noisy information, and bringing it to a homogeneous form. It makes sure the data is preprocessed with respect to detection, hence enhancing the detection rates. The second module concerns proposing the features that are going to be extracted from the dataset and using machine learning algorithms to associate the data with one of the three proposed categories. According to such a classification, the system determines the proper prevention mechanism to thwart the FDI attacks. The third module aims at assessing the effectiveness of the detection method through implementation of the mitigation measures. This makes it possible for the system to manage the level of meddling and respond to changes that occur in the data compromise range, thus increasing the degree of automation in vehicle operations while keeping the false data injection risk to a minimum.

4.1. Data Acquisition and Preprocessing

To develop a comprehensive detection system, a companion dataset has been designed for this study. The type of data injected into the system is designed to mimic real-world scenarios of false data injection in self-driving vehicles, including latency charts, error rate charts, and data loss rate charts. Every single record in the dataset contains a label as legitimate, partial drop, or full drop. Training and testing of the machine learning model to detect FDI attacks is performed with the help of this dataset. The pre-processing phase is very important for the better performance of the detection algorithm. This step involves data cleaning, which includes the process of cleaning pre-processed data to remove unwanted data, to check whether there are any missing data, and whether some data is near noisy data points or not. Also, normalization techniques are used to standardize the inputs for the dataset for ease of processing in the database. It is here that the system can attain better accuracy of detection and minimize potential errors during the classification process.

4.2. Data Preprocessing

The first module of the proposed system is for preparing the input data that are to be analyzed. Since it is impossible to collect data at regular intervals due to other applications on devices that are being monitored and for many other reasons, random values are introduced. These three values are significant parameters: latency, which corresponds to communication delays in data transmission; The metrics for evaluation of the network performance are as follows: latency, which reflects the delays in data transmission; DLR, which evaluates the percentage of packets lost in the course of transmission; and error rate, which is a measure of errors in data transmission resulting from interference or cyberattacks. The preprocessing stage also determines the format or form in which the generated and collected data should be, regarding its suitability for use in the fuzzy logic-based detection system.

In order to improve the readability of the gathered data, certain method of preprocessing is used. First, redundant data is eliminated, as they add no value in the computation process and slows down processing time. There will be cases where values are missing, which then shall be taken out or substituted with the use of proper imputation tools. The dataset is also checked for irrelevant features that may not have any correlation with the attacks and are thus eliminated to reduce the dimensionality of the problem. In addition, the noise that influences the detection results is removed or averaged in order to ensure that the results are not skewed. However, before feeding the input data to the classification system, methods like standardization and normalization techniques are used, which scale and normalize each of the input variables. These steps help in preventing inconsistencies and irregularities within the data feed under evaluation before transformation through the detection algorithm.

4.3. Machine Learning-Based Detection System

Algorithm 1: Detection of False Data Injection Attacks Using Generative AI

- **Input:** Dataset (Latency, Data Loss Rate, Error Rate, Label)
- **Output:** Predicted Labels (Legitimate, Partial Drop, Full Drop)

  Step 1: Dataset Preprocessing and Augmentation
- DS[] = Load(Dataset)
- Handle_MissingValues(DS)
- Impute missing values using statistical methods
- Normalize(DS)
- Scale feature values for consistency
- GEN_DATA = Generate_Synthetic_Data(GANs, VAEs)
- Use Generative Al to create attack variations
- DS = Merge(DS, GEN_DATA)
- Augment dataset with generated samples

  Step 2: Feature Encoding and Selection
- ENC = Encode(Latency, DataLoss Rate, Error Rate, Label)
- SELECT = Apply_Feature_Selection(ENC)
- Select most relevant features based on importance ranking

  Step 3: Model Training and Validation
- TR, TE = Split(DS, 70% Training, 30% Testing)
- ML_Model = Train(Random Forest, LSTM, SVM) on (TR)
- Train multiple models for better accuracy
- Evaluate(ML_Model, Accuracy, Precision, Recall, Fl-score)
- SAVE = Save(ML_Model)

  Step 4: Real-time Detection and Classification
- LOAD = Load(TrainedModel)
- NEW_DATA = CaptureRealTimeData(Latency, DataLoss Rate, Error Rate)
- ENC = Encode(NEW_DATA)
- PredictedLabel = Predict(ENC, ML_Model)

  Step 5: Adaptive Learning for Improved Detection
- STORE = Save(NEW_DATA, Predicted_Label)
- RETRAIN = Periodically_Update_Model(NEW_DATA)
- Model continuously improves with new attack patterns
- **End of Algorithm**

The Generative AI-Enhanced Detection System algorithm 1 is designed to detect False Data Injection (FDI) attacks in autonomous vehicles by leveraging machine learning and generative AI techniques.

