MDPI

Article

Enhancing Automotive Security with a Hybrid Approach towards Universal Intrusion Detection System

Md Rezanur Islam [1], Mahdi Sahlabadi [2], Munkhdelgerekh Batzorig[2] and Kangbin Yim [2,*]

[1] Department of Software Convergence, Soonchunhyang University, Asan-si, Korea, arupreza@sch.ac.kr
[2] Department of Information Security Engineering, Soonchunhyang University, Asan-si, Korea
* Correspondence: yim@sch.ac.kr; Soonchunhyang University, Asan-si, Korea) +xx-xxxx-xxx-xxxx (F.L.)

Abstract

Security measures are essential in the automotive industry to detect intrusions in-vehicle networks. However, developing a one-size-fits-all Intrusion Detection System (IDS) is challenging because each vehicle has unique data profiles. This is due to the complex and dynamic nature of the data generated by vehicles regarding their model, driving style, test environment, and firmware update. To address this issue, a universal IDS has been developed that can be applied to all types of vehicles without the need for customization. Unlike conventional IDSs, the universal IDS can adapt to data distribution shifts caused by changes in driving style, vehicle platform, or firmware updates. In this study, a new hybrid approach has been developed, combining Pearson correlation with deep learning techniques. This approach has been tested using data obtained from four distinct mechanical and electronic vehicles, including Tesla, Sonata, and two Kia models. The data has been combined into two frequency datasets, and wavelet transformation has been employed to convert them into the frequency domain, enhancing generalizability. Additionally, a statistical method based on independent rule-based systems using Pearson correlation has been utilized to improve system performance. The system has been compared with eight different IDSs, three of which utilize the universal approach, while the remaining five are based on conventional techniques. The accuracy of each system has been evaluated through benchmarking, and the results demonstrate that the hybrid system effectively detects intrusions in various vehicle models.

Keywords: CAN; IVN; Wavelet; Pearson; IDS

1. Introduction

Artificial intelligence has brought about a revolution in the automotive industry, allowing for the development of fully automated vehicles [1]. However, as technology continues to advance, there is a growing concern regarding the security of In-Vehicle Networks (IVNs) [2,3]. Industry is obligated to recognize the risks associated with IVNs and take necessary precautions to ensure the safety of both drivers and passengers. Modern vehicles are equipped with integrated electronic systems and sensors to enrich the connectivity feature. This phenomenon extends the surface of the attack and brings attention to the Intrusion Detection System (IDS). However, because of the nature of the data generated by vehicles, it is difficult to build an IDS solution that is appropriate for every scenario [4] for two reasons: First, every vehicle has a unique network architecture that generates very varied data profiles [5] for a scenario; second, the data generations vary depending on the circumstances and driving styles [6]. It is challenging to establish an extensive dataset that

Received:
Revised:
Accepted:
Published:

covers a wide range of scenarios because of these factors. There are few available datasets that present very limited scenarios. The researchers primary motivation for avoiding cross-validation is their awareness of that. The majority of the AI algorithms appear to operate with high-accuracy matrixes because the dataset contains relatively little situational data, yet the model is not generalizable. It is critical to use an IDS that can handle these variations and provide accurate results.

IDSs primarily monitor Controller Area Network (CAN) data because it is the main communication protocol within IVN [7]. IDSs face challenges when deployed across different CAN datasets [8]. CAN networks vary in function and design to meet specific vehicle needs and cater to the complexity of electronic and autonomous vehicles. For instance, Kia has three distinct CAN channels with Electronic Control Unit (ECU) numbers of approximately 100, each responsible for different functions [9]. CAN networks handle crucial functions with B-CAN, C-CAN, and M-CAN to manage body, chassis, and multimedia systems, respectively. These networks vary in speed and priority to cater to specific vehicle needs. Although the Kia models share the same channel, their internal physical CAN ID and payload differ [4]. Similarly, the BMW K-CAN, V-CAN, and K2-CAN consist of different components according to the manufacturer's internal functional design [10]. A significant difference between mechanical and electronic vehicles is their IVN architectures. This is because manufacturers offer various features to attract customers, and autonomous vehicles require many sensors to ensure safety [11]. For example, Tesla's ECUs vary in number, with each version around four ECUs [12], and it has nine types of CAN channels [13]. CAN networks undoubtedly vary, as demonstrated by these industrial examples.

Despite the importance of improving vehicle safety, the deployment of conventional IDS has faced significant challenges due to variations in network architecture and data generation practices among manufacturers. To address this issue, several study efforts have been conducted, including federated learning [4] and transfer learning [14]. However, transfer learning does not always bring a positive impact on new tasks. When there is little in common between domains, knowledge transfer may fail, leading to negative transfer and reduced performance on the target task [15]. Furthermore, with its decentralized data sources, federated learning faces data privacy and security challenges and high communication bandwidth needs. For IDS, the main issue is communication overhead, affecting efficiency and scalability due to the high data transmission costs during training [16]. On the other hand, universal IDS can effectively detect attacks in different vehicles, regardless of their data profile or network architecture. By further standardizing universal IDS, we can ensure system interchangeability, enhance reliability and meet regulatory requirements [7]. Utilizing AI technology with wavelet feature extraction and Pearson correlation, universal IDS provides innovative vehicle capabilities and serves as a practical cybersecurity solution for the evolving automotive environment. It also improves vehicle-wide adaptability by streamlining IVN monitoring and addressing changes in data profiles and network architectures. This study demonstrates the flexibility of IDS by training two models on low-frequency infusion data from two vehicles and testing them on three different vehicles to showcase their ability to adapt to various data frequencies.

This study focuses on DoS, Fuzzing, and Replay attacks that target CAN protocol in IVNs. These attacks [17] present unique challenges to vehicular systems and represent the most significant cybersecurity threats to IVNs. The study conducts these attacks on high and low-frequency bases since anomaly injection in IVNs is frequency agnostic [17]. The frequency, nature, and structure of malicious messages significantly impact IVN anomaly detection. The study aims to understand cybersecurity threats in vehicular networks better and create more effective countermeasures against them.

Section 2 discusses various IDS, including universal IDS and conventional IDS approaches. It also explores the application of Wavelet and Pearson methods in security. **Section 3** outlines methodologies for feature extraction for deep learning purposes and explains the preprocessing of data. **Section 4** focuses on the implementation of IDS. **Section 5** evaluates the results obtained from the implemented methodologies. **Section 6** provides a comprehensive discussion of the findings and insights of the study. Finally, **Section 7** concludes the paper by summarizing the key outcomes and contributions of the study.

2. Related Works

The section on universal approaches examines the limitations of existing IDSs, highlighting their constraints. The feature selection section explores the adaptability challenges of conventional IDSs in practical situations. Finally, the data conversion section focuses on the concept of data generalization.

2.1. Universal Approached Intrusion Detection

To the best of our knowledge, only three studies [18–20] have been conducted on a universal approach. Firstly, Novikova et al. proposed an unsupervised anomaly detection approach for CAN bus, identifying consistent signals across nine vehicles grouped into 32 subgroups. However, practical implementation faces challenges due to the requirement for separate autoencoder models for each subgroup, especially in resource-constrained IVN [21,22]. Accuracy measurements like false positive rates were not provided to evaluate their IDS.

