Attack Detection for Intelligent Vehicles via CAN-Bus: A Lightweight Image Network Approach

Sheng Gao, Linchuan Zhang, Lei He, Xiaoyang Deng, Huilin Yin, *Member, IEEE*, and Hao Zhang

***Abstract*—This article investigates the security detection mechanism of intelligent vehicles under DoS attack. A lightweight CanNet-based attack detection mechanism is developed, which is suitable for both periodic and aperiodic DoS attacks. Different from the existing attack detection mechanisms, the proposed one utilizes a designed CAN image generation scheme to convert CAN traffic data into CAN images, which makes the attack visible and traceable. And the lightweight CanNet image classification network is constructed to detect the abnormalities in the generated CAN images. In order to exploit the maximum correlation of data attributes among the acquired CAN traffic data, copula entropy, the equivalent form of mutual information, is introduced into the CAN image generation scheme. In addition, an extended CAN image color coding scheme is also designed to cope with the secure detection of traffic data beyond the 12-bit CAN ID. Finally, the effectiveness of the designed detection mechanism is validated by the comparative experiments employing the CAN traffic data from YF Sonata (the type of car).**

***Index Terms*—CAN-bus traffic data, DoS attack, intelligent vehicles, lightweight classification network, attack detection.**

I. Introduction

With the rapid development of science and technology, automobiles are no longer a simple mechanical system. A growing number of advanced embedded electronic products are integrated into cars. Although the interconnection of electronic devices improves the comfort, functionality, and safety of driving, the increasingly rich interfaces make it easier for in-vehicle networks to become the target of hackers. In particular, the controller area network (CAN) [1], [2] is usually vulnerable to network attacks because of its lack of encryption, authentication and integrity verification. Once anonymous attackers take control of CAN, users of vehicles may face the security issues of information leakage, vehicles' losing control and so on. In recent years, the attacks against the CAN bus of vehicles have occurred frequently. For instance, the attack method against automotive CAN is proposed, which can control window lifting, warning lights, and antilock brake system [3]. The attack methods of the CAN network are fully analyzed [4], and it is shown that the attackers can hack into the car CAN-bus through physical connection such as OBD-II ports and remote access such as Bluetooth, WiFi and Cellular network. In 2019, Tencent Cohen Laboratory demonstrate that the remote wireless intrusion can be achieved by hijacking the vulnerabilities in in-vehicle infotainment systems of BMW. Additionally, they also make use of security flaws of the central gateway as an access to inject malicious messages into the internal CAN bus, thus the control of the underlying in-vehicle network is over taken [5]. Compared to traditional automobiles, modern automobiles face higher unprecedented network security risks. As cars become increasingly intelligent and connected, it is possible for in-vehicle systems to connect to the Internet, thereby exposing them to attacks from hackers and malware. Attackers may use vulnerabilities to hack into the control system of the vehicle, access vehicle information, hijack the vehicle, etc., which poses a potential risk to the security and privacy of the drivers [6]. Designing an effective intrusion detection system (IDS) for automobiles is a considerable challenge.

Recent years have witnessed massive research on CAN intrusion detection through various methods, which can generally be divided into signature-based, statistical-based and machine learning-based detection. The signature-based approach detects attacks by exploiting a set of identified characteristics or rules stored in the IDS's database module [7], [8]. Comparing the activity of a network with the attack features stored in IDS, if they match the stored malicious patterns, an alert will be triggered. A new automotive identification system, Voltage IDS, is proposed, which identifies the sender by detecting various signatures of the electrical CAN signal corresponding to the CAN message [9]. The signature-based approach is promising since it is not complex and can effectively improve the detecting accuracy of the known attacks. Nonetheless, this method requires periodic database updates and can only detect the known type of attacks.

Manuscript received 12 May 2023; accepted 15 July 2023. Date of publication 19 July 2023; date of current version 19 December 2023. This work was supported in part by the National Natural Science Foundation of China under Grants 62273255, 62350003, and 62088101, in part by the Shanghai International Science and Technology Cooperation Project under Grants 22510712000 and 21550760900, in part by the Shanghai Municipal Science and Technology Major Project under Grant 2021SHZDZX0100 and in part by the Fundamental Research Funds for the Central Universities. The review of this article was coordinated by Dr. Shiva Raj Pokhrel. *(Corresponding authors: Huilin Yin; Hao Zhang.)*

Sheng Gao, Lei He, Xiaoyang Deng, Huilin Yin, and Hao Zhang are with the College of Electronic and Information Engineering, the Department of Control Science and Engineering, Tongji University, Shanghai 200092, China (e-mail: 2110134@tongji.edu.cn; 2130738@tongji.edu.cn; dxytju@tongji.edu.cn; yinhuilin@tongji.edu.cn; zhang_hao@tongji.edu.cn).

Linchuan Zhang is with the Shanghai Research Institute for Intelligent Autonomous Systems, Tongji University, Shanghai 201210, China (e-mail: 2211059@tongji.edu.cn).

Digital Object Identifier 10.1109/TVT.2023.3296705

The statistical-based IDS method, comparing with the current statistical observations, has the pre-determined normal network behavior. When the deviation reaches a certain threshold, it will trigger the alarm. The concept of entropy-based vehicular network attack detection is introduced, which is effective for detecting DoS attacks [10]. The information entropy value is often adopted to describe the degree of confusion and uncertainty of information, which can be utilized in CAN network detection to reflect the communication situation. Therefore, the increasing number of attacks would raise the value of entropies which indicates that intrusions have occurred in the CAN bus. A novel information entropy-based method is proposed to solve the intrusion detection problem in a vehicular network environment [11], in which a fixed number of messages are employed as sliding windows to avoid the interference of information entropy caused by different baud rates and non-periodic CAN messages. The statistically based intrusion detection schemes do not require prior knowledge of normal behavior. Instead, it can learn the expected behavior of the system through observation. However, the standard library of normal models in statistical-based intrusion detection schemes is not easy to set, which may lead to false positives and false negatives. Note that the small amount of malicious messages injected by malicious attackers is difficult to identify, which motivates our reasearch.

Machine learning-based intrusion detection technology adopts latest artificial intelligence approaches to learn network data transmission behavior. The detection accuracy can be improved by training models continuously. A single-class branch vector machine based on the kernel of the radial basis function is proposed, which first learns the normal behavior of the baseline, then the abnormal behavior is classified according to the deviation [12]. An intrusion detection technology utilizing deep neural network is proposed in [13], where the low-dimensional features of CAN packets extracted are trained and then appllied to identify normal packets and hacker packets. An anomaly detection method based on long-short time memory neural network (LSTM) is proposed in [14], which can predict the packets of the next state. By comparing the difference between the predicted value and the actual state value with the pre-set threshold, the abnormalities are judged to exist or not. Anomaly detection based on machine learning has the advantages of flexibility, adaptability, and more. However, the computing capacity of in-vehicle ECUs is often limited, and machine learning-based intrusion detection requires large amount of computing resources, affecting the real-time nature of detection.

Consequently, in view of the aforementioned CAN intrusion detection method, the current IDS of in-vehicle CAN still has the following challenges.

- High real-time requirement. If the vehicle's CAN network anomaly detection technology can not meet the requirements of real-time calculation, it may eventually lead to vehicle's failure and accidents.
- Get rid of the dependence of the detection mechanism on preset thresholds. Preset thresholds can have a bad impact on the detection rate and the false alarm rate of detection.
- Low memory occupation. If the detection mechanism takes up too much memory, it will place a high demand on the memory space of the intelligent vehicles.

Motivated by these challenges, a novel lightweight CanNet-based attack detection approach is proposed. The main contributions of this article are summarized as follows:

1) A more stealthy aperiodic DoS attack is proposed. This attack strategy is exploited to optimize the designed detection mechanism.
2) A new CAN image encoding scheme is developed. Compared to the methods utilizing pure data, this scheme has better visualization for the attack. Meanwhile, it is also more conducive to the traceability of the attack moment than the existing one-hot encoding method.
3) A lightweight CanNet-based attack detection mechanism for intelligent vehicles via the CAN-bus is designed. A pooling layer and a fully connected layer are removed to obtain the lightweight detection mechanism, i.e., with the advantage of low memory occupation. And by increasing the width of network, more features can be captured to improve the performance of detection.

