Analysis of Recent Deep-Learning-Based Intrusion Detection Methods for In-Vehicle Network

Kai Wang, Aiheng Zhang, Haoran Sun, and Bailing Wang

***Abstract*— The development and popularity of vehicle-to-everything communication have caused more risks to the in-vehicle networks security. As a result, an increasing number of various and effective intrusion detection methods appear to guarantee the security of in-vehicle networks, especially deep-learning-based methods. Nevertheless, the state-of-the-art deep-learning-based intrusion detection methods lack a quantitative and fair horizontal performance comparison analysis. Also, they have no comparative analysis of the detection capability for the unknown attacks as well as on the time and hardware resource consumption of their intelligent intrusion detection models. Therefore, this paper investigates ten representative advanced deep-learning-based intrusion detection methods and illustrates the characteristics and advantages of each method. Moreover, quantitative and fair experiments are set to make horizontal comparison analyses. Also, this study provides some significant suggestions on baseline method selection and valuable guidance, for the direction of future research about lightweight models and the ability to detect unknown attacks.**

***Index Terms*— In-vehicle intrusion detection, deep learning, neural network, vehicular networks.**

I. Introduction

VEHICLE-TO-EVERYTHING (V2X) communication, as well as vehicular networking, has become a popular trend as a key functional component of the emerging intelligent transportation system in the current information society [1]. In recent years, more and more manufacturers have embedded communication protocols in cars, such as Controller Area Network (CAN), to realize various intelligent services and even autonomous driving [2]. However, more network connections provide more opportunities for attackers, and there are a large number of evolving new attacks in cyberspace, resulting in more risks to vehicle security and passenger safety.

CAN bus, although built by Bosch in 1985, is still in fact the de-facto standard for the communications of Electronic Control Units (ECUs) in the in-vehicle networks (IVNs) of modern cars, due to its simplicity, low prices, high efficiency, and stability. In detail, CAN is a message-oriented broadcast-based serial communication protocol, through which the ECUs handle the information transmission concerning all aspects of the car behaviors. Specific CAN messages contain communication priority and various control field information, but neither information about the sender or receiver ECUs nor message authentication mechanisms are embedded, which significantly degrades both security and safety [3].

To solve the security and safety problems, many recent studies about intrusion detection on CAN bus have emerged [4]. Some surveys for IVNs evaluated the comparative performance of previous intrusion detection methods, such as [5], [6] and [7], respectively based on traditional statistics, machine learning, and ensemble learning, but their performances are not good as deep-learning-based methods. For example, the traditional anomaly detection methods based on time statistics have been proved quite fragile [8]. Moreover, message authentication or encryption will lead to inapplicable performance with unacceptable delay in resource-constrained CAN devices [9]. A method based on segmented federated learning [10] is lightweight and can deal with imbalanced data using servers, but the computing mechanism between vehicle and external server communication is complex and has large consumption. Therefore, we only focus on deep-learning-based methods with local vehicle limited resources computing.

Among all, intrusion detection using deep learning technologies may be the most dominant approach for IVNs, for its ability to process an increased amount of data without requiring prior knowledge of domain-specific expertise [11], [12]. There are several surveys on the topic of intrusion detection system (IDS) using deep learning technologies for the vehicular CAN bus [3], [4], [13], [14], [15]. However, they do not have a fair horizontal performance comparison in an identical dataset as well as the same experimental settings, making it very difficult to infer the usage scenarios and performance differences of existing solutions. Also, the baseline methods the experiments conducted in these studies are relatively traditional and backward in performance. Furthermore, they lack an evaluation of the model's ability to detect unknown attacks (e.g., zero-day attacks [16]), which reflects the degree of safety and security that models can provide, and resource consumption to adapt to constrained embedded resource situations.

Hence, our main work and contribution are shown below:

- We select the most representative and advanced methods with the compared performance advantages as the representative of each class of algorithms, carry out horizontal comparison experiments, and get more fair comparative

Manuscript received 18 February 2022; revised 20 July 2022 and 16 September 2022; accepted 1 November 2022. Date of publication 24 November 2022; date of current version 8 February 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62272129 and in part by the Double First-Class Scientific Research Funds of HIT under Grant IDGA10002093. The Associate Editor for this article was S. Garg. *(Corresponding author: Bailing Wang.)*

Kai Wang, Aiheng Zhang, and Bailing Wang are with the School of Computer Science and Technology, Harbin Institute of Technology, Weihai 264200, China (e-mail: dr.wangkai@hit.edu.cn; zahboyos@163.com; wbl@hit.edu.cn).