Secondly, Mehmet et al. introduced WINDS, a Wavelet-based Intrusion Detection System with a specific accuracy matrix, requiring lower computational resources compared to [18]. It aims to universally enhance vehicle security against specific CAN ID injections. WINDS is rule-based, employing data generation to extract features transformed into wavelet coefficients, demanding high computational power [23]. However, it may generate false alerts with attack frequency changes and lacks cross-validation [24].

Thirdly, Rezanur et al. utilized a combination of heatmaps from CAN ID sequences, time gaps, and hamming distances of CAN IDs to develop a universal IDS from real vehicles. They applied various CNN architectures as IDS to find the best model. However, a concern is that the study was conducted on only one vehicle with only high-frequency injection. Therefore, further research is needed to validate the ability of the IDS to be manufacturer-independent and demonstrate universal IDS capability.

In summary, Novikova et al. conducted a study on achieving universal IDS capability. They utilized diverse datasets from different vehicles, emphasizing the importance of multi-source data. In contrast, Mehmet et al. and Rezanur et al. employed Continuous Wavelet Transform (CWT) and heat-map techniques, highlighting the significance of feature selection and data conversion. These methodologies enhance data generalization and standardization, ultimately improving adaptability and facilitating analysis across various applications in the vehicle network system. More details on this topic can be found in Section 2.2.

2.2. Feature Selection for Intrusion Detection

This section reviews conventional IDSs methodology and input features (CAN ID sequence-based, payload sequence-based, full data utilization-based, voltage signal-based, and hybrid-based detection) from the universal IDS point of view. Numerous studies [2,25–28] have been done on IVN security, mainly on their applicability to data sources in vehicular environments.

Table 1. Comparative Analysis of Vehicle CAN Data Characteristics Across Different Platforms and Data Collection Years.

Vehicle / Dataset	Data Collection Year	Number of CAN IDs	Data Length (s)	Average Time Gap (s)
Kia (LISA)	2022	45	2085	0.4772
Kia (HCRL)	–	45	2085	0.4772
Sonata (HCRL)	–	27	1943	0.5132
Tesla (LISA)	2021	69	3126	0.3195
Tesla (LISA)	2022	121	1652	0.6095
Tesla (LISA)	2024	267	2871	0.3476
Tesla (LISA)	2025	156	2472	0.4032

Yu et al. introduce TCE-IDS, employing a time interval conditional entropy algorithm to analyze decimal message IDs and data blocks from the CAN network in real-time for cyberattack detection. However, it relies on predefined rules and lacks the generalizability of a universal IDS, necessitating frequent customization for vehicle or firmware updates.

Xun et al. propose an approach focusing on voltage signals from the CAN bus, using FeatureBagging combined with CNN for identifying physical layer attacks. While effective, it may be less proficient in detecting application layer attacks [29]. Implementing an IDS at the application layer offers context awareness and content inspection advantages. However, a universal IDS may require assistance with voltage features due to diverse ECU hardware configurations.

Mansourian et al. propose a payload IDS. They utilize LSTM and ConvLSTM prediction models with a Gaussian Naïve Bayes classifier, achieving high precision in distinguishing attack-free and attack data. Challenges include detecting fuzzy attacks, resource demands on limited ECUs, and intruders using packet injection to alter CAN ID while maintaining the payload.

Jedh et al. study CAN ID sequence-based IDS, utilizing Messages-Sequence Graphs and various techniques, including cosine similarity, Pearson correlation, threshold-based methods, LSTM-RNN, and Change Point Detection (CPD). While effective, the study's limited dataset raises concerns about generalizability to diverse vehicles and driving conditions. Additionally, attackers manipulating payload while maintaining CAN ID consistency pose challenges for the IDS.

Lo et al. present HyDL-IDS, a hybrid CNN-LSTM-based intrusion detection system for IVN. While effective, it has limitations, including potential bottlenecks in LSTM's information extraction and computational overhead, especially for ECUs with limited capacity. If the primary model fails, the entire system may fail. Another limitation is that the CAN ID and payload are standardized according to CAN DBC [17], requiring customization for each vehicle [30].

As a result, it is important to address the limitations of current IDS systems. These systems operate only with specific data sources from particular vehicles and lack the ability to generalize data.

2.3. Data Generalization for Intrusion Detection

Wavelet transforms enhance data generalization by capturing both time and frequency information [31], enabling a multi-resolution view that isolates key features while reducing noise [23,32]. This transform allows for efficient dimensionality reduction and temporal localization, which improves pattern recognition in non-stationary data like signals or time series. By compressing data and retaining only essential features, wavelets help machine learning models focus on relevant patterns, leading to improved generalization and reduced overfitting.

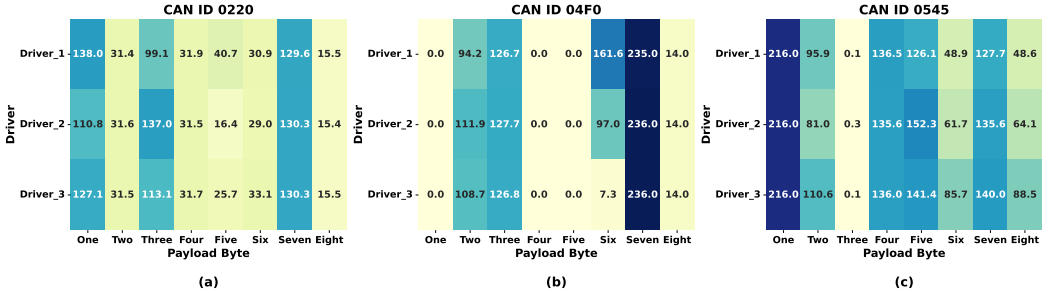

Figure 1. Comparative Analysis of Driver-Dependent CAN Payload Patterns for Multiple CAN IDs

CAN systems generate large amounts of data (shown in Table 1) that can be difficult to analyze due to its changing nature. Frequency domain analysis is a helpful approach to manage this complexity and gain a better understanding of the data. The Fourier transform is a commonly used technique in frequency domain analysis, but it has the drawback of compromising frequency and time resolution, which is problematic when dealing with non-stationary data [33,34]. This compromise is a common challenge in extensive data analysis, especially with large-scale datasets. Wavelet analysis is a powerful tool for handling evolving data. Unlike the Fourier transform, wavelet analysis is effective in dealing with non-stationary data [32]. It uses a wavelet function to analyze data at multiple scales, allowing the identification of broad and fine-scale patterns. This approach provides valuable insights into the frequency and time characteristics of the data [35], making it particularly suitable for datasets with dynamic frequency patterns. Wavelet analysis is widely used in various domains, including image analysis, telecommunications, anomaly detection, and biomedical data analysis [36–41]. The wavelet transform builds upon the short-time Fourier transforms, offering high-frequency resolution at lower frequencies and high-time resolution at higher frequencies [42]. This characteristic overcomes the limitations of Fourier transforms, making it valuable in extensive data analysis.

On the other hand, Pearson correlation is a useful method for detecting cyberattacks as it can identify unusual network traffic patterns [27,43,44]. It measures the linear relationship between variables and effectively detects deviations from expected behavior in network data. Analyzing correlations between various network parameters can reveal anomalies and potentially malicious activities. The resilience to outliers and ability to capture positive and negative correlations make Pearson correlation a valuable tool for identifying subtle attack patterns in complex network environments.