The rest of the article is organized as follows. Section II describes related work on classification networks and detection mechanisms. The detection problem under DoS attacks for intelligent vehicles via CAN-bus is introduced and described in Section III. Section IV illustrates the encoding scheme for converting CAN traffic data into CAN images and the design of the lightweight CanNet-based detection mechanism. In Section V, the effectiveness of the proposed mechanism is demonstrated by the comparative experiments. Finally, the discussions and conclusions are presented in Section VI.

Notations: $\lfloor M \rfloor$ is defined as the largest integer not greater than M, $|M|$ denotes the absolute value of the number M. Let 0X be the prefix of a hexadecimal number. $INT(M)$ represents the function that converts a number M in other decimal levels to a decimal number. $log(M)$ stands for the natural logarithm of a positive number M. $\sqrt{M}$ indicates the square of a non-negative number M.

II. Related Work

In this section, a literature review of the work, which is classical and applied in comparative experiments, related to convolutional neural networks and lightweight networks in image classification, is presented. Meanwhile, the review of previous anomaly detection mechanisms is made to enhance the significance and motivation of our research.

A. Convolutional Neural Network

As a deep learning method, convolutional neural network (CNN) specializes in image recognition and is widely utilized in many fields [15], [16]. It is mainly made up of convolutional layer, pooling layer, fully connected layer and classification layer. As the core module of the CNN, the convolutional layer is connected to other layers through sparse connections. The pooling layer can further compress the data. After the convolution and pooling operations of the network, the output results are

usually flattened, which is benefit to facilitate connection with the fully connected layer and realize signal classification and recognition ultimately.

VGG16 [17] and DenseNet121 [18] are typical networks based on CNN. VGG16 is a convolutional neural network model developed by the Visual Geometry Group (VGG) of the university of Oxford, which consists of 13 convolutional layers, 3 fully connected layers and 5 pooling layers. VGG16 uses multiple 3×3 convolutions and enlarges the depth of the network to improve task performance. However, the fully connected layer of the VGG16 model has many parameters, occupies a lot of memory, and consumes a lot of computing resources, which makes the VGG16 model encounter obstacles in front-end deployment. Furthermore, VGG16 lacks an effective method to prevent the problems of gradient vanishing, gradient explosion and slow convergence speed. DenseNet121 directly reuses features by densely connecting all the previous layers with the latter layers to solve the gradient vanishing problem. The concatenate operation of DenseNet121 allows a large number of features to be reused, and the channels of a unique feature map for each layer are reduced, which bringing about less parameters and more efficient calculation. In order to perform multiple concatenate operations, the data needs to be copied multiple times, and the memory increases rapidly, which requires certain memory optimization techniques.

The above deep convolutional neural networks have improved the performance of multiple computer vision tasks to a new level. The general trend is to build deeper and more complex networks to achieve higher accuracy. However, these networks do not necessarily meet the requirements of mobile devices in terms of scale and speed.

B. Lightweight Convolutional Neural Network

The lightweight convolutional neural networks can be designed by compressing the normal convolutional neural networks by quantization, pruning and knowledge distillation methods. These methods can reduce the complexity and computation cost of the network [19]. Another kind of method is to directly design lightweight deep convolutional neural networks. The typical lightweight deep convolutional neural network includes MobileNet [20], ShuffleNet [21] and Efficientnet [22]. MobileNet applies depthwise separable convolution instead of traditional convolution to reduce computational cost and improve network efficiency. ShuffleNet utilizes pointwise group convolution instead of 1×1 convolution operation to reduce the complexity of convolution operation. In order to overcome the side effects of group convolution, a channel shuffling operation is also proposed to improve the features information flow between channels. EfficientNet can make the networks have better classification accuracy by balancing network width, depth and image resolution [22]. The above networks are all deep neural networks, which are designed to learn deeper features to improve recognition accuracy. In the attack signal data detection task of embedded mobile devices, more fine-grained features are needed than deep features, due to gradient disappearance and computing resources limited issues.

For lightweight convolutional neural networks based on LeNet5, reducing the depth and increasing the network width are necessary to avoid increasing the complexity of the network [23]. The traditional LeNet5 network is the first CNN that is proposed for handwriting image recognition. It is composed of 2 convolutional layers, 2 pooling layers, 2 fully connected layers and a classification layer. Six 5×5 convolutional kernels are applied to the image in the first layer to extract features. Maxpooling layers are added in the second and fourth layer for downsampling. The third layer is composed of sixteen 5×5 convolution kernels. In the fifth and sixth layers, fully connected layers are employed. Since the LeNet5 network is designed for handwriting image recognition, the convolutional network structure needs to be redesigned according to the task requirements to compress the number of training parameters. The above work motivates us to investigate the anomaly detection mechanism with a lightweight network where the detection performance is guaranteed while memory occupation is reduced.

C. Anomaly Detection Mechanisms

Network security has become an issue that cannot be ignored in the vehicular networking systems [24]. Although the vehicle CAN bus is the core of connected vehicle communication, it has also become one of the key targets of hackers. DoS attack in CAN bus is a common attack method, in which the attacker sends a large number of forged messages or error messages to saturate the CAN bus of the target vehicle and cause the vehicle control system to fail [25]. In order to ensure the security and stability of the vehicle networking system, researchers have proposed many CAN-bus DoS attack detection methods based on machine learning or deep learning, which can effectively identify and filter out abnormal messages to ensure the normal operation of the vehicle [26], [27], [28].

A deep learning based approach named Rec-CNN is proposed in [29], which employs a convolutional neural network and recurrence images for detecting attacks on the CAN bus. Unlike existing methods, this approach captures the temporal dependencies in the sequence of CAN frame arbitration IDS using recurrence plots. The proposed method was tested on a publicly available dataset and a target vehicle, achieving an accuracy of 99.9% at an attack frequency of 10 ms. The IDS-IVN in [30] is another deep learning-based attack detection method for in-vehicle networks. IDS-IVN utilizes a compact representation of traffic features and convolutional neural networks to extract and classify them into intrusive and non-intrusive categories. Performance is demonstrated on a benchmark dataset and compared with existing methods, achieving 99% accuracy and a 0.32% false-positive rate for intrusion detection. A collaborative deep support vector data description (Coll-DSVDD) method is proposed in [31] to improve the detection accuracy of vehicular network attack detection. The method combines deep convolutional neural networks and support vector data description in multiple vehicular networks to generate an optimized network without retraining. A self-supervised method for detecting anomalies in CAN using noise-laden pseudo-normal data is

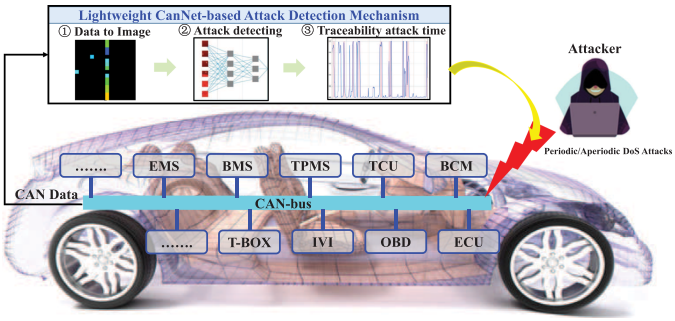

Fig. 1. Structure of the lightweight CanNet-based attack detection mechanism for DoS attacks.

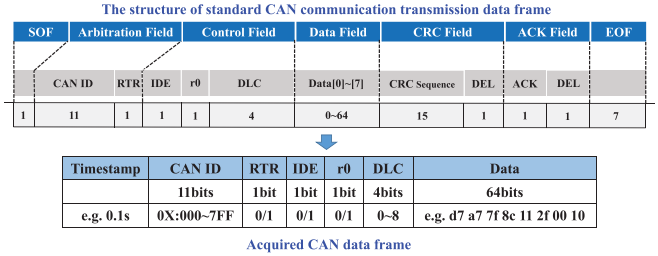

Fig. 2. Structure of standard CAN-bus protocol transmission and acquired CAN data frames.

proposed to address the issue of data dependency in supervised learning [32]. The method comprises a generator and a detector, both of which are deep learning models. The generator generates pseudo-normal traffic data using normal network traffic, which is then added with noise to make it deviate slightly from true normal data. The anomaly detector is trained to classify normal traffic and the noise-laden pseudo-normal traffic as normal and anomalous, respectively. Experimental results indicate that the proposed method outperforms other semi-supervised learning methods and is significantly better at detecting unknown attacks.