Haoran Sun is with the Big Data Center, State Grid Corporation of China, Beijing 100031, China (e-mail: ran131110@163.com).

Digital Object Identifier 10.1109/TITS.2022.3222486

results of detection effect. Thus, we can provide baseline selection suggestions for future research.
- We study and evaluate the inference delay and memory consumption of 10 representative algorithms in order to make them adapt to the limited embedded resource environment of IVNs. And, we present a comprehensive analysis of the models' adaptability.
- We evaluate the detection ability of those 10 models toward unknown attacks, and find that the existing algorithms have weak detection ability for unknown attacks, or even do not take unknown attacks into consideration. According to this point, we put forward the future development direction of intrusion detection technology in the in-vehicle network.

The paper is organized as follows. Section II introduces the promising deep-learning-based IDSs in recent years. Section III designs the comparative experimental method. Section IV describes the experimental results and analyses. Section V gives suggestions on future research directions in the vehicular intrusion detection areas, and Section VI concludes this study. For convenient reading, we provide a list of abbreviations in Table VII in the appendix.

II. State-of-the-Art Deep-Learning-Based Intrusion Detection Methods in Recent Years

Since the "Hacking and Countermeasure Research Lab" (HCR Lab, http://ocslab.hksecurity.net) contributes to the data-driven security field by sharing their datasets (e.g., the CAN dataset for intrusion detection [17] and the car-hacking dataset [18]) to the public, the data-driven security equipped with deep learning models for intrusion detection in IVNs, especially in CAN bus, has been exploited extensively in past years.

In this study, we have done a great number of literature researches about IDSs using deep learning technologies. To compare a wider variety of algorithms, we selected one algorithm from each category with comparative advantages as the representative algorithm. The algorithms selected have comprehensive coverage of a wide range of model structures and training or inference types, such as based on convolutional neural network (CNN) structure [18], Long Short Term Memory (LSTM) structure [19], Generative Adversarial Network (GAN) network structure [20] and autoencoder structure [9], [21]. And, they also involve supervised and unsupervised models, binary and multi-classification models. Finally, ten up-to-date models with the best performance in each category are selected and previous study details are shown in Table I.

For instance, the Reduced Inception-ResNet model proposed in [18] is one typical state-of-the-art solution, which is taken as the representative algorithm based on CNN structure, because of its almost perfect detection performance and ability to process large amounts of data. The CANTransfer in [22] is another advanced IDS, which is chosen as the representative algorithm using Transfer Learning based on Convolution LSTM (ConvLSTM) structure. Due to the advantage of spatial-temporal characteristics of the ConvLSTM, CANTransfer can firstly extract knowledge from plenty of normal data as well as previously known but limited intrusion data, and then use the one-shot transfer learning technology to enable the detection ability for new intrusions. The one-shot transfer learning refers to training the model with only hints of a new type of intrusion sample and then migrating the newly learned mapping relation to the original model. Hence, the generalization ability of the model can be enhanced and over-biasing problems due to the imbalanced datasets are more likely to be solved.

The CAN-ADF [23] is an ensemble framework of rule-based and recurrent neural network (RNN)-based models to detect typical in-vehicle intrusions (e.g., hazardous Denial of Service (DoS), fuzzing and replay attacks), which represents the RNN-based structures and multi-classification models. The work in [19] proposes to detect intrusion behaviors via anomaly analysis based on the time series prediction (TSP) method on the data field of every CAN message, using a typical LSTM structure, but get advanced detection performance. The optimized deep denoising autoencoder (O-DAE) proposed in [9] employs an evolutionary-based optimization algorithm namely ecogeography-based optimization (EBO) into each layer of the deep denoising autoencoder for training without premature convergence. O-DAE is selected as the representative algorithm of the autoencoder structure. The work in [20] builds the enhanced generative adversarial network (E-GAN), which is based on the basic principle of the GAN model in [27] but enhances the GAN discriminator by adding a CAN communication matrix. That is also a state-of-the-art solution as a representative algorithm of GAN structure.

The lightweight dynamic autoencoder network (LDAN) in [21] is another representative of the autoencoder structure. Its main characteristic is the lightweight design which reduces the computational cost and model size of the deep learning method. The autoencoder and classifier network in LDAN are constructed with lightweight neural units. Although the LDAN model in the original study was trained based on non-V2X datasets (UNSW-15, KDD99), its characteristic of unsupervised and lightweight is quite suitable for the limited resources of embedded IVNs, which is also the reason we selected it. We also investigated other intrusion detection algorithms not proposed for vehicular networks, which are inapplicable to vehicular network environments or have inferior performance to LDAN, and are excluded in this paper.