Securing the IVN is challenging and time-consuming due to the unique data profiles of each vehicle. Conventional IDS solutions require customization for each vehicle type and struggle with firmware updates and configuration changes. A universal IDS is needed to simplify security measures, eliminate the need for customization, and ensure robust protection against evolving data patterns and adversarial attacks. The solution incorporates advanced machine learning techniques and a two-stage verification process, providing a standardized and effective security solution for all types of vehicles.

3. Features Extraction and Data Preprocessing

3.1. Data Collection

Vehicle-related study requires data collection, and the On-Board Diagnostics II (OBD-II) system is a commonly used method. However, not all ECUs can be accessed via the OBD-II port, which limits the amount of data that can be collected. To overcome this limitation, researchers use the ECU Direct Approach (EDA) [45], which involves acquiring data through an internal gateway using line-tapping tools while accessing the vehicle's CAN network through an integrated central unit (ICU)[45]. The PEAK CAN system is the

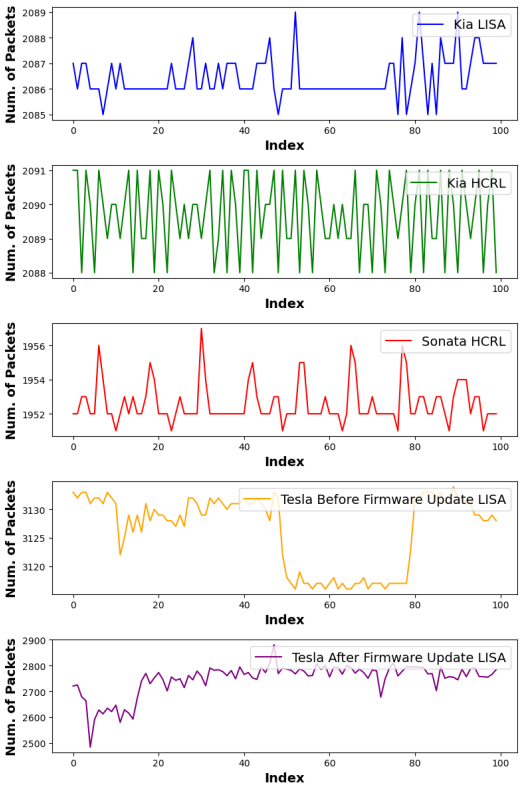

Figure 2. Data Generation Variability Among Vehicles

interfacing device for collecting normal driving data through the EDA method. This study employed a testbed framework incorporating real-world data to address safety concerns for attack data collection. In this controlled environment, attacks such as Fuzzing, DoS, and Replay were simulated to generate attack data for analysis and as input for deep learning models.

3.2. Feature Extraction

This study analyzes diverse vehicle data to detect unauthorized data injections in vehicles. Four datasets of different vehicles were examined to identify common features that can help improve vehicle security. The results show differences in internal IDs and payload data among vehicles. The CAN DBC formation causes differences between vehicles of different manufacturers and models [17]. The internal data structure may change significantly following a firmware update or a change in driver behavior. Fig. 1 presents driver-wise mean CAN payload heatmaps for three representative CAN IDs (0x0220, 0x04F0, and 0x0545). For each CAN ID, CAN frames were filtered by identifier, payload bytes were converted from hexadecimal to integer values, and mean byte values were computed per driver. Rows correspond to drivers, columns to payload bytes, and color intensity indicates the average payload magnitude using a unified color scale. While some bytes remain consistent across drivers, reflecting protocol-controlled fields, others exhibit clear driver-dependent variations. Notably, Bytes Three and Five in CAN ID 0x0220, Byte Six in CAN ID 0x04F0, and Bytes Five to Eight in CAN ID 0x0545 show systematic differences across drivers, suggesting that specific payload fields capture driver-dependent operational characteristics. As a result, conventional IDSs that directly use internal features such as CAN IDs and payloads face challenges in real-world performance. Therefore, this study avoids relying on such physical/internal features for attack detection and instead analyzes statistical features that exhibit common patterns indicative of unauthorized message injection.

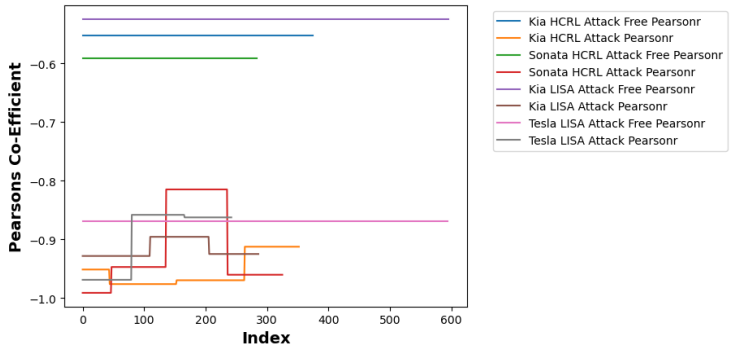

Figure 3. Pearson Coefficient Threshold

Table 1 illustrates that Kia (LISA) and Kia (HCRL) exhibit similar statistical features despite sharing the same manufacturer and model. Although Sonata (HCRL) may have fewer CAN IDs, its data generation and average time gap metrics are consistent with those of other vehicles. Furthermore, the Tesla (LISA) autonomous vehicle generates data at the highest rate and with the shortest time intervals between messages, prioritizing safety and relying on numerous sensors. The amount of data generated by the Tesla vehicle varies depending on the driver's activity or firmware updates, resulting in a noticeable discontinuity in the statistical features shown in Fig. 2, where the data generation process of electronic vehicles fluctuates arbitrarily compared to that of mechanical vehicles.

3.3. Data Generalization

The aim of this study is to convert time series data obtained from a vehicle's internal systems into frequency domain. The ultimate goal is data generalization to develop a universal IDS that can be applied to a wide range of vehicles. The data is segmented into intervals of 0.01 seconds, and various metrics are calculated for each segment. These metrics include the amount of data generated and the average time gap between data packets within the segment.

$$c_{j+1,k} = \sum_{n=-\infty}^{\infty} h_n \, c_{j,k+n2^j} + \sum_{n=-\infty}^{\infty} g_n \, d_{j,k+n2^j} \tag{1}$$

$$d_{j+1,k} = \sum_{n=-\infty}^{\infty} h_n \, c_{j,k+n2^j} - \sum_{n=-\infty}^{\infty} g_n \, d_{j,k+n2^j} \tag{2}$$

Next, a chunk of 100 values is selected for conversion into a high-resolution frequency domain representation. This process effectively converts 1 second (0.01 x 100) of statistical data into a frequency domain representation, which provides greater insight into the underlying frequency components of the CAN data. This approach allows for more detailed data analysis and can help identify patterns and trends within the CAN data.

The data conversion method used is the Discrete Wavelet Transform (DWT), and the wavelet coefficients extracted from this transformation are used as deep-learning input features due to their computational efficiency [23]. However, challenges arise due to variations in coefficient lengths, and deep learning methods inherently require assistance in handling multi-dimensional data simultaneously. To address these issues, a "padding" technique is applied to extend shorter coefficients with zeros. This ensures uniform data dimensions for precise deep-learning analysis.