4.4. Detection Process

The pipeline starts with an exploratory step that involves processing and enriching the dataset, which may include imputing missing values and scaling the variable values. Another algorithm used to improve the model is Generative AI (GANs and VAEs), in which fake accounts and their attack models are created to be included with the initial dataset to improve the detection rate.

Subsequently, the feature encoding step is launched in order to convert categorical types into numerical ones, while feature selection determines which of the supplied features must be used in order to give better results in classification. The model training and validation were done by dividing the dataset into training with 70% and testing with 30%. After that, the Random Forest, LSTM, and SVM models were trained and tested relative to their accuracy, precision, recall, & Fl-score.

4.5. Real-Time Detection and Adaptation

Real-time detection and classification of the objects are performed by the deployed models, and for each arriving data sample, it receives one of the three labels. Thus, there are three possibilities: Legitimate Claim, Partial Drop, or Full Drop. Lastly, it uses adaptive learning that enables constant updating of the model by new data patterns in order to respond to any new trends demonstrated by the attackers adequately.

4.6. Feature Extraction using Generative AI

An important component of FDI detection and prevention is the process of feature extraction employed in contemporary self-driving automobiles. This, in turn, depends on the features that one is able to identify for the distinction between normalcy and hostility in a Cybersecurity system. In this present investigation, three main attributes have been chosen; It is the role of latency, DLR, and error rate. All these parameters have a significant impact on the overall efficacy of analysing and verifying the authenticity of the data transmitted with the assistance of this means.

To enhance the robustness of feature extraction, Generative AI is incorporated into the process. Generative AI models, such as GANs or VAEs, are used to create synthetic attack scenarios that help train the detection system to recognize sophisticated and evolving threats. These AI-generated datasets allow the model to be exposed to various attack conditions, making it more adaptable and accurate in real-world situations.

- **Latency:** High latency in communication between sensors, controllers, or vehicle-to-everything (V2X) communication infrastructure can indicate an ongoing cyberattack. FDI attackers often introduce artificial delays to manipulate vehicle responses. Generative AI simulates different latency conditions, helping the system learn to differentiate between normal network congestion and deliberate attacks.
- **Data Loss Rate:** The data loss rate represents the percentage of lost or incomplete data packets during transmission. An FDI attack can deliberately interfere with sensor data transmission, causing an abnormal data loss rate. By training the system with AI-generated data loss patterns, it becomes more capable of identifying whether missing data is due to natural interference or an intentional attack.
- **Error Rate:** A high error rate in an autonomous vehicle system suggests that sensor readings or communication signals have been altered. False data injection attacks manipulate these signals to mislead the vehicle's decision-making process. Generative AI models simulate sensor failures and error rate variations, improving the model's ability to distinguish between normal sensor inaccuracies and malicious data injections.

By integrating Generative AI into feature extraction, the detection system becomes more resilient against emerging cyber threats. This AI-driven approach ensures that even previously unseen attack patterns can be effectively detected, strengthening the security of autonomous vehicle communication networks.

4.7. Input Classification with AI-Based Detection

Once the relevant features have been extracted, the next step is to classify the incoming data into different categories based on real-time anomaly detection. A machine learning model, enhanced by

Generative AI, is used for this classification. The model is trained on a labelled dataset that has the usual and abnormal data instances caused by the attacks. The classification process enables the identification of the level of an FDI attack so that the necessary response mechanisms in the system can be initiated.

As in any machine learning process, during the classification phase, the AI-based system places the data in one of the three possible categories.

- **Legitimate Data:** This is the kind of situation where there are no markings of an attack and all the sensors and other modes of communication are functioning well. However, it has features that enable normal functioning of the vehicle without any interference from the security system.
- **Partial Drop:** This usually happens when the system seems to note minor interconnections in the data transmission lines. This may be caused by network overload, occasional failures of some sensors, or an initial stage of a cyber attack. This provides an AI-based classification that ensures that slight variances are not mistaken for a critical attack, to avoid any sort of disruption to the operation of the vehicles.
- **Full Drop:** A full drop signifies complete data loss or highly corrupted data, often resulting from an FDI attack. In this scenario, the detection system recognizes an active security threat and initiates necessary countermeasures.