In addition, there are some advanced integrated structures, which are difficult to categorize into a single structure but are pioneering. We choose three models combined with typical and basic structures which have high detection performance. For example, a model named HyDL-IDS was proposed in [24], which is a combination of CNN and LSTM structures using the spatial and temporal representation of IVN traffic. Another work in [25] is CANet combined with LSTM and autoencoder structures. Because of the characteristics of these two structures, CANet is an unsupervised learning method and can also capture the temporal dynamics of CAN messages. Rec-CNN model in [26] is a CNN-based structure combined with recurrence plots to generate data, which adds the temporal dependency of a sequence into image data input to the model.

However, the limitation of all the above works is that they lack a horizontal comparison with each other and evaluations

TABLE I
THE REPRESENTATIVE STATE-OF-THE-ART COUNTERMEASURES IN PREVIOUS STUDIES

Research Work	Key Technique	Categories	Evaluation Types	Evaluation Dataset	Baseline Methods	Contribution
Reduced Inception-ResNet [18]	reduced Inception-ResNet model	supervised	binary classification	car-hacking dataset	1) LSTM, 2) ANN, 3) SVM, 4) KNN, 5) NB, 6) DT	1) reduced the unnecessary complexity in the architecture of the Inception-ResNet model to fit the CAN messages, 2) provided a method to preprocess CAN traffic data into a two-dimension pattern.
CANTransfer [22]	convolutional LSTM	supervised	binary classification	CAN dataset for intrusion detection	1) OTIDS, 2) OCSVM, 3) IF, 4) RNN with Heuristics	provided a preliminary consideration for enhancing the generalization ability of the model toward more domains.
CAN-ADF [23]	ensemble of rule-based and RNN-based detection	supervised	multi-classification	CAN dataset for intrusion detection	OTIDS	designed for multiple classifications with a little consideration of unknown attack detection
TSP [19]	LSTM with two kinds of input data format	supervised	binary classification	privately captured CAN traffic from a branded vehicle	None	1) ability of detection on known attacks using the LSTM algorithm without inspecting the protocol semantics, 2) demonstrated the influence of different data field forms on different detection performances.
O-DAE [9]	deep denoising autoencoder network	supervised	binary classification	1) privately captured CAN traffic, 2) simulated dataset from CANoe software, 3) CAN dataset for intrusion detection	1) ANN with two hidden layers, 2) KNN, 3) DT	optimized the deep denoising autoencoder with high performance by employing an ecogeography-based optimization.
LDAN [21]	dynamic autoencoder network built with lightweight neural units and lightweight structures	supervised	binary classification	1) KDD99 dataset, 2) UNSW-NB15 dataset	1) NB, 2) KNN, 3) SVM, 4) DNN, 5) LSTM, 6) DAN	1) designed new lightweight neural units with expansion and compression structure, 2) designed a new method to compute the loss between autoencoder and classifier, 3) ability to efficiently extract high-level representation features.
E-GAN [20]	enhanced GAN discriminator with CAN communication matrix	unsupervised	binary classification	1) privately captured CAN traffic from an online CAN network prototype, 2) privately captured CAN traffic from real vehicle	GAN	1) enhanced the GAN discriminator to be able to detect tampering attacks, 2) introduced the useful CAN communication matrix for a vehicle model.
HyDL-IDS [24]	combination of CNN and LSTM	supervised	binary classification	car-hacking dataset	1) NB, 2) DT, 3) MLP, 4) CNN, 5) LSTM	1) proposed method to characterize the behavior of CAN traffic based on spatial and temporal representation, 2) combined the CNN and LSTM structures for extracting spatial and temporal features.
CANet [25]	combination of LSTM and autoencoder	unsupervised	binary classification	1) privately captured real CAN traffic, 2) synthetic CAN data	1) LSTM, 2) autoencoder	1) proposed a novel network combining LSTM and autoencoder structures, 2) ability to take interdependencies of signals of multiple CAN IDs.
Rec-CNN [26]	ensemble of RPs and CNN	supervised	both binary and multiple classifications	1) privately captured CAN traffic, 2) car-hacking dataset	Reduced inception-ResNet	1) provided methods using recurrence plot to get the temporal dependency of sequence data and convert it to image, 2) designed the network only with two-layered CNN which is quite lightweight.

of resource consumption. These works, except CAN-ADF, lack consideration for unknown attack detection capabilities during the design phase. As a result, our main work is to re-implement these models in a consistent experimental environment and to obtain a fair comparison result on both performance comparison and resource consumption. Also, we evaluate the capabilities of unknown attack detection for each model.