The DWT of a function $x(t)$, as described in Eq. (1), is computed with respect to the wavelet function $\psi_{m,n}(t)$, where m and n represent the scale and translation parameters, respectively, and $T_{m,n}$ is the wavelet coefficient that captures how closely the signal $x(t)$

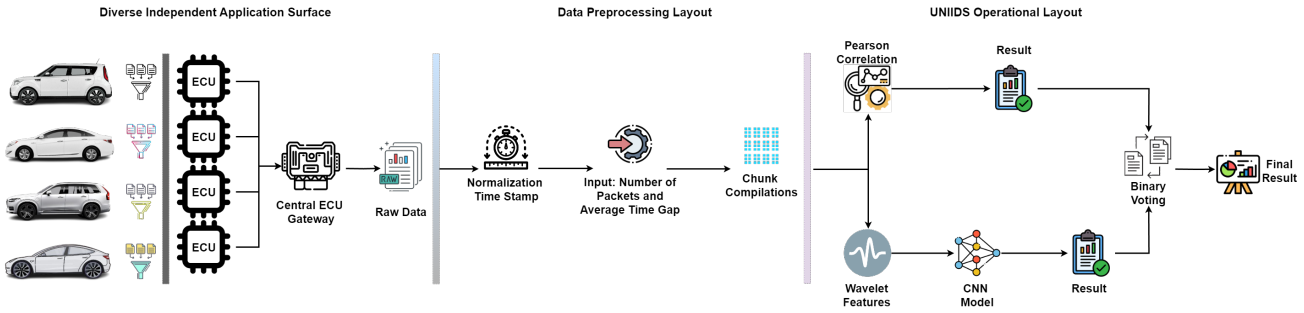

Figure 4. Abstract overview of the proposed detection system

matches the wavelet function at a particular scale m and translation n. This transformation involves integrating the signal $x(t)$ over the range from negative infinity to positive infinity using the wavelet function $\psi_{m,n}(t)$. The result decomposes the signal into its frequency components, allowing for the analysis of both broad and fine-scale patterns within the signal. This process is particularly useful for analyzing time-varying data, such as IVN data, by revealing its frequency and time characteristics.

For the wavelet transform, a decomposition level of 10 and a symmetric mode were chosen, where the Daubechies wavelet with eight coefficients (db8) was used for both training and testing data. Eq. (2) and Eq. (3) represent the computation of approximation coefficients $c_{j,k}$ and detail coefficients $d_{j,k}$, respectively, at different scales and positions, where h_n and g_n are the scaling and wavelet coefficients of the Daubechies 8 wavelet, and 2^j represents the scale factor.

To apply this transform iteratively and decompose the data into multiple levels of approximation and detail coefficients, the pywt.wavedec() function was used. The symmetric parameter determined whether the input data was extended symmetrically at the boundaries before applying the transform. The level parameter specified the number of decomposition levels to be computed.

4. Implementation of Experimental Methods: A Comprehensive Approach

The experiment utilized the ResNet-50 model, a CNN with 50 layers and shortcut connections to enhance gradient flow, which is highly regarded for its accuracy in image classification and object recognition tasks; further details can be found in [46], and the early stop method was implemented to prevent overfitting during training.

The process starts by collecting data from various CAN sources. This data is then sent through the gateway system to facilitate analysis. Timestamp alignment is used to synchronize the time intervals with the data generation events for each vehicle. For example, when generating 100 data packets, it is important to measure how long it takes for each vehicle to create these packets. By accurately aligning the timestamps, the data remains consistent and reliable, especially in applications where timing is critical. It also affects the response time of IDS. Once the timestamps are synchronized, the system quantifies packet counts and calculates the average time intervals between them. These features are then subjected to wavelet conversion to enhance their ability to detect attacks from unknown sources. A CNN model based on the ResNet-50 architecture is used for binary class classification in intrusion detection. The working principle process is illustrated in Algorithm 1. Initially, Algorithm 1 uses an initial chunk size of 0.01 seconds, which may vary for each vehicle based on their data generation rates. For Sonata, the chunk size is fixed at 0.01 seconds as a reference due to its relatively lower data generation rate. For Kia and Tesla, the chunk limits are set at 0.009 seconds and 0.0065 seconds, respectively. Subsequently, a secondary chunk with a fixed length of 100, approximately 1 second of data

Algorithm 1: Experimental Layout

Input: *data* *(List of raw dataframes)*
Output: *Result* *(Combined result)*
Step 1: Static Calculation for 0.01 Sec;
Initialize: S_1, Segment_{time}, start, end;
for *i in length* **do**
  $S_1 \leftarrow \{df_{ij} \mid \text{start}_{0.0\,\text{ms}} \leq df_{i1} \leq \text{end}_{10\,\text{ms}}, i = 1, \ldots, m\}$;
  $\text{Segment}_{time} \leftarrow S_1[\text{Num\_of\_Packets}], S_1[\text{Ave\_Time\_Gap}]$;
  start $\leftarrow$ end;
  end $\leftarrow$ end $+ 0.01$;
Step 2: Feature Selection 0 to 100 (1 Sec);
Initialize: $Seg_{out}, start_0, limit_{100}, length_N$;
for *i in range*$(start_0, length_N, limit_{100})$ **do**
  $Seg_{out} \leftarrow \text{df}_{input}[start_0 : limit_{100}]$;
Step 3: Wavelet Decomposition and Scaling;
Initialize: $wavelet_{db8}, mode_{sym}, level_{10}$;
for *i in* Seg_{out} **do**
  coeffs $\leftarrow$ pywt.wavedec$(Seg_{out}, wavelet_{db8}, mode_{sym}, level_{10})$;
  $\text{coeffs}_{scaled} \leftarrow \frac{S - x_{\min}}{x_{\max} - x_{\min}} \leftarrow$ coeffs;
Step 4: Time Series Wavelet Matrix 10 sets (1 Sec);
Initialize: $\text{CNN}_{out}, X_{\text{test}}, y_{\text{test}}$, start, end, length;
for *i in length* **do**
  $X_{\text{test}}[i] = X[\text{start} : \text{end}]$;
  $y_{\text{test}}[i] = y[\text{end}]$;
  $\text{CNN}_{out} \leftarrow \text{CNN}_{ResNet50} \leftarrow X_{\text{test}}$
Step 5: Pearson Correlation Calculation;
Initialize: Pearson_{out}, correlation, p-value;
for *i in* $Segment_{time}$ **do**
  correlation, p-value $=$ pearsonr$(i[\text{Num_of_Packets}], i[\text{Ave_Time_Gap}])$;
  if *correlation* ≤ -0.7 **then** $\text{Pearson}_{out} \leftarrow 1$;
  else $\text{Pearson}_{out} \leftarrow 0$;
Step 6: Binary Voting;
Initialize: Final_{out};
for i_1, i_2 *in* $CNN_{out}, Pearson_{out}$ **do**
  if $i_1 == 0 \wedge i_2 == 0$ **then** $\text{Final}_{out} \leftarrow$ Attack Free;
  else $\text{Final}_{out} \leftarrow$ Attack;

(varied vehicle to vehicle), is applied. This data is then converted into wavelets, generating ten wavelet coefficients as deep learning features. Finally, deep learning preprocessing is performed, and the extracted features are inputted into the model for non-linear and dynamic analysis, enabling the model to detect complex patterns and relationships within the data that are crucial for accurate predictions or classifications.

$$\rho = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} \tag{3}$$

This IDS has been designed with a universal approach to be compatible with all types of vehicles. However, in this experiment, it was found that the accuracy of deep learning models alone was relatively low compared to the conventional IDSs standards. To improve accuracy, a secondary rule-based IDS was added. These secondary IDS utilize the same features, data generation rate, and average time gap through Pearson correlation. Pearson correlation is a statistical measure, also known as Pearson's correlation coefficient, that helps determine the strength and direction of the linear relationship between two continuous variables. The coefficient ranges from -1 to 1, where 1 indicates a perfect positive correlation, -1 indicates a perfect negative correlation, and 0 denotes no linear correlation [27].