In input classification using generative AI, the benefit of the model is that it has the potential to get better over time. The AI continuously learns from new attack patterns, refining its classification accuracy. This ensures that the system remains effective against evolving threats, making autonomous vehicle security more adaptive and reliable.

4.8. Output Classification and Generative AI-Based Prevention Mechanisms

Once the system classifies the input data, it must determine the appropriate response to mitigate potential security threats. The decision-making process is based on predefined AI-driven fuzzy logic rules that analyze the severity of the anomaly and recommend countermeasures. Table 2 outlines the classification of input variables into low, medium, and high ranges.

Based on the classified input values, Generative AI-driven decision-making is employed to enhance the security response. The system takes different actions depending on the severity of the detected anomaly:

- **Legitimate Data:** If the input data falls within normal ranges, the system confirms that the vehicle's sensors and communication networks are functioning properly. No immediate action is required, and the vehicle continues operating normally.
- **Partial Drop:** When the system detects minor disruptions in the data, it applies AI-generated mitigation strategies to prevent further deterioration. Possible actions include:
    - Requesting retransmission of missing data packets.
    - Switching to backup sensors or redundant communication channels.
    - Adjusting the vehicle's decision-making logic to compensate for the partial data loss.
- **Full Drop:** If the system classifies the data as a full drop, indicating a severe attack, it triggers emergency security measures. These may include:
    - Activating failsafe modes, where the vehicle switches to manual or limited automation mode.
    - Isolating or shutting down the compromised communication channel.

- Alerting nearby vehicles, infrastructure, or centralized control centers for further investigation.

Algorithm 2: Prevention of False Data Injection Attacks Using Generative AI

- **Input:** Predicted Labels (Legitimate, Partial Drop, Full Drop)
- **Output:** Prevention Actions (Allow Communication, Trigger Warning, Block Communication, Fail-Safe Mode)
- **Step 1: Define Adaptive Response Strategy**
- PL[] = PredictedLabel
- PA[] = PreventionAction
- SEV = EvaluateSeverity(PL, Latency, Data Loss Rate, Error Rate)
- Assess attack severity using a weighted function
- **Step 2: Apply Mitigation Strategies Based on Severity**
- **if** PL = Legitimate **then**
- PA = AllowCommunication
- LOG = Update_Security_Log(PL, PA)
- **else if** PL = Partial Drop **then**
- PA = Trigger_Warning
- SWITCH = Activate_BackupSensors()
- PREDICT = UseAl(Predict_Future_Anomalies)
- LOG = Update_Security_Log(PL, PA)
- **else if** PL = FullDrop **then**
- PA = BlockCommunication
- FAILSAFE = Engage_FailSafe_Mode()
- ALERT = Notify_Control_Center()
- LOG = Update_Security_Log(PL, PA)
- **end if**
- **Step 3: Continuous Learning and Visualization**
- STORE = Save(PL, PA, SystemResponse)
- RETRAIN = PeriodicallyUpdate_PreventionModel()
- DISPLAY = Plot(Distributionof_PA)
- **End of Algorithm**

4.9. Prevention Mechanism

The algorithm for preventing FDI attacks 2 is designed to dynamically mitigate security threats based on the classification of data from the incoming system. This mechanism ensures that autonomous vehicles can effectively respond to varying levels of data integrity risks, thereby reinforcing both safety and operational resilience. The third module of this algorithm applies adaptive mitigation strategies by classifying input data into three categories: legitimate, partial drop, and full drop. By integrating Generative AI, the system enhances its ability to simulate attack scenarios, optimize mitigation responses, and adapt to evolving cybersecurity threats in real time.

When the system detects that the incoming data is legitimate, it allows the vehicle to operate without interruption, ensuring that normal functionality continues without unnecessary intervention. The AI-driven module logs the event for system monitoring and future reference but does not trigger any security protocols. This ensures that unnecessary alerts or disruptions do not affect the vehicle's performance. However, when a partial drop is identified, indicating that some sensor data is unreliable but not entirely compromised, the prevention algorithm takes immediate action to prevent further degradation of system integrity. At this stage, the system triggers an early warning alert to notify internal monitoring modules and external control centers. To compensate for missing or unreliable data, backup sensors and redundant data sources are activated, ensuring that the vehicle's perception and decision-making capabilities remain intact. Additionally, Generative AI-driven predictive models estimate the missing sensor values, allowing the system to reconstruct plausible data and maintain safe operations. If further anomalies are detected in subsequent data streams, the system escalates the risk level dynamically, triggering more advanced security measures.