III. Quantitative Experimental Method

To comprehensively explore the comparative performance advantages and resource consumption in a fair experimental environment, all of the ten representative methods above are implemented from scratch in this section. Settings of the fair and quantitative experimental are described below.

A. Physical Configurations

The artificial intelligent server used for all the experiments is equipped with an Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz, a GPU of NVIDIA-SMI 450.80.02, and a 128GB memory. In addition, the operating system is CentOS 7 and the deep learning methods are implemented with TensorFlow 2.0 and NumPy.

B. Real Datasets

To evaluate the practical performance of related intrusion detection methods, the datasets used in experiments should be collected from real cars in real scenarios rather than theoretical hypotheses. Up to now, it is difficult to collect CAN traffic data of IVN. There are two different but all real and widely used open datasets, the CAN dataset for intrusion detection [17] (termed as dataset-1) and the car-hacking dataset [18] (termed as dataset-2), which are constructed by logging the real-time CAN message via the on-board diagnostic (OBD-II) port of two running vehicles (KIA Soul and Hyundai Sonata) with message attacks.

However, the dataset-1 loses data label information. Among the 10 representative methods, there are both supervised models (i.e. the models must be trained with the help of labeled data) and unsupervised models. Also, we need to use the same dataset in our quantitative experiments. Thus, only dataset-2 is adopted as the benchmark dataset to carry out comparison experiments in our study.

Dataset-2 contains normal CAN messages and four typical attack data: DoS attack, fuzzy attack, spoofing the drive gear attack and spoofing the RPM gauze attack. They are stored in five *.csv* files respectively. Each piece of data contains four data features, including timestamp, identifier (ID, in hexadecimal format), data length code (DLC, valued from 0 to 8) and data payload (8 bytes), and the label of a CAN message. It also has a very imbalanced style, where there are 988,987 attack-free CAN messages, and 14,237,978 normal messages (except for the attack-free ones) mixed with 2,331,497 anomaly messages, as shown in Table II.

TABLE II
CAR-HACKING DATASET

Attack Type	# of messages	# of normal messages	# of attack messages
DoS Attack	3,665,771	3,078,250	587,521
Fuzzy Attack	3,838,860	3,347,013	491,847
Spoofing the drive gear	4,443,142	3,845,890	597,252
Spoofing the RPM gauze	4,621,702	3,966,805	654,897
GIDS: Attack-free (normal)	988,987	988,872	0

Two types of Flags in the dataset: T or R.
T represents the attack messages while R represents normal.

TABLE III
ATTACK CLASSIFICATION FOR INTRUSION DETECTION

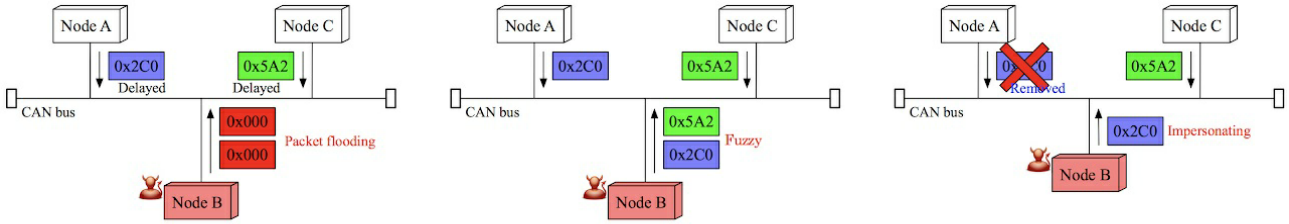
Type	Classification
DoS Attack	Known, used for model training and predicting
Fuzzy Attack	Unknown, only used for the model predicting
Impersonation Attack	Unknown, only used for the model predicting

C. Attack Scenarios

In this study, three types of network-based message attacks, DoS attack, fuzzy attack and impersonation attack, are used for the evaluation of selected detection methods. They can all attack vehicles via network connections and have slightly different attacking behaviors yet huge different levels of detection difficulty. In all the comparative experiments in this paper, as shown in Table III, the DoS attack is used as training data that makes the models learn how to distinguish attacks, which we called known attacks. Fuzzy and impersonation attacks are treated as unknown threats that the models have never met before in the training process (that is, they are never used for training but directly used for inference, which is also known as predicting process, see Table IV). They are used to test whether the current algorithm has the ability to detect not only known attacks but also unknown and sophisticated threats.