The Pearson correlation coefficient, denoted by ρ, measures the strength and direction of the linear relationship between two variables, X and Y. The term $\text{cov}(X, Y)$ represents the covariance, which quantifies how much X and Y change together. The standard deviations of X and Y, denoted as σ_X and σ_Y respectively, indicate the variability within each variable. A positive ρ suggests a positive correlation, meaning that as one variable increases, the other tends to increase, and vice versa. Conversely, a negative ρ indicates a negative correlation, where one variable tends to decrease as the other increases. In Eq. 4, attack-free and attack data are inputted separately to determine the threshold. Fig. 3 illustrates highly negative correlations between data generation and time gap in the attack data. When considering attack-free data for Kia and Sonata, the coefficient range is set above -0.6. However, for attack data, the coefficient range is set below -0.8. In the case of Tesla, the attack-free data crosses the threshold due to the higher data generation amount compared to mechanical vehicles, and the data generation amounts have a discontinuity in everyday driving situations portrayed in Fig. 2. This issue significantly affects the time gap sequence, resulting in a corresponding reaction in the covariance of Pearson. The combined IDS uses two independent algorithms, shown in Fig. 4, to determine whether data is an attack through a binary voting process. If both algorithms output 0, the data is considered attack-free. On the other hand, if either of the algorithms outputs a 1, the IDS classifies the data as an attack.

This Fig.4 illustrates the over all processing pipeline for UIDS. It shows multiple vehicles collecting raw data, which is then passed through preprocessing stages, likely filtering or feature extraction mechanisms. The preprocessed data is then fed into both Pearson correlation analysis and a wavelet transformation. The wavelet-transformed data is subsequently input into a ResNet-50 model. Afterward, the outputs from both models are combined using a binary voting mechanism to make a final decision, ensuring more robust and accurate results.

5. Result Evaluation

The study utilizes two frequency injections: low-rate periodic injections from HCRL and high-frequency injections from LISA. The study aims to evaluate the performance of a machine learning algorithm trained on two different vehicle datasets, Sonata and Kia, using low-frequency periodic injection during the training phase. The algorithm's effectiveness

Table 2. Performance metrics of UIDS (ResNet-50) across different vehicle datasets for low and high frequency injection attacks.

Train Vehicle	Low Frequency Periodic Injection						High Frequency Injection					
	Sonata Test (HCRL)			Kia Test (HCRL)			Kia Test (LISA)			Tesla Test (LISA)		
	F1 Score	**Acc**	**AUC**	**F1 Score**	**Acc**	**AUC**	**F1 Score**	**Acc**	**AUC**	**F1 Score**	**Acc**	**AUC**
Kia (HCRL)	0.90	0.90	0.91	0.95	0.95	0.95	0.87	0.87	0.89	0.87	0.87	0.89
Sonata (HCRL)	0.88	0.88	0.88	0.89	0.88	0.88	0.96	0.96	0.94	0.99	0.98	0.97

Table 3. Performance metrics of hybrid UIDS (ResNet-50 + Pearson) across different vehicle datasets for low and high frequency injection attacks.

Train Vehicle	Low Frequency Periodic Injection						High Frequency Injection		
	Sonata Test (HCRL)			Kia Test (HCRL)			Kia Test (LISA)		
	F1 Score	**Acc**	**AUC**	**F1 Score**	**Acc**	**AUC**	**F1 Score**	**Acc**	**AUC**
Kia (HCRL)	0.98	0.98	0.98	0.97	0.97	0.97	0.90	0.90	0.92
Sonata (HCRL)	0.98	0.98	0.98	0.98	0.98	0.98	1.00	1.00	1.00

was tested in four scenarios: Sonata to Sonata low-frequency periodic injection, Sonata to Kia low-frequency periodic injection, Sonata to Kia high-frequency injection, and Sonata to Tesla high-frequency injection. The same approach was applied to the Kia-trained algorithm.

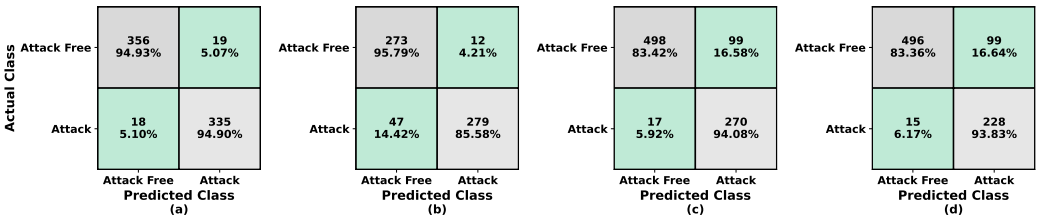

Figure 5. Confusion Matrix for ResNet-50 IDS Training on Kia and Test to Different Vehicles (a) Kia (HCRL) to Kia (HCRL), (b) Kia (HCRL) to Sonata (HCRL), (c) Kia (HCRL) to Kia (LISA), (d) Kia (HCRL) to Tesla (LISA)

The performance evaluation of a deep-learning model is shown in Fig. 5 and 6, as well as Table 2. These results specifically pertain to the deep-learning model's ability to detect attacks in various vehicle data scenarios. When trained on attack-free data for Kia and Sonata, the detection accuracy was approximately 96%, as demonstrated in Fig. 5 (a) and (b), using low-frequency injection training. However, the accuracy dropped to around 95% for detecting malicious data in Kia, and 86% for Sonata. In the case of high-frequency injection, the accuracy of detecting attack-free data for Kia and Tesla was relatively low at 84%, as seen in Fig. 5 (c) and (d). Despite this, malicious data was accurately detected at approximately 94% for both vehicles. Moving on to the accuracy matrix presented in Table 2, the model trained with Kia achieved an accuracy rate of 90-95% for low-frequency periodic attacks, and 87%-89% for high-frequency injection attacks.