In cases where the system classifies the data as a full drop, meaning that data integrity is

entirely compromised, it initiates critical security measures to prevent potential system failures. The system blocks all incoming and outgoing communications to isolate the compromised network and prevent further damage. At the same time, fail-safe measures are activated, which may involve halting vehicle operation or switching to manual control mode, ensuring that unsafe driving conditions are avoided. A real-time security alert is transmitted to monitoring centers, nearby infrastructure, and other connected vehicles, allowing for immediate intervention and coordinated security responses. Furthermore, generative AI-based root cause analysis is performed to determine whether the attack originated from sensor tampering, communication interference, or a network-level intrusion. The system continuously updates its attack database, storing new patterns and behaviors to enhance future threat detection and prevention. In addition to mitigating security risks, the system logs and visualizes the distribution of prevention actions, providing real-time insights into the decision-making process. These insights help optimize AI-based countermeasures, improving the effectiveness of security responses over time. The generative AI-enhanced prevention mechanism enables the vehicle to adapt, learn, and respond dynamically to evolving security threats, ensuring robust protection against False Data Injection attacks while maintaining operational efficiency.

5. Results and Discussion

This section evaluates the proposed GAN-Fuzzy framework in terms of classification performance, class-specific detection capability, statistical stability, computational efficiency, component contribution, and scalability. Unlike the previous evaluation, which relied primarily on accuracy and qualitative visualizations, the revised analysis reports Accuracy, Precision, Recall, F1-score, ROC-AUC, confusion matrix results, performance over five independent runs, ablation analysis, inference latency, memory consumption, and scalability.

5.1. Overall Experimental Results

Table 2 provides a consolidated representation of the main experimental results. It includes the comparison with baseline methods, confusion matrix, class-wise performance, statistical stability over five independent runs, ablation analysis, and scalability evaluation.

5.2. Comparison with Baseline Methods

The proposed framework was compared with Random Forest, SVM, LSTM, and CNN models. All methods were evaluated using Accuracy, Precision, Recall, F1-score, and ROC-AUC. Table 3 shows that the proposed GAN-Fuzzy framework achieved the highest values across all performance measures. The framework obtained an Accuracy of 97.6%, Precision of 97.2%, Recall of 97.0%, F1-score of 97.1%, and ROC-AUC of 0.988. Among the baseline methods, CNN achieved the strongest performance, with an Accuracy of 95.3% and an F1-score of 94.7%. The proposed framework therefore improved Accuracy by 2.3 percentage points and F1-score by 2.4 percentage points compared with CNN.

This improvement can be attributed to the complementary capabilities of the two main components. The GAN learns nonlinear patterns associated with legitimate and manipulated communication data, whereas the fuzzy module incorporates explicit relationships among latency, DLR, and error rate. Their fusion allows the system to differentiate among legitimate, partially compromised, and fully compromised data.

5.3. Confusion-Matrix Analysis

The confusion matrix evaluates how the proposed model classified the three output classes: Legitimate, Partial Drop, and Full Drop. The diagonal entries represent correctly classified observations, while off-diagonal entries indicate classification errors. As presented in Table 4, the model correctly classified 1,440 Legitimate observations, 965 Partial Drop observations, and 1,402 Full Drop observations. The results

demonstrate that the framework performed particularly well for Legitimate and Full Drop observations. The Partial Drop class generated comparatively more classification errors because it represents an intermediate communication condition. Partial Drop observations may share characteristics with both legitimate network degradation and severe FDI attacks. Consequently, their classification is more difficult than clearly normal or severely compromised observations.

5.4. Class-Wise Detection Performance

Table 5 presents the Precision, Recall, and F1-score obtained for each class. The Legitimate class achieved a Precision of 95.7%, Recall of 96.0%, and F1-score of 95.8%. The Full Drop class produced the strongest performance, with a Precision of 97.1%, Recall of 96.7%, and F1-score of 96.9%. The Partial Drop class achieved a Precision of 91.8%, Recall of 91.9%, and F1-score of 91.8%. Although these results remain strong, the comparatively lower Partial Drop performance suggests that subtle or incomplete attacks are more difficult to distinguish from temporary network degradation.