As illustrated in Fig. 1 where nodes A and C are legitimate ECUs in the target vehicle while node B is the attacker, the details of these attacks are given as follows.

1) *DoS attack*: Since the CAN bus is a shared communication channel with a broadcast nature, all the ECU nodes (A, B and C) send messages with different priorities determined by embedded CAN IDs (e.g., $0 \times 2C0$ in message from node A and $0 \times 5A2$ from C). Specifically, a lower CAN ID means a higher priority to use the CAN bus for communications. Based on this, a DoS attacker (node B) aims to flood the CAN bus with numerous forged messages with low ID values (even the lowest 0×000 with the highest priority) in every short time interval. Thus, almost all the communication resources are occupied so that messages from other nodes will be delayed or denied from publishing into the same channel. In this case, communication in the involved CAN bus will be created for other ECUs with unacceptable time latency. This may lead to system failure, for example, unable to respond to driver's commands in time and cause traffic accidents.



(a) DoS attack (b) Fuzzy attack (c) Impersonation attack

Fig. 1. Three types of network-based attacks in dataset-1 (From [17]).

TABLE IV
THE PARTITION OF THE CAN DATASET FOR TRAINING AND TESTING

Attack Type	Training Set		Testing Set	
	# of normal message	# of attack message	# of normal message	# of attack message
GIDS: Attack-free (normal)	692,209	0	0	0
DoS Attack	2,154,775	411,264	153,912	29,355
Fuzzy Attack	0	0	167,350	24,592
Spoofing the drive gear	0	0	138,326	29,862
Spoofing the RPM gauze	0	0	114,509	32,744

2) *Fuzzing attack*: In this type of attack scenario, fake messages sent from malicious ECUs intrude into the CAN bus at a slower rate than DoS attack. However, instead of using low ID values like DoS attack, fuzzy attack has random ID values, which avoids the vulnerability of obvious detection characteristics. A fuzzy attack can cause unexpected malfunctions such as an abrupt shift in a vehicle gearbox or an error reminder from the dashboard of a vehicle, thus, driving safety will be seriously impacted. It is very challenging to achieve simple detection based on some unique features, especially for the fuzzy attack that sends messages at the same rate and with the same IDs (e.g., $0\times2C0$ and $0\times5A2$ in Fig. 1(b)) as normal CAN messages. Therefore, more sophisticated countermeasures are required to detect fuzzy attacks.
3) *Impersonation attack*: This attack can realize unauthorized service access by eavesdropping or spoofing legitimate authentication credentials, such as replaying some previously sniffed CAN messages from other ECUs [22], spoofing the drive gear and the RPM gauze [18]. Attackers can manipulate the drive gear to a given constant but valid value (e.g., $0 \times 2C0$ in Fig. 1(c)) by using an impersonation node (e.g., node B) which assumes the identity of the legitimate node A connected in the CAN bus, resulting in legitimate gear abnormal behavior. This type of attack generally intrudes into the CAN bus at a reasonable rate that seems perfectly normal, making it very difficult to detect.

D. Experimental Setup for Model Training and Predicting

To make a horizontal comparison among the recent state-of-the-art intrusion detection methods, the quantitative experiments are conducted, but we retain the characteristics of each model and their personalized optimal hyperparameter settings. The details of our experimental setup are described in this part.

1) *The training parameters*: Because all the detection methods in this comparison experiment are based on deep learning models, we need to search for the optimal hyperparameters for each model in order to achieve the best effect of them. Therefore, we provide a 3-fold cross-validation strategy to get the best hyperparameters for each model and avoid overfitting. The average loss curves of 3-fold cross-validation are shown in Fig. 2. The point with the lowest loss value in the validation curve is the optimal epoch of the corresponding model. For other hyperparameters, we also follow the methods in the original studies to get the best ones. For example, [26] proposed to select the hyperparameters for Rec-CNN models using an algorithm called Hyperband which is also used in our study.
2) *The partition of the dataset*: We designed the experiments not only to evaluate the performance of detecting known attacks but also to evaluate whether these intrusion detection methods have the ability to detect unknown attacks. As shown in Table IV, 70% of normal and DoS attack messages are taken as a training set, and we randomly take consecutive 5% messages from the remaining 30% as a testing set for each test. Especially, CANTransfer is also proposed to use one-shot transfer learning in [22] to detect unknown attacks. Therefore, we evaluate the effect of one-shot transfer learning by adding a hint (16 messages) of fuzzy attack data as the training set for transfer learning, compared with the zero-shot method (original CANTransfer without hints of data for training).
3) *The data pre-processing*: The data pre-processing methods fully follow the original studies of each proposed intrusion detection model. The details of pre-processing