These methods mentioned above can improve the security of vehicle CAN bus to a certain extent. However, further research and development of more efficient and accurate attack detection methods are still needed to cope with the increasingly complex network environment and attack methods.

III. Problem Formulation

A. CAN-Bus Based Intelligent Vehicle System

The CAN-bus, which plays a significant role in the communication between control units within intelligent vehicles, is widely applied [33]. In such vehicles, the electronic control units (ECU), such as engine management system (EMS), battery management system (BMS), tire pressure monitoring system (TPMS), etc., all transmit information via the CAN-bus communication protocol as presented in Fig. 1. The detailed standard transmission mode of this communication protocol is presented in Fig. 2. The standard CAN communication format and the related profiles are detailed in [34], [35]. The format of CAN data acquired in this article is illustrated below.

Firstly, the timestamp indicates the time in seconds when the data was transferred. Then, the arbitration field containing the CAN identifier field (CAN ID) and the remote transfer request (RTR), where the CAN ID is an 11-bit identifier in the hexadecimal range (0X0000∼ 0X07FF) and is a functional address by which the CAN receiver filters the data frames. RTR is adopted to distinguish whether the frame is a data frame or a remote frame, where 0 is a data frame and 1 is a remote frame. After that, the control field includes the identifier extension bit (IDE), the reserved bit 0 (r0) and the data length code (DLC), where IDE is applied to indicate whether the frame is in standard or extended format, the r0 is reserved for subsequent use, and the DLC denotes the length of the data actually sent in the frame. Finally, the data [0]∼[7] are transmitted to the corresponding CAN ID via the CAN-bus. As a plaintext communication protocol, CAN-bus is broadcast to communicate among ECUs. It adopts the priority arbitration mechanism, that is, the smaller the CAN ID, the higher priority of communication. This mechanism is exploited by attackers to create and implement cyber attacks to disrupt normal CAN communication.

B. DoS Attacks

In this subsection, the DoS attack in which the attacker performs the attack periodically is discussed first, and then, a more stealthy DoS attack that utilizes an aperiodic time to perform the attack is studied. For the periodic DoS attacks, the attack strategy [36] against the CAN-bus is adopted, in which the DoS attacker occupies the CAN-bus by sending high-priority packets periodically and frequently, thus realizing that the CAN-bus cannot serve other normal control units, resulting in the crash of the CAN network. The periodic DoS attack can be described in mathematical form as follows,

$$\zeta^p(k) = \begin{cases} 1, & k = \lfloor \frac{n\tau_2}{\tau_1} \rfloor, n = 1, 2, 3, \ldots, \lfloor \frac{k_f}{\tau_2} \rfloor, \\ 0, & \text{otherwise}, \end{cases} \tag{1}$$

where $\zeta^p(k)$ is an indication of whether the DoS attack occurred at time k, $k = 1, 2, 3, \ldots k_f$, k_f is the moment of termination. $\zeta^p(k) = 1$ means that the DoS attack occurred at the current moment. Conversely, it implies that the DoS attack is not occurring at the current moment. τ_1 and τ_2 are the vehicle system sampling moment and the periodic DoS attack occurrence time designed by the attacker, respectively. There are two scenarios for τ_1 and τ_2, when $\tau_1 > \tau_2$, the attacker implements DoS attacks more frequently; when $\tau_1 \leq \tau_2$, the attacker carries out DoS attacks less frequently.

For the aperiodic DoS attack, its disorderly characteristics make it difficult to be detected. This attack can be designed as

$$\zeta^{ap}(k) = \begin{cases} 1, & k = \sum_{i=1}^{n} \varrho(i), n = 1, 2, 3, \ldots, \lfloor \frac{k_f}{D} \rfloor, \\ 0, & \text{otherwise}, \end{cases} \tag{2}$$

where $\varrho(i)$ is the random factor, which is a random integer from 0 to D selected by the random function $rand$, D is a finite positive integer satisfying $D \leq k_f$. $\zeta^{ap}(k)$ and $\zeta^p(k)$ represent similar purposes, namely whether the DoS occurs at time k.

Remark 1: There are two primary approaches for the DoS attacker to implement the attack, namely remote access and

physical access [37]. Telematics services provided by car manufacturers, such as Toyota's G-BOOK, Rongwei's inkaNet and Nissan's CarWings, are utilized by the attacker to carry out remote attacks. In this way, the attacker is not restricted by the distance. The OBD-II port of the vehicle can also be used by the attacker to gain physical access to the CAN bus of the vehicle to execute the attack.

Remark 2: For the CAN-bus-based intelligent vehicle system, it is simpler for an attacker to implement the DoS attack at or near the moment of vehicle sampling time τ_1 to achieve maximum damage, and for this reason, the effective number of DoS attacks by an attacker is highest when there is an integer multiple relationship between the vehicle sampling time τ_1 and the periodic attack time τ_2, which is beneficial for the attacker to minimize resources and maximize interests. In addition, compared to periodic DoS attacks, the maximum threshold D of the random factor for aperiodic DoS attacks needs to be set much smaller than the termination time k_f by the attacker, so that the number of effective aperiodic DoS attacks will increase.

C. Problem Statement

Advancements in networks and unmanned vehicle technology have heightened security threats to vehicles from external sources, resulting in numerous challenges. The anomaly detection is one of the most significant issues in ensuring the safe-driving of vehicles. Meanwhile, inspired by the rapidly updated, widely used and highly accurate image classification techniques such as VGG16 [17], Densenet121 [18], etc., it can be applied in the design of attack detection to enhance the performance of the detection mechanism. The main research problem in this article is to design a high-accuracy and real-time lightweight detection mechanism for CAN-bus-based intelligent vehicles subjected to the periodic DoS attack (1) and the aperiodic DoS attack (2).

IV. The Design of Attack Detection Mechanism

In this section, the attack detection mechanism is designed and described. The overall attack detection flow is demonstrated in Fig. 3. The detection process is mainly divided into two aspects, one is training the lightweight CanNet network utilizing the labelled CAN traffic data, and the other is detecting the real CAN traffic data on the basis of the trained network parameters. The main modules of the designed attack detection mechanism, including anomalous data attribute analysis, correlation analysis, data-to-image mechanism, lightweight CanNet network, and traceability of attack time, are depicted in detail below.

A. Anomalous Data Attribute Analysis

Fig. 2 indicates that the acquired CAN bus data frames contain a variety of data attributes, most of which have no relationship with the attacks. Consequently, the anomalous data attributes need to be analyzed to select the attribute that best characterizes the attack. Based on statistical methods, the frequency of the data corresponds to the data attribute selection method is adopted. Since the priority arbitration mechanism is utilized by attackers

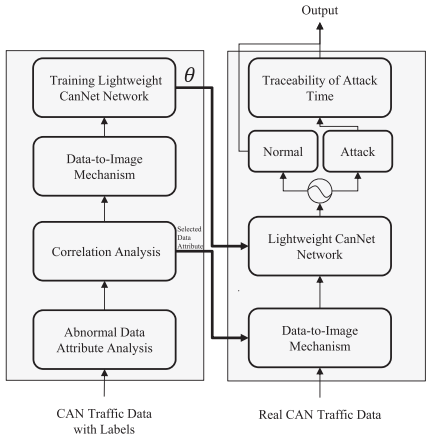

Fig. 3. Flowchart of the attack detection mechanism based on lightweight CanNet network.

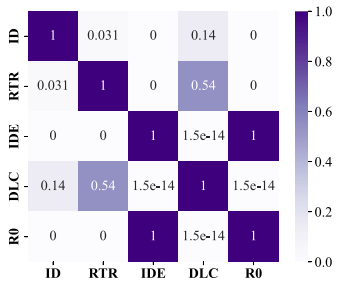

Fig. 4. Heat map of the correlation between CAN ID and CAN-bus data attributes based on the absolute value of CE.

to implement the DoS attack, the abnormal CAN ID of the CAN-bus data frame appears most frequently and is then selected as the data attribute that best characterizes the attack. In addition, the primary and essential attribute is the timestamp attribute, which is necessary for the attack detection mechanism to find out when the attack occurs.