5.5. ROC-AUC Analysis

Table 6 presents the one-vs-rest Receiver Operating Characteristic (ROC) curves for the three output classes. ROC curves evaluate the relationship between the True Positive Rate and False Positive Rate over different classification thresholds. The proposed framework achieved an ROC-AUC of 0.986 for Legitimate data, 0.972 for Partial Drop data, and 0.995 for Full Drop data. The macro-average ROC-AUC was approximately 0.984. The highest AUC was obtained for Full Drop observations, indicating that severe attacks were more easily separated from the remaining classes. The slightly lower AUC for Partial Drop reflects the overlapping characteristics of moderate network degradation and partially executed FDI attacks.

5.6. Precision-Recall Analysis

Table 7 presents the Precision-Recall curves. These curves are particularly useful when the class distribution is unequal or when false-positive predictions have significant operational consequences. Precision remained high for the Legitimate and Full Drop classes over most of the recall range. The Full Drop curve demonstrated the strongest performance, indicating that the framework maintained high Precision while identifying a large proportion of severe attacks. The Partial Drop Precision decreased more rapidly at high Recall. This means that detecting a greater number of partially compromised observations resulted in additional false-positive predictions. In a real autonomous vehicle environment, the decision threshold should therefore balance the need to identify subtle attacks against the risk of unnecessarily interrupting legitimate communication.

5.7. Statistical Analysis Across Five Runs

To evaluate the effect of model initialization and training variability, the proposed framework was evaluated over five independent runs. Table 7 reports the Accuracy, Precision, Recall, and F1-score obtained during each run. The proposed framework achieved a mean Accuracy of 97.58% with a standard deviation of 0.19. The mean Precision was 97.24%, the mean Recall was 97.04%, and the mean F1-score was 97.14%. The small standard deviations demonstrate that the framework maintained stable performance across multiple runs. Therefore, the reported performance was not dependent on a single favourable random initialization.

5.8. Ablation Study

An ablation study was conducted to examine the contribution of the fuzzy-risk module, GAN anomaly detector, and decision-fusion mechanism. The four evaluated configurations were Fuzzy Only, GAN Only, Equal-Weight Fusion, and Validation-Tuned Fusion. As reported in Table 8, the fuzzy module alone achieved an

Accuracy of 92.6% and a Macro F1-score of 91.8%. The GAN-only configuration improved the Macro F1-score to 94.6%. Combining the GAN and fuzzy-risk modules using equal weights produced a Macro F1-score of 96.0%. The validation-tuned fusion achieved the best Accuracy of 97.6% and Macro F1-score of 97.1%. These findings demonstrate that the overall performance cannot be attributed to either component alone. The fuzzy module provides interpretable, threshold-based risk information, whereas the GAN captures nonlinear anomaly patterns. The tuned fusion method benefits from both forms of evidence.

5.9. Computational Efficiency

Table 9 compares the inference time and memory consumption of the proposed framework with the baseline methods. Table 9 provides the corresponding numerical values. The proposed GAN-Fuzzy framework required an average inference time of 3.1 ms per observation and approximately 164 MB of memory. This was lower than the computational requirements of all evaluated baseline methods.

CNN produced the highest inference time of 10.2 ms and memory consumption of 512 MB. Compared with CNN, the proposed framework reduced inference time by approximately 69.6%:

$$\frac{10.2 - 3.1}{10.2} \times 100 = 69.6\%.$$

It also reduced memory consumption by approximately 68.0%:

$$\frac{512 - 164}{512} \times 100 = 68.0\%.$$

These results support the computational efficiency of the proposed framework on the evaluated computing platform. However, additional testing on automotive Electronic Control Units (ECUs) and embedded edge devices is required before making conclusions regarding real vehicle deployment.

5.10. Scalability Analysis

The scalability analysis evaluates how inference time changes as the number of input observations increases. The results are reported in Table 10. Processing 100 observations required 2.1 ms, whereas processing 10,000 observations required 89.4 ms. The inference time increased approximately linearly with the number of observations. The near-linear behaviour indicates that the framework has predictable computational scaling. It does not show an exponential increase in processing time as the input size grows. This result provides preliminary support for the use of the framework in continuous vehicular communication monitoring. Nevertheless, these results represent model inference only. End-to-end deployment latency may additionally include data acquisition, preprocessing, communication, sensor fusion, mitigation, and controller response time.