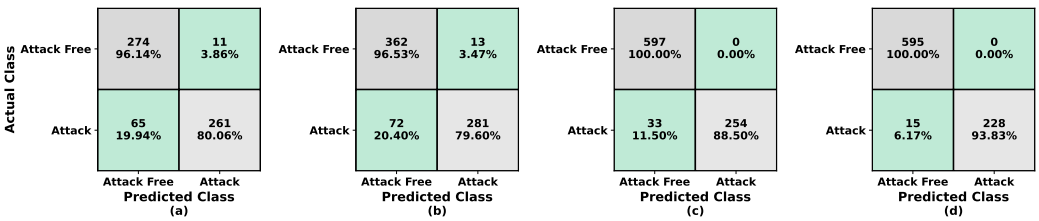

Figure 6. Confusion Matrix for ResNet-50 IDS Training on Sonata and Test to Different Vehicles (a) Sonata (HCRL) to Sonata (HCRL), (b) Sonata (HCRL) to Kia (HCRL), (c) Sonata (HCRL) to Kia (LISA), (d) Sonata (HCRL) to Tesla (LISA)

In comparison, Sonata trained model outperformed Kia trained model in detecting low-frequency injection attacks, as shown in Fig. 6 (a) and (b). Sonata achieved a detection accuracy of over 96% for attack-free data for both vehicles, while Kia's accuracy for attack data was only approximately 80%. However, Sonata had a 100% accuracy in detecting attack free data on Fig. 6 (c) and (d) for high-frequency. But in attacks, both Kia and Tesla had detection accuracies ranging from approximately 90% to 94% for attack data. Similarly, both vehicles achieved close to 100% accuracy in detecting malicious data. The accuracy rates of the deep-learning model, as presented in Table 2, were 88% for low-frequency periodic attacks and 96%-99% for high-frequency injection attacks when trained with Sonata data. These results demonstrate the effectiveness of the deep-learning model in detecting attacks in various vehicle data scenarios. In order to improve the overall accuracy of the universal IDS, which currently has relatively modest accuracy compared to conventional IDSs standards, a hybrid approach has been introduced. This approach involves combining a deep-learning model with an additional IDS.

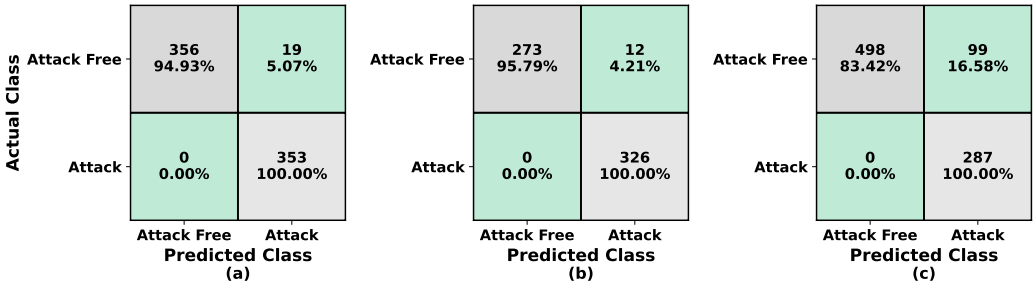

Figure 7. Confusion Matrix for Hybrid IDS Training on Kia and Test to Different Vehicles (a) Kia (HCRL) to Kia (HCRL), (b) Kia (HCRL) to Sonata (HCRL), (c) Kia (HCRL) to Kia (LISA)

A hybrid IDS was developed by combining a rule-based (Pearson correlation coefficients) IDS with a deep-learning model. The performance evaluation of the hybrid IDS is presented in Fig. 7, Fig. 8, and Table 3. For low-frequency injection training with Kia, as shown in Fig. 7 (a) and (b), the detection accuracy for Kia and Sonata attack-free data is approximately 96%. In comparison, malicious data was detected with 100% accuracy. In the case of high-frequency injection, as depicted in Fig. 7 (c), Kia attack-free data were accurately detected at around 84%, with malicious data accurately detected at 100%. Referring to the accuracy matrix presented in Table 3, the model trained with Kia achieved 98% and 97% accuracy for low-frequency periodic and 91% high-frequency injection, respectively. Compared to Sonata, Kia's accuracy performance showed a decrease of less than 1%.

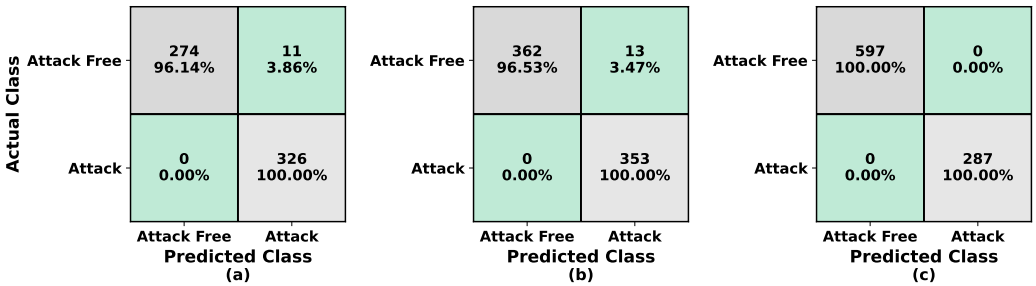

Figure 8. Confusion Matrix for Hybrid IDS Training on Sonata and Test to Different Vehicles (a) Sonata (HCRL) to Sonata (HCRL), (b) Sonata (HCRL) to Kia (HCRL), (c) Sonata (HCRL) to Kia (LISA)

In the case of low-frequency injection training with Sonata, as shown in Fig. 8 (a) and (b), the detection accuracy for Sonata and Kia attack-free data is approximately 97%. On the other hand, malicious data was detected with a 100% accuracy rate. For high-frequency

Table 4. Comparative Analysis of IDS Techniques and Performance Across Various Studies.

Approaches	Author	Data Sets	Features	Algorithm	Performance
Conventional	[28]	HCRL	Full Data	CNN, LSTM	100%
	[26]	HCRL	DLC, Payload	LSTM-GNB, ConvLSTM-GNB	100%
	[27]	Authors	CAN ID	Pearson, LSTM	98.45%
	[2]	HCRL, Authors	CAN ID, Time gap	TCE-IDS rule-based	99%
	[25]	Authors	Voltage Signals	SVDD	97%
Universal	[19]	Synthetic Data	Entropy	WINDS rule-based	98%
	[18]	Authors, SynCAN	Signal Features	Autoencoder	Signature
	[20]	LISA	CAN ID, Timegap, Hamming Distance	CNN	99%
Proposed	Rezanur et al.	HCRL, LISA	Entropy, Timegap	CNN	**97%-100%**

injection, as depicted in Fig. 8 (c), Kia attack-free data were detected accurately at 100%, with malicious data also accurately detected at 100%. Referring to the accuracy matrix presented in Table 8, the model trained with Sonata achieved an accuracy of 98% for low-frequency periodic injection and 100% for high-frequency injection. When comparing our results with other study, the overall accuracy meets the benchmark compared to conventional IDS and universal IDS shown in Table 4.

6. Discussion

Based on the evaluation of the experimental results, it has been observed that injecting data at low frequencies can lead to mispredictions. This is because the injected data is generated periodically and with a relatively small volume, causing the system to behave like a typical data flow. As a result, misclassification can occur. On the other hand, high-frequency injections on Kia-trained model to mistakenly predict attack-free data as an attack. This misidentification only happens in the case of replay attacks, which have similar characteristics to attack-free data. Two advanced deep-learning models have been implemented in intrusion detection systems: one from Kia and the other from Sonata. Remarkably, Kia's model is highly effective in detecting low-frequency injection attacks, while Sonata's model excels at detecting high-frequency injection attacks for mechanical vehicles. Interestingly, both hybrid model from Kia and Sonata perform equally well for mechanical vehicles. For optimal deployment of IDS in real-world scenarios, it is advisable to use a models with varying data injection frequencies.