B. Correlation Analysis

For the detection of whether the acquired data is subject to the DoS attack, it is not sufficient to only consider the data attributes that behaves most abnormally after the attack, e.g. CAN ID in DoS attacks, Data packets in injection attacks, etc. Therefore, to provide a more comprehensive characterization of the attack, correlation analysis is required and illustrated below. The degree of correlation between the remaining attributes of the CAN data frame and the CAN ID is calculated according to the copula entropy (CE) [38], [39], and demonstrated in Fig. 4. The opposite value of the copula entropy $\mathfrak{F}$ is considered to filter out the weakly correlated attributes and select the strongly correlated ones. The copula entropy for attributes M_1 and M_2 is determined as follows,

$$\mathfrak{F}(M) = -\sum_{\psi} \Gamma(\psi) \log(\Gamma(\psi)), \tag{3}$$

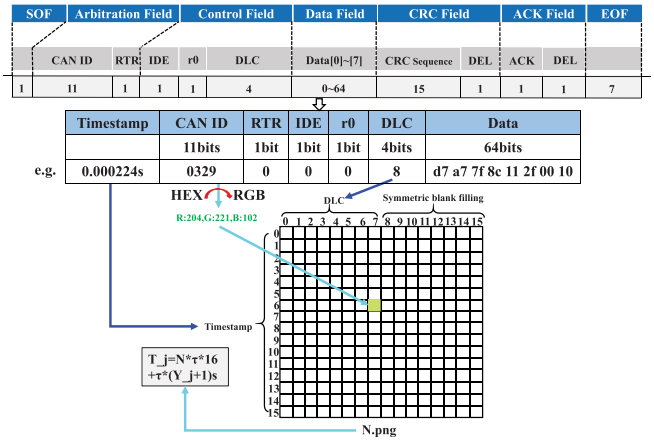

Fig. 5. Schematic diagram of CAN image mapping based on CAN-bus data.

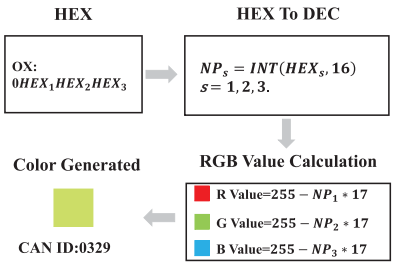

Fig. 6. Diagram of the color generation mechanism based on CAN ID.

where $M = [M_1, M_2]$ is a vector composed of data attribute variables, ψ is the marginal functions of M, and $\Gamma(\psi)$ is copula density computed via [39].

It is evident from Fig. 4 that the DLC attribute has the strongest correlation with the CAN ID compared to the rest of the data attributes. For this reason, DLC data attributes are selected to plot the CAN image and the rest of the data attributes are filtered out.

Remark 3: In contrast with the classical Pearson Correlation (PC) coefficient [40] which can only measure linear correlations and implicitly make Gaussian assumptions see Appendix A for details, CE is a correlation tool that removes the Gaussian assumptions can be applied to nonlinear, non-Gaussian, multivariate, symmetric, monotonic transformation invariance, etc. In addition, CE is an equivalent to Mutual Information (MI) [39], where MI is a measure of interdependence among variables in information theory, for which CE is adopted to help select attributes with strong correlations.

C. Data-to-Image Mechanism

The data attributes selected in the previous subsection are utilized to draw CAN images, in which the details of the CAN image drawing and the image color coding mechanism are displayed in the Figs. 5 and 6, respectively.

The specific procedures for conversion from CAN-bus data to CAN images are as follows.

- First, the physical meaning for the horizontal and vertical axes of the image is determined by the data attributes selected in Section IV-B. It should be mentioned that the timestamp is taken as the vertical axis and the DLC as the horizontal axis in this article. In order to highlight the CAN ID corresponding to the maximum value of 8 of the DLC in the image, the horizontal axis is expanded to 16 bits and the remaining 8 bits are filled with 0, i.e., black.
- Then, the points corresponding to the determined time and DLC are filled with the RGB color corresponding to the CAN ID.
- Finally, the data of the 16-bit timestamps corresponding to the horizontal axis are selected to draw the corresponding CAN images.

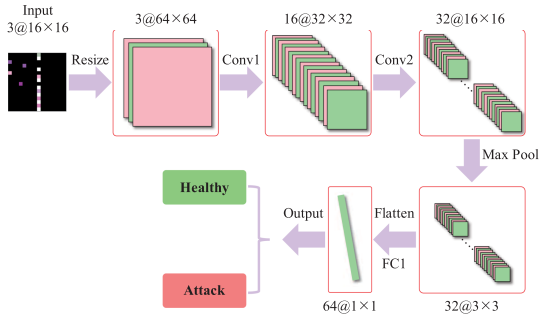

Fig. 7. Structure diagram of the lightweight CanNet network.

Remark 4: Compared to the one-hot encoding method, our proposed coding method directly reflects the corresponding CAN ID through RGB colors, which not only speeds up the conversion but also facilitates the DoS attack visualization. Since the CAN ID has a range of 0X000 $\sim$ 0X7FF, the mechanism in this subsection employs the three hexadecimal values of the CAN ID corresponding to the three RGB channels to generate the CAN image. It is foreseeable that the number of devices and applications in the vehicle will gradually increase to boost driver's comfort and convenience. When the hexadecimal value of the CAN ID exceeds 12 bits, the CAN traffic data to CAN image mechanism designed in this subsection will not work. Therefore, the scheme designed in Appendix B is a preferred method for generating CAN images in this scenario. The reasons for employing the scheme designed in Section IV-C instead of the scheme designed in Appendix B to generate CAN images are elaborated in Section V-C.

D. Lightweight CanNet Network

For the above generated CAN images, a lightweight CanNet network is developed to detect anomalous parts of the CAN images, where the overall structure of CanNet is exhibited in Fig. 7. The design of the entire network can be divided into two parts: convolutional feature extraction and fully connected layers for dimensionality reduction. In convolutional feature extraction part, CanNet adopts two convolution layers to increase the channel dimension to 32, so that all image information can be extracted as efficiently and comprehensively as possible. The convolutional neural networks in this part are designed to solve image classification problems based on deep learning methods. These networks are considered as a complex function, which employs image data as input and outputs the class label of the

TABLE I
A COMPARISON BETWEEN LENET5 [23] AND CANNET

Layer	Kernel Number × Kernel Size		Output		Number of Weights	
	LeNet5	CanNet	LeNet5	CanNet	LeNet5	CanNet
Conv1	$6 \times 5 \times 5$	**16 × 3 × 3**	$6 \times 60 \times 60$	**16 × 32 × 32**	156	**160**
Pool1	2×2	-	$6 \times 30 \times 30$	-	12	-
Conv2	$16 \times 5 \times 5$	**32 × 3 × 3**	$16 \times 26 \times 26$	**32 × 16 × 16**	7932	**2336**
Pool2	2×2	**8 × 8**	$16 \times 13 \times 13$	**32 × 3 × 3**	32	**64**
Connected Layer1	120	**64**	$1 \times 1 \times 120$	**1 × 1 × 64**	324600	**18496**
Connected Layer2	84	-	1×84	-	10164	-
Classification Layer	10	**2**	1×10	**1 × 2**	840	**130**

image. The function can be formulated as

$$y = f(X, \theta), \tag{4}$$

where y is the output label, X stands for the CAN image data, θ denotes the parameters of the neural network, and $f(\cdot)$ represents the neural network.

The convolution operation mainly performs feature extraction and feature mapping on the input data to obtain different feature information. In order to reduce network computation as well as spatial dimensionality, and improve generalization capability, a pooling layer (Max Pool) is added after the convolution layer. Through the max pooling operation, the size of the feature map is reduced, and the remaining parameters provide all necessary information for image classification. The pooling layer mainly focuses on feature selection and information filtering on the feature maps transmitted by the convolutional layer to improve the fault tolerance of the model. The output of the Max Pool layer is linked with the fully connected layer (FC1), which converts high-dimensional channels into expected output dimensions. Generally the output of the model is a high dimensional vector, which could be treated as a distribution $q(X)$. By evaluating the difference between $q(X)$ and the real distribution $p(X)$ of the input CAN image, the training process adjust the gradient direction to update parameters of the model. The cross entropy loss $H(p, q)$ calculated in a similar way to (3) is expressed as

$$H(p, q) = -\sum p(X) \log(q(X)). \tag{5}$$

The entire training process of lightweight CanNet network is essentially an optimization process, this process can be written as

$$\min_{\theta} \quad \rho(\theta),$$
$$\text{where} \quad \rho(\theta) = \mathbb{E}_{(X,y)\sim\mathcal{D}}[L(X, y, \theta)],$$

where $\mathbb{E}$ means expectation, $\mathcal{D}$ is the distribution of the training dataset. The optimization process minimizes the expectation of the loss function of the neural network on all samples of the training set by adjusting the model parameters θ. This provides an approximate set of parameter solution for equation (4). In other words, the model output with this set of parameters θ is similar to the ground truth label of the CAN image, and the model is able to classify images.