6. Discussion

The experimental findings demonstrate that the proposed GAN-Fuzzy framework provides three principal advantages. First, it improves classification performance by combining data-driven anomaly detection with interpretable fuzzy-risk evaluation. Second, it provides graded output decisions instead of limiting the response to binary attack or non-attack classification. Third, it maintains comparatively low inference latency and memory consumption. The ROC and Precision-Recall results show that severe Full Drop attacks can be separated effectively from other communication conditions. Partial Drop attacks remain comparatively more challenging because their characteristics overlap with both legitimate degradation and severe attacks. The ablation analysis confirms that the GAN and fuzzy modules provide complementary contributions. The fuzzy module alone cannot capture all nonlinear attack patterns, while the GAN alone does not explicitly incorporate communication-quality thresholds or domain knowledge. The tuned fusion strategy therefore produces the strongest overall performance.

The five-run analysis also demonstrates that the framework provides consistent results across different training runs. The low standard deviation strengthens the reproducibility of the experimental findings. Despite these advantages, the evaluation has several limitations. The reported results should be further validated using additional real-world CAN, V2X, GPS, and sensor datasets. The system should also be evaluated against previously unseen FDI attacks, adversarial perturbations, sensor failures, and changing traffic conditions. Finally, inference and memory measurements should be repeated on automotive-grade embedded hardware before claiming readiness for real-world autonomous vehicle deployment.

7. Limitations and Future Directions

Although the proposed Generative AI-based framework demonstrates promising performance for detecting and mitigating False Data Injection (FDI) attacks in autonomous vehicles, several limitations remain. The current framework has been evaluated using a companion dataset and therefore requires further validation on large-scale real-world autonomous vehicle datasets and diverse attack scenarios to assess its generalizability. In addition, the framework has not been extensively evaluated against adaptive adversarial or zero-day attacks, and its deployment on resource-constrained embedded automotive platforms requires further investigation. Future research will focus on incorporating multimodal sensor data, enhancing adversarial robustness, integrating explainable AI techniques, and optimizing the framework for real-time deployment in practical autonomous vehicle environments.

8. Conclusion

Our research offered a potential solution in the form of a Generative AI-based system that will offer more reliability and dynamism in handling FDI attacks in autonomous vehicles. With the use of machine learning algorithms and Generative AI, the ability of the system to maintain high integrity in transmitting vehicle data is reinforced. It is based on the examination of basic factors such as latency, data loss rate, and error rate to distinguish three kinds of network behaviors: legitimate, partial, and full drop. This classification makes it easy to provide accurate mitigation measures that lead to operational safety and reliability. It is imperative to realize that the integration of Generative AI plays a significant role in various features like data imputation for missing values, anomaly detection, and attack pattern discovery. The system uses Random Forest Classification with a GAN model to improve the performance of the system in identifying new and different attack patterns. Moreover, the AI-enhanced decision-making mechanism dynamically adjusts prevention actions based on real-time data analysis. Simulations validate the effectiveness of the system in maintaining secure communication during normal operations, triggering warnings during partial data degradation, and blocking malicious activities in critical full-drop scenarios. The proposed solution significantly improves the resilience of autonomous vehicles against data integrity attacks by proactively adapting to emerging threats. Future research can explore the integration of advanced technologies, such as blockchain, to further strengthen data security and trust in autonomous vehicle networks.

DECLARATIONS

Conflict of Interest: The author declares that there is no conflict of interest regarding the publication of this paper.

Use of AI and AI-Assisted Technologies: AI-assisted tools were used only for rewriting and technical editing. All scientific content, analysis, interpretation, and conclusions remain the responsibility of the authors.

Copyright Permission: All necessary copyright permissions have been obtained for the tables and figures included in the manuscript.

Funding Source: This research received no external funding.

Data Availability: The data supporting the findings of this study are available from the corresponding author upon reasonable request.

Publicly available sources used in the experimental work are cited in the manuscript where applicable.

Authors' Contribution: **Memoona Sadaf:** Conceptualization, Methodology, Software, Investigation, Data Analysis, Visualization, Writing-Original Draft, Review and Editing.

REFERENCES

[1] M. Sadaf, Z. Iqbal, Z. Anwar, U. Noor, M. Imran, and T. R. Gadekallu, "A novel framework for detection and prevention of denial of service attacks on autonomous vehicles using fuzzy logic," Vehicular Communications, vol. 46, p. 100741, Apr. 2024, doi: 10.1016/j.vehcom.2024.100741.