During testing on Tesla, the hybrid model shows limited performance due to a discontinuity in data generation (see Fig. 2). The differing data generation processes between mechanical and electronic vehicles influence the time gap between packets, significantly impacting the Pearson correlation coefficient (see Fig. 3). While Pearson correlation effectively identifies linear relationships and anomalies in network traffic, it struggles with the complex, non-linear patterns typical of CAN bus data, where data generation varies with driving conditions and driver behavior, resulting in shifting temporal and spatial correlations. ResNet-50 mitigates this limitation by using deep learning to capture non-linear features, accommodating the dynamic nature of vehicle datasets and improving intrusion detection accuracy. In mechanical vehicles, omitting Pearson correlation reduces prediction accuracy by approximately 10%, underscoring a trade-off between accuracy and broader scalability. By combining Pearson's statistical insights with ResNet-50's capacity to adapt to evolving data patterns, the system enhances detection capabilities for mechanical vehicles, capturing both straightforward and intricate patterns across varied driving scenarios. Future studies should focus on developing UIDS electric vehicle-specific attack detection methods that optimize accuracy and scalability, strengthening detection robustness across a broader range of electric vehicle applications.

While the proposed hybrid UIDS improves cross-vehicle intrusion detection by combining wavelet-domain deep learning with a Pearson-correlation rule, several limitations

should be acknowledged. First, scalability to electric vehicles remains constrained: as observed in the EV evaluation, EV network traffic can exhibit discontinuous, sensor-driven dynamics and mode-dependent bursts that alter the packet-generation/time-gap relationship, which may cause benign behavior to cross the correlation threshold and reduce robustness unless EV-specific calibration or adaptive thresholding is introduced. Second, the experimental validation was performed on a limited set of vehicle datasets and attack classes under controlled collection conditions; therefore, broader benchmarking across additional vehicle platforms, diverse network architectures (e.g., CAN-FD and automotive Ethernet), and a wider range of adversarial strategies is required to strengthen claims of universal generalization and deployment readiness.

7. Conclusion

This study presented a UIDS that integrates wavelet-domain feature representation with a hybrid decision mechanism combining deep learning and Pearson-correlation–based anomaly screening. Evaluation across multiple vehicle platforms and two injection-frequency regimes demonstrated that data-generation heterogeneity and injection frequency materially influence IDS transfer performance, with particularly challenging behavior observed in the electric-vehicle case due to discontinuous data generation and its impact on the packet/time-gap relationship. Overall, the hybrid design supports cross-vehicle intrusion detection under heterogeneous in-vehicle network characteristics, while highlighting the need for electric-vehicle–aware refinement.

Future work will extend this research in three directions. First, electric-vehicle–oriented detection strategies should be developed to explicitly model discontinuous and event-driven traffic regimes, including adaptive thresholding (or regime-aware calibration) to reduce false alarms when benign EV behavior crosses correlation thresholds. Second, validation should be broadened across additional vehicle platforms, diverse network architectures (e.g., CAN-FD and automotive Ethernet), and a wider range of adversarial behaviors to strengthen evidence for universal generalization and deployment readiness. Third, deeper analysis of how electric-vehicle–specific data-generation processes and attack impacts propagate through the in-vehicle network is required to improve generalizability and sustain accuracy under evolving driving conditions and system updates.

Author Contributions: Md Rezanur Islam: Methodology, Analysis, Model Deployment. **Mahdi Sahlabadi:** Original draft preparation. **Munkhdelgerekh Batzorig**: Original draft preparation. **Kangbin Yim:** Conceptualization of this study, review, editing and supervision.

Data Availability Statement: The data supporting the findings of this study are publicly available. The primary dataset, titled "UIDS-CAN: A Multi-Vehicle CAN Intrusion Detection Dataset", has been archived on IEEE DataPort and is accessible at: https://ieee-dataport.org/documents/uids-can-multi-vehicle-can-intrusion-detection-dataset.

The dataset includes multi-vehicle CAN bus traffic collected under normal driving conditions and controlled cyber-attack scenarios (DoS, Fuzzing, and Replay) across heterogeneous platforms, including mechanical and electronic vehicles. Additionally, this study utilizes the HCRL dataset as a supplementary benchmark for low-frequency periodic injection analysis. No personally identifiable information is included, and all data were collected in compliance with applicable ethical and privacy guidelines.

Acknowledgments: This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the Convergence security core talent training business support program (IITP-20242710008611) supervised by the IITP (Institute for Information Communications Technology Planning Evaluation) and Soonchunhyang University Research Fund.