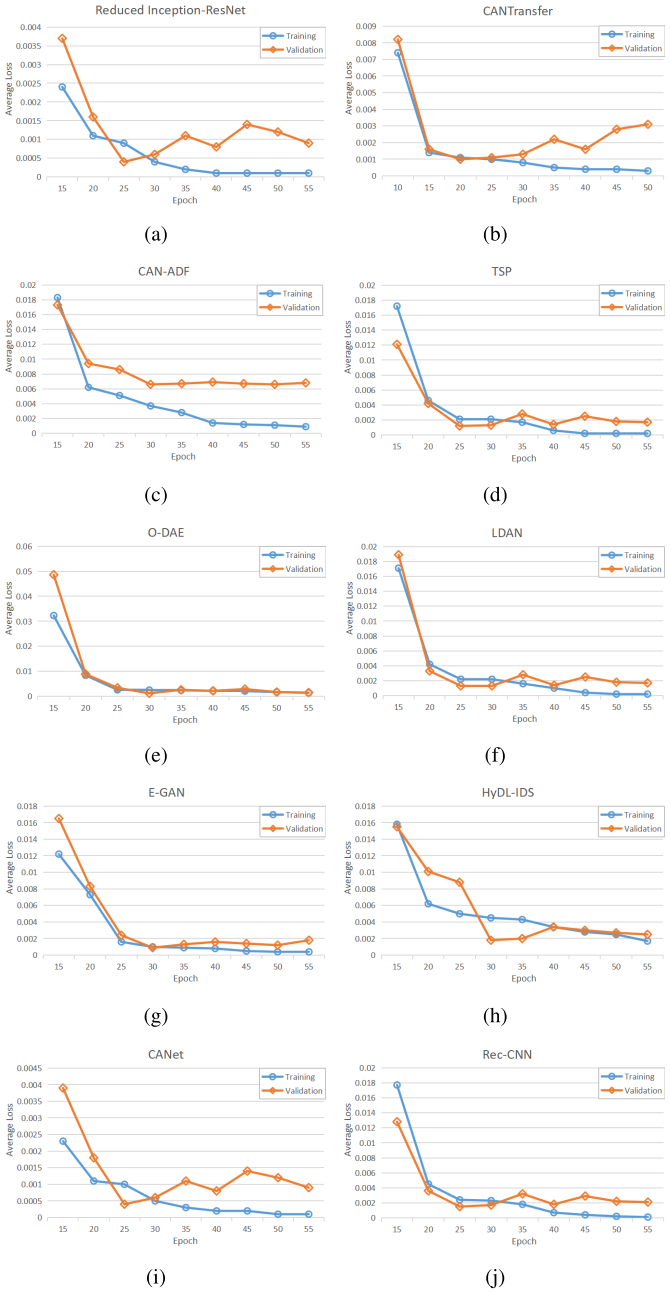


Fig. 2. The average loss value of training and validation process for each model under 3-fold cross-validation strategy.

for each model are shown in Table V. Window size means how many consecutive pieces of data make up a group and form a matrix. 11-feature data is obtained from the original 4-feature data (described in Section III-B, Dataset-2) by subdividing the 64 bits data payload feature into eight features (D1-D8).

E. Comparative Evaluation Metrics

In-vehicle intrusion detection technology is radically designed for practical implementation in automotive industries to ensure the security of IVNs and the safety of passengers. Therefore, the detection performance should be evaluated from all aspects, including the running performance in practical and the detection performance.