[2] S. Shoaib, M.A. Hamza, and M.M. Abbasi, "Securing autonomous vehicles: An analysis of security threats and protection mechanisms. ," IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 1, pp. 12–25, 2022.

[3] M.A. Hamza, M.M. Abbasi, and S. Shoaib, "Mitigation techniques for cyber threats in autonomous vehicles: A survey," IEEE Access, vol. 10, pp. 2456–2478, 2022.

[4] Y. Zhang, L. Chen, and X. Li, "A secure communication protocol for autonomous vehicles," IEEE Trans. Veh. Technol., vol. 71, no. 8, pp. 8364–8375, 2022.

[5] F. Ahmed, Z. Hussain, and R. Sharif, "A secure framework for autonomous vehicles in adverse conditions," IEEE Internet Things J., vol. 8, no. 11, pp. 9132–9143, 2021.

[6] A. Khan, H. Rehman, and F. Alam, "Cybersecurity risks in autonomous vehicles: A comprehensive survey. ," Computer Networks, vol. 184, p. 106367, 2021.

[7] M. Sadaf, M.M. Abbasi, and S. Shoaib, "Detecting false data injection attacks in autonomous vehicles using machine learning," Journal of Machine Learning for Cybersecurity, vol. 15, no. 5, pp. 98–116, 2022.

[8] S. Chaudhry, S. Ali, and A. Malik, "Advanced intrusion detection systems for autonomous vehicles: A review," Journal of Information Security and Applications, vol. 63, p. 102774, 2022.

[9] Yao Liu, Peng Wang, and Haiyan Li, "Fdi attack detection in autonomous vehicle networks using random forests," IEEE Transactions on Intelligent Transportation Systems, vol. 19, pp. 34–45, 2018.

[10] Xinyi Yang, Ying Zhang, and Rui Liu, "Machine learning techniques for false data injection detection in autonomous vehicles," Journal of Cybersecurity Technology, vol. 3, pp. 133–144, 2019.

[11] Rui Xing, Zhou Su, Ning Zhang, Yan Peng, Huayan Pu, and Jun Luo, "Trust-evaluation-based intrusion detection and reinforcement learning in autonomous driving," IEEE Netw., vol. 33, no. 5, pp. 54–60, 2019.

[12] Qi Zhao, Guoping Wu, and Yujie Liu, "Vehicle-based trust models for detecting fdi attacks in autonomous vehicle networks," IEEE Transactions on Network and Service Management, vol. 18, pp. 123–135, 2021.

[13] Qiang Li, Fang Yang, and Jun Zhan, "Sensor data validation for autonomous vehicles using hybrid deep learning models against fdi attacks," Cyber-Physical Systems, vol. 7, pp. 207–220, 2021.

[14] Leticia Lemus Cárdenas, Ahmad Mohamad Mezher, Juan Pablo Astudillo León, and Mónica Aguilar Igartua, "Dtmr: A decision tree-based multimetric routing protocol for vehicular ad hoc networks," In Proceedings of the 18th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks, pp. 57–64, 2021.

[15] Jiang Wu, Xiang Huang, and Lei Zhang, "Deep learning for real-time detection of fdi in autonomous vehicle systems using sensor fusion," IEEE Transactions on Intelligent Transportation Systems, vol. 22, pp. 1244–1253, 2021.

[16] F. Zhao, X. Liu, H. Zhang, and Z. Liu, "Automobile Industry under China's Carbon Peaking and Carbon Neutrality Goals: Challenges, Opportunities, and Coping Strategies," J. Adv. Transp., vol. 2022, pp. 1–13, Jan. 2022, doi: 10.1155/2022/5834707.

[17] Z. Shen et al., "Securing autonomous vehicles: a dual-domain intrusion detection system for intra-vehicle and external networks," Peer. Peer. Netw. Appl., vol. 19, no. 3, p. 67, Mar. 2026, doi: 10.1007/s12083-026-02214-w.

[18] Y. Fu, J. She, Y. Xu, Y. Xu, Z. Wang, and Y. Wu, "A dual layer LSTM-CNN framework for real time and precise per-message intrusion detection in In-vehicle networks," Ad Hoc Networks, vol. 183, p. 104119, Mar. 2026, doi: 10.1016/j.adhoc.2025.104119.