The network structure of CanNet is evolved from the structure of LeNet5. The number of kernels, kernel size, output and number of weights being trained for both are compared in Table I. In contrast to LeNet5, CanNet applies a smaller convolution kernel because the image information in our attack detection work is concentrated in some local areas, and the smaller convolution kernel is conducive to improving the accuracy of the model. CanNet has removed one max pooling layer and streamlined the structure of the fully connected layer, which can greatly reduce the number of model parameters. At the same time, due to the relatively single image samples in CanNet's application scenarios, this operation will not affect the performance of the model.

Remark 5: From Table I, the number of weight parameters to be trained for CanNet is much fewer than that of LeNet5, and a pooling layer and a fully connected layer are removed, which allows CanNet to reduce the memory occupied while the classification performance is ensured by increasing the width of CanNet.

E. Traceability of Attack Time

The lightweight CAN network trained in the previous subsection is utilized to detect the real CAN traffic data. When the current CAN image is detected as an abnormal CAN image, it is necessary to determine the moment when the attack occurred, the calculation method is depicted below. As can be seen in Fig. 5, the corresponding moments are obtained based on the vertical axis, the sampling time and the number of the current CAN image, which is calculated as follows,

$$T_j = N * \tau * 16 + \tau * (Y_j + 1), \tag{6}$$

where T_j is the time corresponding to the j-th CAN ID, N refers to the number corresponding to the current CAN image,

τ indicates the sampling time, and Y_j represents the value of the vertical axis of the CAN image for the j-th CAN ID.

Remark 6: The mechanisms of converting CAN data into CAN image, as presented in Section IV-C and Appendix B, not only benefit the visualization of the attack, but also enable a quick determination of the time at which the attack happened, as the number of the CAN image and the physical meaning of the vertical axis both facilitate the traceability of the attack time. Moreover, it is essential to specify that the value of is time-varying when sampling non-uniformly, and for this reason the time-varying sample moments need to be recorded to ensure the accuracy of the attack traceability.

V. Experiments

In this section, the dataset, experimental environment and testing are illustrated in detail. Most importantly, the comparative experiments are conducted to illustrate the effectiveness of the designed lightweight detection mechanism against DoS attacks on intelligent vehicles over CAN-bus-based communication.

A. Experiment Dataset

The experimental datasets adopted in this article are the open source health dataset and periodic DoS attack dataset, available at the specific open source website: http://ocslab.hksecurity.net /Datasets/CAN-intrusion-dataset. It should be noted that the experimental CAN-bus traffic data is from a laptop connected to the Raspberry Pi3 via a wireless transmission channel. The Raspberry Pi3 is connected to the CAN bus utilizing a Y-cable plugged into the OBD-II port of Hyundai Motor Company's YF Sonata.

The healthy dataset is the CAN traffic data generated when the YF Sonata vehicle is not subjected to the attack, i.e., the data is collected from the normal driving of the YF Sonata vehicle on the road. The periodic DoS attack dataset is captured from the YF Sonata vehicle, which is subjected to periodic DoS attacks by injecting CAN messages with CAN ID 0X000 every 0.3 milliseconds. This causes the CAN bus to be periodically occupied, so that information from other ECUs is unable to be sent during this period. In a real-world scenario, the straight-line YF Sonata vehicle fails to gear shift periodically, i.e. the accelerator will intermittently fail. For the aperiodic DoS attack dataset, a random CAN message with CAN ID 0X000 is injected into the YF Sonata vehicle and the CAN bus is occupied aperiodically. This prevents the information from other ECUs from being transmitted during the occupancy, thus causing the vehicle to suffer from brake failure, steering wheel failure, or accelerator failure, and consequently making it impossible to stop, change lanes, or shift gears.

In conducting the lightweight CanNet network experiments, the dataset is divided according to the training set, test set and validation set as 8:1:1. The datasets applied for the detection of CAN traffic data contain not only the health dataset, the periodic DoS attack dataset and the aperiodic DoS attack dataset, but also the mixed DoS attack dataset with the mixture of the periodic DoS attack dataset and the aperiodic DoS attack dataset. For the mixed DoS attack dataset, in a real-world scenario, the vehicle is subjected to both periodic and aperiodic DoS attacks in turn, and the scenarios corresponding to both the periodic and aperiodic DoS attack datasets described above will all occur.

TABLE II
Impact of Learning Rate

Learning rate	$\mathbf{10^{-4}}$	$\mathbf{10^{-5}}$	$\mathbf{10^{-6}}$
Iterations	1410	1410	1410
FNR (%)	0.0667	0.399	Stuck in local optimum
ER (%)	0.0222	0.144	(Validation set acc $\leq$ 66.7% during training)

B. Experiment Environment and Testing

The equipment and software parameters employed to implement the lightweight CanNet network to detect DoS attacks and other comparative experiments are described below,

- CPU:Intel(R) Xeon(R) Gold 6126 CPU @ 2.60 GHz,
- RAM: 24 GB,
- GPU: NVIDIA RTX3090,
- Operating System: ubuntu 20.04,
- Pytorch Version: pytorch v1.8.0 and cuda v11.1.

It should be noted that the false negative rate (FNR) and error rate (ER) are measured to evaluate the classification performance. Similar to [37], FNR and ER are defined. FNR is the proportion of undetected attack frames, while ER is the overall misclassification rate of frames. A small FNR is crucial for in-vehicle network security because even a few undetected attacks can lead to momentary malfunctions and pose safety risks. Therefore, in an in-vehicle network, FNR is more important for attack detection and should be minimized.

1) Hyperparameter: The learning rate is a critical hyperparameter in training neural networks that can have a significant impact on model performance. It determines the step size at each iteration of the optimization algorithm and affects how quickly the model converges to a solution. The FNR and ER were evaluated while varying the learning rate from 10^{-6} to 10^{-4}. The batch size was set to 256 and the epoch was set to 15. Table II demonstrates the impact of learning rate. The FNR and ER values presented in the table are the mean values obtained from three different experimental groups. According to our experiments, the optimal learning rate is set as 10^{-4}.

2) Detection performance: To evaluate the detection performance of the proposed CanNet, 15 tests were conducted for each attack dataset, and FNR and ER were measured. The shuffled dataset was split into training and test data, with 70% for training and 30% for testing. Table III shows the results obtained by randomly selecting 10,000 images from the test dataset and testing 15 times, including average, standard deviation, minimum, and maximum values. The proposed CanNet shows an FNR and an ER of less than 0.1% in the periodic DoS dataset, aperiodic DoS attack dataset, and mixed DoS attack dataset.

TABLE III
THE SUMMARIZED TEST RESULTS ON EACH DATASET

Dataset	P-DoS		AP-DoS		Mix-DoS	
(%)	FNR	ER	FNR	ER	FNR	ER
min	0.03	0.14	0.03	0.01	0.23	0.12
std	**0.10**	**0.02**	**0.08**	**0.04**	**0.06**	**0.01**
max	0.41	0.22	0.39	0.20	0.39	0.16
mean	0.29	0.17	0.15	0.08	0.28	0.15

C. Evaluation Function

To standardize the performance of the different methods, a performance evaluation function has been introduced. The performance evaluation function is defined as

$$\mathcal{J} = \mathcal{A} + \mathcal{P} + \mathcal{D} + \breve{A} - \mathcal{P}_a - \mathcal{F}_l - \mathcal{T}_i, \tag{7}$$

where $\mathcal{A}$, $\mathcal{P}$, and $\mathcal{D}$ stand for the accuracy, precision, and detection rate of the network method. Accuracy means the percentage of data including normal and abnormal are correctly classified. Precision represents the evaluation of the percentage of classifiers predicted as correct for a particular category. Detection rate is defined in this article as the average of the proportion of detected abnormal data to the total number of abnormal data for periodic, periodic and mixed DoS attack datasets. $\breve{A}$ represents the area under curve (AUC) of the network method. AUC is specified as the area under the ROC curve, where the ROC curve stands for the receiver operating characteristic curve. The AUC indicates a performance measure of the learner's strengths and weaknesses, and generally classifiers with larger AUC are more effective. $\mathcal{P}_a$ and $\mathcal{F}_l$ are the values calculated by applying the z-score normalization method [41] to the parameters and FLOPs of the network method in $\mathbf{MB}$. Parameters are the number of parameters employed in the network applied for detection, i.e., the parameters learned while training the network. FLOPs refers to floating point operations, that is the number of floating point operations, i.e., the amount of computation in the network. Parameters can be simply comprehended as the number of parameters of the network, and FLOPs can be simply understood as the amount of computation in the forward operation of the network. $\mathcal{T}_i$ is denoted as the values processed by the z-score normalization method to the single image detection time in s. The defined evaluation function (7) indicates that the larger the evaluation value $\mathcal{J}$ the better the performance, in other words, a larger $\mathcal{J}$ indicates a higher detection rate, a shorter time consumption, and a lower memory occupation of the detection mechanism.