TABLE V

THE DATA PRE-PROCESSING FOR INTRUSION DETECTION MODELS

Models	Pre-processing
Reduced Inception-ResNet [18]	A matrix of 29 bits CAN ID with 29 window size, 29*29.
CANTransfer [22]	A matrix of 11-features CAN data with 40 window size using 2D spatial transformation proposed in [22], 4*4*40*11.
CAN-ADF [23]	A matrix of 11-features CAN data with 40 window size and 256 channels, 40*11*256.
TSP [19]	A vector of 64 bits binary CAN data payload.
O-DAE [9]	A vector of the combination of 29 bits CAN ID and 64 bits binary CAN data payload.
LDAN [21]	The combination of 29 bits CAN ID and 64 bits binary CAN data payload is reduced to the dimension of 30 by the PCA algorithm, then forming a matrix with 30 window size, 30*30.
E-GAN [20]	A matrix of 29 bits CAN ID with 29 window size, 29*29.
HyDL-IDS [24]	A matrix of 8-features CAN data payload with 32 window size, 8*32.
CANet [25]	A matrix of 29 bit CAN ID with 29 window size, 29*29.
Rec-CNN [26]	A matrix of 29 bit CAN ID with 29 window size, 29*29. After encoding by recurrence plot, a 128*128 matrix is generated.

1) Evaluation Metrics for Running Performance: The running performance is mainly reflected in time consumption and hardware resource consumption. In our study, the time consumption is assessed in two phases: training and predicting. The average training time per epoch is taken to evaluate whether the detection methods are lightweight enough. The average predicting time per message is crucial for the security of IVNs. For high-speed cars, serious traffic accidents may occur if they are attacked and the intrusion is not detected in time. Memory usage is taken to evaluate the hardware resource consumption. If excessive resources are consumed, it will also affect the normal operation of ECU and cause security risks.

2) Evaluation Metrics for Detection Performance: The experimental results are recorded in confusion matrices, including four possibilities, true positive (TP), true negative (TN), false positive (FP) and false negative (FN), defined as follows.

- TP - attack samples correctly labeled anomalous.
- TN - normal samples correctly labeled normal.
- FP - normal samples incorrectly labeled anomalous.
- FN - attack samples incorrectly labeled normal.

To achieve a comprehensive evaluation of detection methods, we calculated some metrics from confusion matrices. The metrics are accuracy, precision, recall, FPR, FNR and F1-score.

Defined in Equation (1), accuracy is a global evaluation metric that is defined by the ratio of the sum of TP and TN

to all the predicted data.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Precision and recall are two important metrics, reflecting the model's ability to extract and distinguish attack samples. Precision is the ratio of TP to the number of samples detected as attacks, as defined in Equation (2). Recall, also called true positive rate (TPR), is defined by the ratio of TP to the number of actual attack samples, as defined in Equation (3).

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

Besides, false positive rate (FPR) and false negative rate (FNR) are also evaluation metrics, describing the probability of detection errors. FPR is the ratio of FP to the number of actual normal samples, as defined in Equation (4). FNR is the ratio of FN to the number of actual attack samples, as defined in Equation (5), which is complementary to recall.

$$FPR = \frac{FP}{FP + TN} \tag{4}$$

$$FNR = \frac{FN}{TP + FN} \tag{5}$$

However, the testing dataset has a very imbalanced style, which leads to biased results. ROC Area Under the Curve (AUC) is a common metric to deal with data imbalance. ROC curve refers to a graph whose horizontal axis is FPR and vertical axis is TPR. It shows the relationship between FPR and TPR with changing threshold value range. When ROC AUC is close to 1, the model has high performance.

Furthermore, to evaluate the model comprehensively from a positive perspective. We used a metric called F-score, which integrates precision and recall together by adding weight coefficient β, as defined in Equation (6).

$$F_\beta = \left(1 + \beta^2\right) \cdot \frac{Precision \cdot Recall}{\left(\beta^2 \cdot Precision\right) + Recall} \tag{6}$$

In this study, we took $\beta = 1$ and then got the standard F1-score, reflecting the comprehensive evaluation with the balance between recall and precision. Combining Equations (2), (3) and (6), F1-score is calculated as:

$$F_1 = \frac{TP}{TP + \frac{FP+FN}{2}} \tag{7}$$

In summary, we took confusion matrices, accuracy, precision, recall, FPR, F1-score and ROC curves to comprehensively evaluate the detection performance of models. We discarded the FNR metric just because it had been implied in recall ($recall = 1 - FNR$).

IV. Results and Analysis

A. Running Performance Evaluation

Since the running performance of intrusion detection models has an important influence on vehicle safety and security, we analyzed the time and memory consumption of the 10 representative models to evaluate their suitability under resource-limited on-board environment. The model with less time and memory consumption is considered more lightweight, because the time consumption is related to the depth and computational complexity of models, and the memory consumption can reflect the complexity of model parameters and the size of models.