References

1. Contreras-Castillo, J.; Zeadally, S.; Guerrero-Ibañez, J.A. Internet of vehicles: architecture, protocols, and security. *IEEE internet of things Journal* **2017**, *5*, 3701–3709.
2. Yu, Z.; Liu, Y.; Xie, G.; Li, R.; Liu, S.; Yang, L.T. TCE-IDS: Time Interval Conditional Entropy-Based Intrusion Detection System for Automotive Controller Area Networks. *IEEE Transactions on Industrial Informatics* **2022**, *19*, 1185–1195.
3. Kim, M.; Oh, I.; Yim, K.; Sahlabadi, M.; Shukur, Z. Security of 6G enabled Vehicle-to-Everything Communication in Emerging Federated Learning and Blockchain Technologies. *IEEE Access* **2023**.
4. Hoang, T.N.; Islam, M.R.; Yim, K.; Kim, D. CANPerFL: Improve In-Vehicle Intrusion Detection Performance by Sharing Knowledge. *Applied Sciences* **2023**, *13*, 6369.
5. Choi, W.; Lee, S.; Joo, K.; Jo, H.J.; Lee, D.H. An enhanced method for reverse engineering CAN data payload. *IEEE Transactions on Vehicular Technology* **2021**, *70*, 3371–3381.
6. Gazdag, A.; Lestyán, S.; Remeli, M.; Ács, G.; Holczer, T.; Biczók, G. Privacy pitfalls of releasing in-vehicle network data. *Vehicular Communications* **2023**, *39*, 100565.
7. Jeong, S.; Kim, H.K.; Han, M.L.; Kwak, B.I. AERO: Automotive Ethernet Real-Time Observer for Anomaly Detection in In-Vehicle Networks. *IEEE Transactions on Industrial Informatics* **2023**.
8. Liu, J.; Zhang, S.; Sun, W.; Shi, Y. In-vehicle network attacks and countermeasures: Challenges and future directions. *IEEE Network* **2017**, *31*, 50–58.
9. An, Y.; Park, J.; Oh, I.; Kim, M.; Yim, K. Design and implementation of a novel testbed for automotive security analysis. In Proceedings of the Innovative Mobile and Internet Services in Ubiquitous Computing: Proceedings of the 14th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2020). Springer, 2021, pp. 234–243.
10. Cai, Z.; Wang, A.; Zhang, W.; Gruffke, M.; Schweppe, H. 0-days & mitigations: roadways to exploit and secure connected BMW cars. *Black Hat USA* **2019**, *2019*, 6.
11. Liu, L.; Lu, S.; Zhong, R.; Wu, B.; Yao, Y.; Zhang, Q.; Shi, W. Computing systems for autonomous driving: State of the art and challenges. *IEEE Internet of Things Journal* **2020**, *8*, 6469–6486.
12. Vdovic, H.; Babic, J.; Podobnik, V. Automotive software in connected and autonomous electric vehicles: A review. *IEEE Access* **2019**, *7*, 166365–166379.
13. Tesla Owners Online Forum. Diagnostic Port and Data Access, 2016.
14. Hoang, T.N.; Kim, D. Supervised contrastive ResNet and transfer learning for the in-vehicle intrusion detection system. *Expert Systems with Applications* **2024**, *238*, 122181.
15. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A comprehensive survey on transfer learning. *Proceedings of the IEEE* **2020**, *109*, 43–76.
16. Agrawal, S.; Sarkar, S.; Aouedi, O.; Yenduri, G.; Piamrat, K.; Alazab, M.; Bhattacharya, S.; Maddikunta, P.K.R.; Gadekallu, T.R. Federated learning for intrusion detection system: Concepts, challenges and future directions. *Computer Communications* **2022**, *195*, 346–361.
17. Islam, M.R.; Sahlabadi, M.; Kim, K.; Kim, Y.; Yim, K. CF-AIDS: Comprehensive Frequency-Agnostic Intrusion Detection System on In-Vehicle Network. *IEEE Access* **2023**.
18. Novikova, E.; Le, V.; Yutin, M.; Weber, M.; Anderson, C. Autoencoder anomaly detection on large CAN bus data. *Proceedings of DLP-KDD* **2020**.
19. Bozdal, M.; Samie, M.; Jennions, I.K. WINDS: A wavelet-based intrusion detection system for Controller Area Network (CAN). *IEEE Access* **2021**, *9*, 58621–58633.
20. Islam, M.R.; Oh, I.; Yim, K. Universal Intrusion Detection System on In-Vehicle Network. In Proceedings of the International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing. Springer, 2023, pp. 78–85.
21. Kim, D.; Im, H.; Lee, S. Adaptive Autoencoder-Based Intrusion Detection System with Single Threshold for CAN Networks. *Sensors* **2025**, *25*, 4174.
22. Shahriar, M.H.; Xiao, Y.; Moriano, P.; Lou, W.; Hou, Y.T. CANShield: Deep Learning-Based Intrusion Detection Framework for Controller Area Networks at the Signal-Level. *IEEE Internet of Things Journal* **2023**.
23. Srivastava, M. Revisiting signal analysis in the big data era. *Nature computational science* **2022**, *2*, 70–71.
24. Limbasiya, T.; Teng, K.Z.; Chattopadhyay, S.; Zhou, J. A systematic survey of attack detection and prevention in connected and autonomous vehicles. *Vehicular Communications* **2022**, p. 100515.
25. Xun, Y.; Deng, Z.; Liu, J.; Zhao, Y. Side Channel Analysis: A Novel Intrusion Detection System Based on Vehicle Voltage Signals. *IEEE Transactions on Vehicular Technology* **2023**.
26. Mansourian, P.; Zhang, N.; Jaekel, A.; Kneppers, M. Deep learning-based anomaly detection for connected autonomous vehicles using spatiotemporal information. *IEEE Transactions on Intelligent Transportation Systems* **2023**.
27. Jedh, M.; Othmane, L.B.; Ahmed, N.; Bhargava, B. Detection of message injection attacks onto the can bus using similarities of successive messages-sequence graphs. *IEEE Transactions on Information Forensics and Security* **2021**, *16*, 4133–4146.

28. Lo, W.; Alqahtani, H.; Thakur, K.; Almadhor, A.; Chander, S.; Kumar, G. A hybrid deep learning based intrusion detection system using spatial-temporal representation of in-vehicle network traffic. *Vehicular Communications* **2022**, *35*, 100471.
29. Wu, W.; Li, R.; Xie, G.; An, J.; Bai, Y.; Zhou, J.; Li, K. A survey of intrusion detection for in-vehicle networks. *IEEE Transactions on Intelligent Transportation Systems* **2019**, *21*, 919–933.
30. Khan, J.; Lim, D.W.; Kim, Y.S. Intrusion detection system can-bus in-vehicle networks based on the statistical characteristics of attacks. *Sensors* **2023**, *23*, 3554.
31. Guo, T.; Zhang, T.; Lim, E.; Lopez-Benitez, M.; Ma, F.; Yu, L. A review of wavelet analysis and its applications: Challenges and opportunities. *IEEe Access* **2022**, *10*, 58869–58903.
32. Chavez, M.; Cazelles, B. Detecting dynamic spatial correlation patterns with generalized wavelet coherence and non-stationary surrogate data. *Scientific reports* **2019**, *9*, 7389.
33. Parsons, S.; Boonman, A.M.; Obrist, M.K. Advantages and disadvantages of techniques for transforming and analyzing chiropteran echolocation calls. *Journal of Mammalogy* **2000**, *81*, 927–938.
34. Feichtinger, H.G.; Strohmer, T. *Gabor analysis and algorithms: Theory and applications*; Springer Science & Business Media, 2012.
35. Zhu, S.; Hadzima-Nyarko, M.; Bonacci, O. Application of machine learning models in hydrology: Case study of river temperature forecasting in the Drava River using coupled wavelet analysis and adaptive neuro-fuzzy inference systems model. In *Basics of Computational Geophysics*; Elsevier, 2021; pp. 399–411.
36. Prasad, L.; Iyengar, S.S. *Wavelet analysis with applications to image processing*; CRC press, 2020.
37. Lakshmanan, M.K.; Nikookar, H. A review of wavelets for digital wireless communication. *Wireless personal communications* **2006**, *37*, 387–420.
38. Kwak, B.I.; Han, M.L.; Kim, H.K. Driver identification based on wavelet transform using driving patterns. *IEEE Transactions on Industrial Informatics* **2020**, *17*, 2400–2410.
39. James, J.; Hou, Y.; Li, V.O. Online false data injection attack detection with wavelet transform and deep neural networks. *IEEE Transactions on Industrial Informatics* **2018**, *14*, 3271–3280.
40. Frangakis, A.S.; Stoschek, A.; Hegerl, R. Wavelet transform filtering and nonlinear anisotropic diffusion assessed for signal reconstruction performance on multidimensional biomedical data. *IEEE Transactions on Biomedical Engineering* **2001**, *48*, 213–222.
41. Han, M.L.; Kwak, B.I.; Kim, H.K. TOW-IDS: Intrusion Detection System Based on Three Overlapped Wavelets for Automotive Ethernet. *IEEE Transactions on Information Forensics and Security* **2022**, *18*, 411–422.
42. Kehtarnavaz, N. Frequency domain processing in digital signal processing system design. *Elsevier* **2008**, pp. 175–196.
43. Hoque, N.; Kashyap, H.; Bhattacharyya, D.K. Real-time DDoS attack detection using FPGA. *Computer Communications* **2017**, *110*, 48–58.
44. Gottwalt, F.; Chang, E.; Dillon, T. CorrCorr: A feature selection method for multivariate correlation network anomaly detection techniques. *Computers & Security* **2019**, *83*, 234–245.
45. Koh, Y.; Kim, S.; Kim, Y.; Oh, I.; Yim, K. Efficient CAN dataset collection method for accurate security threat analysis on vehicle internal network. In Proceedings of the International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing. Springer, 2022, pp. 97–107.
46. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.