D. Comparison of Two CAN Image Coding Schemes

Two schemes for generating CAN images through the CAN traffic data are compared in this subsection, where the scheme 1 utilizes the three hexadecimal values of the CAN ID to generate the three RGB values correspondingly, as specified in section IV-B, and the scheme 2 normalizes the CAN ID to decimal and then applies the heatmap function to generate the three RGB values, as detailed in Appendix B.

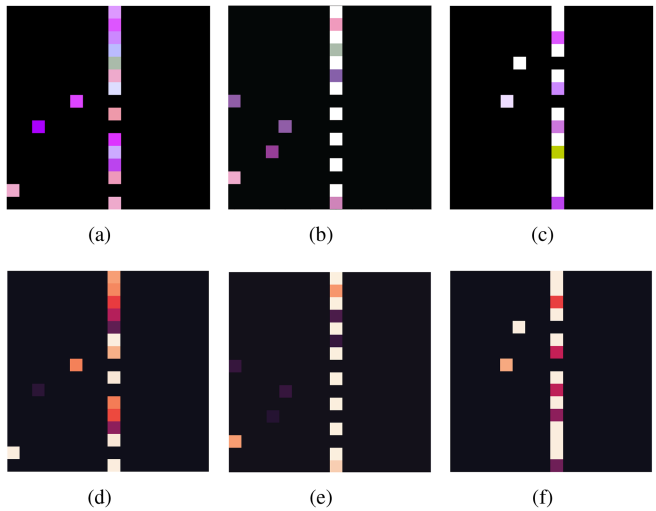

Fig. 8. CAN images. (a)–(c) The CAN images of healthy dataset, periodic and aperiodic DoS attack datasets encoded by scheme 1. (d)–(f) The CAN images of healthy dataset, periodic and aperiodic DoS attack datasets encoded by scheme 2.

TABLE IV
PERFORMANCE COMPARISON OF TWO CAN IMAGE ENCODING SCHEMES UNDER CANNET DETECTION

	Scheme 1(Sec. IV-B)	Scheme 2(Appx. B)
Image Generation Time	**6.90ms**	240ms
Image Detection Time	**12ms**	27ms
Accuracy	**0.9966**	0.9985
Precision	**1.0000**	0.9996
Detection Rate	**0.9977**	0.9973
AUC	**0.999**	0.999

The examples of image generation adopting the two CAN image encoding schemes under health, periodic DoS attacks and aperiodic DoS attacks are displayed in Fig. 8. Meanwhile, the performance of the two CAN image encoding schemes is compared in Table IV in terms of the single image generation and detection time, accuracy, accuracy, detection rate and AUC.

From Table IV, it can be found that the single image generation and detection time of the scheme 1 are superior to those of the scheme 2, which is a great benefit to the real-time characteristic of the whole detection mechanism. In addition, the detection rate and precision of scheme 1 is higher than that of scheme 2, but the accuracy is slightly lower than that of scheme 2, i.e., the false alarm rate of scheme 1 will be a bit higher than that of scheme 2.

TABLE V
PERFORMANCE COMPARISON OF DIFFERENT NETWORK METHODS FOR CAN IMAGE ANOMALY DETECTION

Methods	Parameters	FLOPs	Accuracy			Precision			Detection	AUC	Time
			P-DoS	AP-DoS	Mix-DoS	P-DoS	AP-DoS	Mix-DoS	Rate$_{ave}$		(s)
VGG16 [17]	100.00MB	1.37GB	0.9990	1.0000	0.9998	1.0000	1.0000	0.9998	0.9993	0.999	0.032
Desent121 [18]	7.00MB	233.84MB	0.9990	1.0000	0.9995	1.0000	1.0000	1.0000	0.9992	0.999	0.141
EfficientNet [22]	4.00MB	33.07MB	0.9991	0.9998	1.0000	0.9996	1.0000	1.0000	0.9994	0.999	0.066
ShuffleNetv2 [21]	1.30MB	12.07MB	0.9878	0.9905	0.9956	0.9873	0.9845	0.9976	0.9936	0.998	0.044
MobileNetv2 [20]	0.40MB	5.21MB	0.9813	0.9876	0.9932	0.9787	0.9866	0.9949	0.9891	0.997	0.038
LeNet5 [23]	0.70MB	13.63MB	0.9973	0.9990	0.9995	0.9996	0.9996	0.9998	0.9976	0.999	0.013
CanNet(ours)	**23.70KB**	**1.64MB**	**0.9966**	**1.0000**	**0.9962**	**1.0000**	**1.0000**	**0.9943**	**0.9977**	**0.999**	**0.012**

TABLE VI
PERFORMANCE EVALUATION VALUES FOR DIFFERENT NETWORK METHODS

Evaluation	VGG16 [17]	Desent121 [18]	EfficientNet [22]	ShuffleNetv2 [21]	MobileNetv2 [20]	LeNet5 [23]	CanNet(ours)
$\mathcal{J}$	-1.0665	1.7182	4.0486	4.6797	4.8536	5.4598	**5.5312**

E. Comparison of Different Network Methods

In this subsection, the designed lightweight CanNet mechanism is compared with different classification network methods. The designed mechanism and other different network methods utilize the CAN image encoding scheme 1 for anomaly detection. The performance comparison of each method can be found in Tables V and VI. Compared with the VGG16 and Desent121 methods with high accuracy, precision and detection rate, the designed CanNet is far less than the former in parameters, FLOPs and single image detection time, and can ensure the same performance of AUC. Although the CanNet is lightweight network, it still guarantees the average difference in accuracy, precision and detection rate to be less than 0.33%. It is evident from Table V that the designed method has a superior evaluation value $\mathcal{J}$ than the VGG16 and Desent121 network methods. Moreover, the designed CanNet is preferred to EfficientNet, ShuffleNetv2 and MobileNetv2, which are all lightweight networks, in terms of parameters, FLOPs and single image detection time, while the average difference in their accuracy, precision and detection rate is guaranteed to be less than 0.34%. Also, the AUC of CanNet is improved over that of ShuffleNetv2 and MobileNetv2. These indicate that the application of CanNet for anomaly detection has higher real-time performance and less memory occupation. Table V illustrates that the proposed method has higher evaluation value $\mathcal{J}$ than the EfficientNet, ShuffleNetv2 and MobileNetv2 methods. Last but not least, compared to the main reference LeNet5 network, the designed CanNet optimizes the parameters and FLOPs and reduces the size of memory consumption while inheriting high accuracy, precision, detection rate, AUC and single image detection time. CanNet is far less than the former in parameters, FLOPs and single image detection time, and can ensure the same performance of AUC.

TABLE VII
PERFORMANCE COMPARISON OF DIFFERENT ATTACK DETECTION MECHANISMS

Index	GIDS [36]	INID [37]	CanNet(ours)
Parameters	1.52MB	1.615MB	**23.70KB**
FLOPs	1.59MB	104.1MB	**1.64MB**
Accuracy$_{ave}$	0.9790	0.9963	**0.9976**
Precision$_{ave}$	0.9680	0.9989	**0.9981**
Detection Rate$_{ave}$	0.9960	0.9971	**0.9977**
AUC	0.999	0.999	**0.999**
Time (s)	0.010	0.0167	**0.012**
$\mathcal{J}$	5.4809	5.1423	**5.5312**

F. Comparison of Different Detection Mechanisms

The designed mechanism is compared with other detection mechanisms in this subsection. The performance comparison of different attack detection mechanisms can be found in Table VII. In contrast to the GIDS method [36], the designed CanNet is superior to the former according to $\mathcal{J}$, especially having a large improvement in parameters, accuracy, and precision. Despite the fact that the INID method [37] is higher than the designed CanNet in terms of precision, CanNet is superior to the INID

method in $\mathcal{J}$ and other performance, especially parameters and FLOPs, as evidenced in Table VII. The parameters and FLOPs imply that the designed attack detection scheme has a much lower memory occupation.