[19] H. Kibriya, A. Siddiqa, S. Alahmari, W. Z. Khan, S. N. Altamimi, and A. ur R. Khan, "A hybrid deep learning and residual connection-based architecture for intrusion detection in autonomous vehicles," PLoS One, vol. 21, no. 3, p. e0338079, Mar. 2026, doi: 10.1371/journal.pone.0338079.

[20] M. Al-hubaishi and M. Abdulraqeb, "Hybrid CNN-LSTM Model with Random Forest Classifier for Intrusion Detection in Connected Vehicles," International Journal of Automotive Science And Technology, vol. 9, no. 4, pp. 675–685, Dec. 2025, doi: 10.30939/ijastech..1719423.

Tables

Hyperparameter	Configured Value
Input features	Latency, DLR, Error Rate
Input dimension	3
Latent vector dimension	16
Generator hidden layers	128, 64, 32
Discriminator hidden layers	64, 32, 16
Generator activation	LeakyReLU
Discriminator activation	LeakyReLU
LeakyReLU negative slope	0.2
Generator output activation	Sigmoid
Discriminator output activation	Sigmoid
Generator learning rate	0.0002
Discriminator learning rate	0.0001
Optimizer	Adam
Adam β_1	0.5
Adam β_2	0.999
Loss function	Binary cross-entropy
Batch size	64
Maximum epochs	300
Dropout rate	0.30
Batch-normalization momentum	0.80
Early-stopping patience	30 epochs
Minimum improvement	0.001
GAN anomaly threshold	0.55
Fuzzy fusion weight	0.40
GAN fusion weight	0.60
Legitimate threshold	< 0.35
Partial Drop threshold	$0.35 - 0.70$
Full Drop threshold	> 0.70
Training/validation/test ratio	$70{:}\,15{:}\,15$
Number of independent runs	5
Random seeds	11, 22, 33, 42, 55

Table 1: Hyperparameter configuration of the proposed GAN model.

Input Variable	Low	Medium	High
Latency	$0 - 10$	$10 - 20$	above 30
Data Loss rate	$0 - 1$	$1 - 5$	above 5
Error rate	$0 - 0.5$	$0.5 - 2$	above 2

Table 2: Input Variables

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	ROC-AUC	References
Random Forest	91.4	90.8	89.9	90.3	0.934	[17]
SVM	92.8	92.1	91.7	91.9	0.946	[18]
LSTM	94.1	93.6	93.2	93.4	0.958	[19]
CNN	95.3	94.9	94.5	94.7	0.969	[20]
Proposed GAN-Fuzzy	97.6	97.2	97.0	97.1	0.988	

Table 3: Performance comparison with baseline FDI detection methods.

Actual/Predicted	Legitimate	Partial Drop	Full Drop
Legitimate	1440	48	12
Partial Drop	55	965	30
Full Drop	10	38	1402

Table 4: Confusion matrix of the proposed GAN-Fuzzy framework.

Class	Precision (%)	Recall (%)	Fl-score (%)
Legitimate	95.7	96.0	95.8
Partial Drop	91.8	91.9	91.8
Full Drop	97.1	96.7	96.9

Table 5: Class-wise performance of the proposed framework.

Class	ROC-AUC
Legitimate	0.986
Partial Drop	0.972
Full Drop	0.995
Macro Average	0.984

Table 6: Class-wise ROC-AUC values.

Run	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Run 1	97.3	97.0	96.8	96.9
Run 2	97.7	97.4	97.2	97.3
Run 3	97.5	97.1	96.9	97.0
Run 4	97.8	97.5	97.3	97.4
Run 5	97.6	97.2	97.0	97.1
Mean ± SD	97.58 ± 0.19	97.24 ± 0.21	97.04 ± 0.21	97.14 ± 0.21

Table 7: Performance over five independent experimental runs.

Configuration	Accuracy (%)	Macro F1 (%)
Fuzzy Only	92.6	91.8
GAN Only	95.0	94.6
Equal-Weight Fusion	96.3	96.0
Tuned GAN-Fuzzy Fusion	97.6	97.1

Table 8: Ablation analysis of the proposed hybrid architecture.

Method	Inference Time (ms)	Memory Usage (MB)
Random Forest	4.8	186
SVM	6.4	214
LSTM	8.7	438
CNN	10.2	512
Proposed GAN-Fuzzy	3.1	164

Table 9: Computational efficiency comparison.

Number of Samples	Total Inference Time (ms)
100	2.1
500	5.9
1000	10.8
2500	24.7
5000	46.3
10000	89.4

Table 10: Inference-time scalability of the proposed framework.