VI. Discussions and Conclusion

A. Discussions

The designed lightweight CanNet network applied in detection mechanism has a good performance in terms of real-time, memory usage and performance of detection. However, there are still the following cases which are not considered in our work. These cases are discussed according to the detection mechanism itself.

1) Delays in information transmission caused by environmental factors (e.g., the environments with weak signal reception like overpasses or tunnels), are challenging for detection mechanisms. However, the designed detection method learns the anomalous features of the generated CAN images, for which delay variables can be added to ensure that the moment of attack can be accurately obtained when the signal reception is delayed.
2) With the integration and intelligence of software and hardware, the number of units in intelligent vehicles communicating through the CAN-bus will increase. For this reason, when the multi-digit CAN ID such that the image color coding scheme 1 cannot be used, how to optimize the scheme 2 that generates the corresponding RGB values after normalization, that is, to reduce the CAN image generation time of scheme 2, is a significant problem.
3) For anomaly detection of 0-day DoS attacks that do not utilize the priority arbitration mechanism, the data attributes for CAN image drawing need to be reselected according to the Section IV to ensure the accuracy of detection.
4) How to draw CAN images for data attributes that contain multiple data attributes with the same correlation is a problem that needs to be discussed. A method of data attribute selection for CAN images utilizing weighting factors can be adopted to cope with this problem.
5) Although the DoS attack is implemented via CAN ID=0X0000 in the experimental dataset, the detection method learns the features of the CAN image to identify the attack signal. As can be observed from Fig. 3, the designed attack detection mechanism is still valid for other attacks. Since the different attacks have corresponding anomalous data attributes. With the help of the abnormal data attribute analysis and correlation analysis, it is possible to select the specific data attributes employed to draw the CAN image, consequently, the other types of attacks can be detected by adopting the trained lightweight CanNet network.

Furthermore, the limitations and potential drawbacks of the designed detection mechanism are analyzed as follows. Firstly, the detection mechanism has a fundamental limitation since it is trained on historical data, which may result in a decrease in detection accuracy in scenarios where the data is not covered. Meanwhile, the low amount of data can impact the detection rate, so that the sample generation approach, such as adversarial networks, can be employed to provide more data. Consequently, the detection rate and the performance of the detection mechanism for the unknown attacks can be improved. On the other hand, with the advantage of image recognition, the detection mechanism achieves high precision and real-time detection. However, in recent years, the attacks against images have attracted the attention of academia and industry, as well as have become a potential drawback of the proposed approach. When CAN images generated by the detection mechanism are subjected to such attacks, the false alarm rate is increased. The advanced techniques [42] can be applied to address this challenge to improve the detection rate.

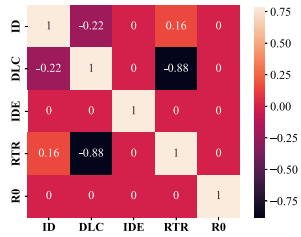

Fig. 9. PC-based heat map of the correlation between CAN ID and CAN-bus data attributes.

B. Conclusion

The DoS attack detection mechanism based on a lightweight CanNet network is designed for the security detection of intelligent vehicles communicating over the CAN bus. The effectiveness and advantages of the proposed detection mechanism, i.e., high detection rate, high real-time and low memory consumption, are verified by experiments with CAN traffic data from an open-source real YF Sonata. In addition, an innovative CAN image encoding method helps to improve the real-time performance of detection mechanism. In future works, the detection performance of the proposed detection mechanism will be enhanced by utilizing data augmentation, the transformation approach of the CAN image color coding scheme 2 will be optimized to reduce the generation time of a single CAN image, and a collaborative detection mechanism will be designed for CAN traffic data containing multiple types of attacks.

Appendix

A. Correlation Analysis of Data via Pearson Correlation

The Pearson Correlation coefficient is employed hereby to measure the linear correlation of the CAN ID with the other data attributes of the CAN data frame and the degree of correlation is visualized in Fig. 9. The Pearson Correlation coefficient for the two variables, M_1 and M_2, is calculated as follows,

$$\mathcal{P}(M_1, M_2) = \frac{Cov(M_1, M_2)}{\sqrt{DM_1 * DM_2}}, \tag{8}$$

where $Cov(M_1, M_2)$ indicates the covariance between M_1 and M_2, DM_1 and DM_2 represent the variances of M_1 and M_2, respectively.

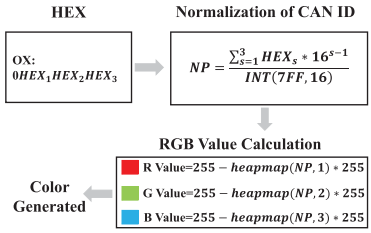

Fig. 10. Diagram of the color generation mechanism based on normalized CAN ID.

According to [40], it is known that the higher the absolute value of the coefficient $|\mathcal{P}(M_1, M_2)|$, the stronger the linear correlation between M_1 and M_2. From Fig. 9, the highest PC correlation with CAN ID is DLC, with a correlation coefficient of $|\mathcal{P}| = 0.22$. However, for intelligent vehicles applied CAN-bus communication, it is debatable whether the relationship between CAN ID and DLC is linear, and it is generally considered that $|\mathcal{P}| \leq 0.3$ is considered that there is no linear correlation between the two attributes. In addition, whether there is a non-linear relationship between the two attributes, the PC coefficient cannot give a reasonable explanation. So the CE, an equivalent form of MI, is adopted to investigate the correlation between CAN ID and other data attributes of CAN data frames in Section IV-B, which is a more convincing correlation analysis.

B. Normalized CAN Image Encoding

As the image encoding method that is insensitive to the number of CAN ID bits, it applies the heatmap function to convert the CAN ID value into a 3-bit RGB value by normalizing the CAN ID. The specific conversion process is shown in Fig. 10.

In contrast to the CAN image encoding method in Fig. 5, the mechanism in Fig. 10 is performed under the normalization of the CAN ID. This technique expands the number of CAN ID bits and has a better performance in converting CAN images when the CAN ID data consists of 12 or more bits. However, it should be pointed out that calling the heatmap function to convert the 1-bit normalized CAN ID into a 3-bit RGB value according to the selected color system is a time-consuming calculation process, which will affect the real-time performance of the detection mechanism.

ACKNOWLEDGMENT

We would like to thank all editors and reviewers who help us improve the article. We are grateful for the efforts from colleagues in Sino-German Center of Intelligent Systems. The authors are also grateful to the Ph.D. candidate of Tongji University, Mr. Zhencun Jiang for his suggestion on our research. We sincerely thank Miss. Yetong Qin and Mr. Haodong Yang for their help in our experimental work.

REFERENCES

[1] S. Tuohy, M. Glavin, C. Hughes, E. Jones, M. Trivedi, and L. Kilmartin, "Intra-vehicle networks: A review," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 534–545, Apr. 2015.

[2] H. Sun, M. Sun, J. Weng, and Z. Liu, "Analysis of ID sequences similarity using DTW in intrusion detection for can bus," *IEEE Trans. Veh. Technol.*, vol. 71, no. 10, pp. 10426–10441, Oct. 2022.

[3] T. Hoppe, S. Kiltz, and J. Dittmann, "Security threats to automotive can networks-practical examples and selected short-term countermeasures," *Rel. Eng. Syst. Saf.*, vol. 96, no. 1, pp. 11–25, 2011.

[4] S. Checkoway et al., "Comprehensive experimental analyses of automotive attack surfaces," in *Proc. USENIX Secur. Symp.*, 2011, pp. 77–92.

[5] Z. Cai, A. Wang, W. Zhang, M. Gruffke, and H. Schweppe, "0-days and mitigations: Roadways to exploit and secure connected BMW cars," in *Proc. Black Hat USA*, 2019, pp. 1–37.

[6] S. R. Pokhrel, "Software defined Internet of Vehicles for automation and orchestration," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3890–3899, Jun. 2021.

[7] C. Kruegel and T. Toth, "Using decision trees to improve signature-based intrusion detection," in *Proc. Lecture Notes Comput. Sci.*, 2003, pp. 173–191.

[8] H. Olufowobi, C. Young, J. Zambreno, and G. Bloom, "SAIDuCANT: Specification-based automotive intrusion detection using controller area network (CAN) timing," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 1484–1494, Feb. 2020.

[9] W. Choi, K. Joo, H. J. Jo, M. C. Park, and D. H. Lee, "VoltageIDS: Low-level communication characteristics for automotive intrusion detection system," *IEEE Trans. Inf. Forensic Secur.*, vol. 13, no. 8, pp. 2114–2129, Aug. 2018.

[10] M. Müter and N. Asaj, "Entropy-based anomaly detection for in-vehicle networks," in *Proc. IEEE Intell. Veh. Symp.*, 2011, pp. 1110–1115.

[11] W. Wu et al., "Sliding window optimized information entropy analysis method for intrusion detection on in-vehicle networks," *IEEE Access*, vol. 6, pp. 45233–45245, 2018.

[12] A. Theissler, "Anomaly detection in recordings from in-vehicle networks," in *Proc. Big Data Appl. Princ. 1st Int. Workshop*, 2014, pp. 11–12.

[13] M.-J. Kang and J.-W. Kang, "A novel intrusion detection method using deep neural network for in-vehicle network security," in *Proc. IEEE 83rd Veh. Technol. Conf.*, 2016, pp. 1–5.

[14] A. Taylor, S. Leblanc, and N. Japkowicz, "Anomaly detection in automobile control network data with long short-term memory networks," in *Proc. IEEE Int. Conf. Data Sci. Adv. Anal.*, 2016, pp. 130–139.

[15] L. Alzubaidi et al., "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, pp. 1–74, 2021.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–14.

[18] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2261–2269.

[19] L. Zhao, L. Wang, Y. Jia, and Y. Cui, "A lightweight deep neural network with higher accuracy," *PLoS One*, vol. 17, no. 8, 2022, Art. no. e0271225.

[20] A. G. Howard et al., "Mobilenets: Efficient convolutional neural networks for mobile vision applications," pp. 1–9, 2017, *arXiv:1704.04861*.

[21] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6848–6856.

[22] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[23] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[24] S. R. Pokhrel and M. B. Hossain, "Data privacy of wireless charging vehicle to grid (V2G) networks with federated learning," *IEEE Trans. Veh. Technol.*, vol. 71, no. 8, pp. 9032–9037, Aug. 2022.

[25] S. Longari, D. H. N. Valcarcel, M. Zago, M. Carminati, and S. Zanero, "CANnolo: An anomaly detection system based on LSTM autoencoders for controller area network," *IEEE Trans. Netw. Serv. Manage.*, vol. 18, no. 2, pp. 1913–1924, Jun. 2021.

[26] W. Lo, H. Alqahtani, K. Thakur, A. Almadhor, S. Chander, and G. Kumar, "A hybrid deep learning based intrusion detection system using spatial-temporal representation of in-vehicle network traffic," *Veh. Commun.*, vol. 35, pp. 100471.1–100471.17, 2022.

[27] Z. Liu et al., "Lightweight trustworthy message exchange in unmanned aerial vehicle networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 2144–2157, Feb. 2023.
[28] K. Wang, A. Zhang, H. Sun, and B. Wang, "Analysis of recent deep-learning-based intrusion detection methods for in-vehicle network," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 1843–1854, Feb. 2023.
[29] A. K. Desta, S. Ohira, I. Arai, and K. Fujikawa, "Rec-CNN: In-vehicle networks intrusion detection using convolutional neural networks trained on recurrence plots," *Veh. Commun.*, vol. 35, pp. 100470.1–100470.13, 2022.
[30] H. Alqahtani and G. Kumar, "A deep learning-based intrusion detection system for in-vehicle networks," *Comput. Elect. Eng.*, vol. 104, pp. 108447.1–108447.16, 2022.
[31] J. Mai et al., "Anomaly detection method for vehicular network based on collaborative deep support vector data description," *Phys. Commun.*, vol. 56, pp. 101940.1–101940.10, 2023.
[32] H. M. Song and H. K. Kim, "Self-supervised anomaly detection for in-vehicle network using noised pseudo normal data," *IEEE Trans. Veh. Technol.*, vol. 70, no. 2, pp. 1098–1108, Feb. 2021.
[33] M. Farsi, K. Ratcliff, and M. Barbosa, "An overview of controller area network," *Comput. Control. Eng.*, vol. 10, no. 3, pp. 113–120, 1999.
[34] R. Bosch, *CAN Specification Version 2.0*. Gerlingen, Germany: Rober Bousch GmbH, Postfa- Ch, 1991.
[35] S. C. HPL, "Introduction to the controller area network (CAN)," *Appl. Rep. SLOA101*, pp. 1–17, 2002.
[36] E. Seo, H. M. Song, and H. K. Kim, "Gids: Gan based intrusion detection system for in-vehicle network," in *Proc. IEEE Annu. Conf. Privacy, Secur. Trust*, 2018, pp. 1–6.
[37] H. M. Song, J. Woo, and H. K. Kim, "In-vehicle network intrusion detection using deep convolutional neural network," *Veh. Commun.*, vol. 21, pp. 100198.1–100198.13, 2020.
[38] J. Ma, "Discovering association with copula entropy," pp. 1–12, 2019, *arXiv:1907.12268*.
[39] J. Ma and Z. Sun, "Mutual information is copula entropy," *Tsinghua Sci. Technol.*, vol. 16, no. 1, pp. 51–54, 2011.
[40] J. Benesty, J. Chen, Y. Huang, and I. Cohen, "Pearson correlation coefficient," in *Noise Reduction in Speech Process*. vol. 2. Berlin, Heidelberg: Springer, 2009, pp. 1–4.
[41] A. Jain, K. Nandakumar, and A. Ross, "Score normalization in multimodal biometric systems," *Pattern Recognit.*, vol. 38, no. 12, pp. 2270–2285, 2005.
[42] J. Wang et al., "Objectformer for image manipulation detection and localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 2364–2373.

Sheng Gao received the B.Sc. degree in automation from Donghua University, Shanghai, China, in 2019. He is currently working toward the Ph.D. degree in control science and engineering with Tongji University, Shanghai. His research interests include optimal control, cyber-physical systems, robot and cyber security.

Linchuan Zhang received the B.S. degree in intelligent science and technology from Qingdao University, Qingdao, China, in 2019 and the M.S. degree in control engineering from Shandong University, Jinan, China, in 2022. He is currently working toward the Ph.D. degree in intelligent science and technology with Tongji University, Shanghai, China. His research interests include robotics, deep learning, SLAM, and automation systems.

Lei He received the B.Sc. degree in automation from Dalian Maritime University, Dalian, China, in 2021. He is currently working toward the M.Sc. degree in control science and engineering with Tongji University, Shanghai, China. His research interests include cyber-physical systems and attack detection.

Xiaoyang Deng received the B.Sc. degree from the School of Electronics and Information Engineering, Tongji University, Shanghai, China, where he is currently working toward the masters degree with the Department of Control Science and Engineering. He focuses on the theory of adversarial machine learning.

Huilin Yin (Member, IEEE) received the Ph.D. degree in control theory and control engineering from Tongji University, Shanghai, China, in 2006 and the M.S. double degree from Tongji University and the Technical University of Munich, Munich, Germany. She is currently the Professor of TUEV SUED Chair with the Electronic and Information Engineering College, Tongji University. Her research interests include environment perception of intelligent vehicles and safety for autonomous driving.

Hao Zhang received the B.Sc. degree in automatic control from the Wuhan University of Technology, Wuhan, China, in 2001 and the Ph.D. degree in control theory and control engineering from the Huazhong University of Science and Technology Wuhan, China, in 2007. She is currently a Professor with the School of Electronics and Information Engineering, Tongji University, Shanghai, China. From December 2011 to December 2013, she was a Postdoctoral Fellow with the City University of Hong Kong, Hong Kong. Her research interests include network-based control systems and multi-agent systems. Dr. Zhang was the recipient of the National Science Fund for Excellent Young Scholars of China and also a Changjiang Young Scholar Distinguished Professor in 2019. She is an Associate Editor for THE IEEE INTELLIGENT TRANSPORTATION SYSTEMS MAGAZINE, *Intelligence & Robotics*, and *Security and Safety*.