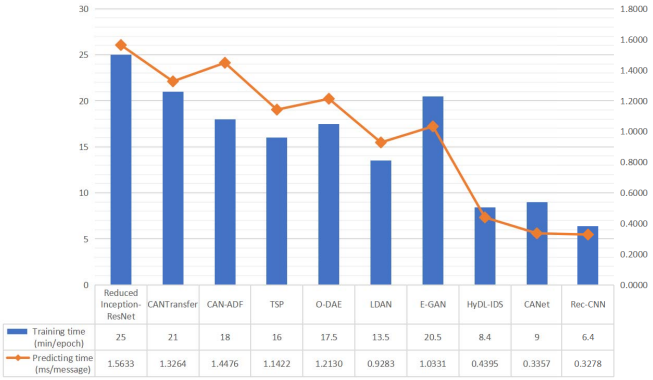


Fig. 3. The time consumption of ten representative models (the bar chart describes the average training time per epoch of models; the line graph describes the average predicting time per message of models).

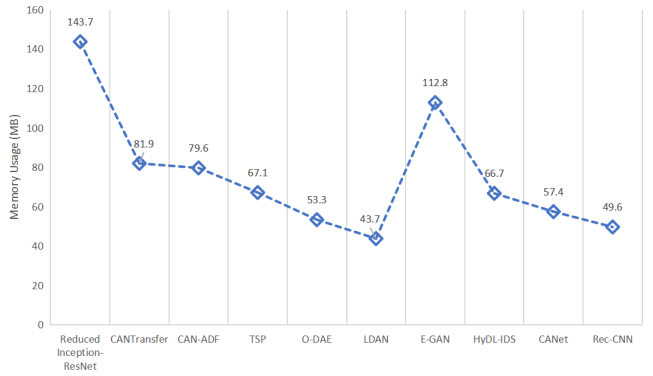


Fig. 4. The memory consumption of ten representative models during inference process.

The experimental results of average training time and predicting time consumption are shown in Fig. 3. It demonstrates that the Rec-CNN model has less time consumption in both the training process and inference process due to the only 5 network layers of the Rec-CNN model [26]. On the contrary, Reduced Inception-ResNet has a relatively deep and complex network structure [18]. It shows a result of the longest training time and predicting time as expected, about 25 min per epoch and 1.5633 ms per message respectively. Although O-DAE and LDAN are both based on autoencoder structure, LDAN uses the dynamic network completely consisting of lightweight units, which makes it less time consumption than O-DAE.

In Fig. 4, it shows the memory usage of 10 representative models when they predict CAN messages. We concern more about the inference process than the training process in our study under the restricted on-board conditions, since the inference process is on ECUs while the training process is offline.

TABLE VI
THE DETECION PERFORMANCE OF TEN METHODS UNDER ALL TYPES OF ATTACKS

		Accuracy	Precision	Recall	FPR	F1-score
Reduced Inception-ResNet	DoS Attack (Known)	0.9993	0.9995	0.9963	0.0001	0.9980
	Fuzzy Attack	0.8730	0	0	0.0002	-
	Gear Spoofing Attack	0.8223	0	0	0.0001	-
	RPM Spoofing Attack	0.7774	0	0	0.0003	-
CANTransfer	DoS Attack (Known)	0.9991	0.9990	0.9951	0.0002	0.9971
	Fuzzy Attack	0.8718	0	0	0.0001	-
	Fuzzy Attack (1-shot)	0.8664	0.9794	0.0309	0.0001	0.0599
	Gear Spoofing Attack	0.8223	0	0	0.0002	-
	RPM Spoofing Attack	0.7774	0	0	0.0003	-
CAN-ADF	DoS Attack (Known)	0.9938	0.9826	0.9785	0.0033	0.9805
	Fuzzy Attack	0.8715	0.0505	0.0002	0.0006	0.0004
	Gear Spoofing Attack	0.8222	0	0	0.0004	-
	RPM Spoofing Attack	0.7769	0.1200	0.0005	0.0012	0.0011
TSP	DoS Attack (Known)	0.9802	0.9100	0.9728	0.0183	0.9403
	Fuzzy Attack	0.8714	0	0	0.0005	-
	Gear Spoofing Attack	0.8221	0	0	0.0005	-
	RPM Spoofing Attack	0.7774	0	0	0.0003	-
O-DAE	DoS Attack (Known)	0.9933	0.9742	0.9843	0.0050	0.9792
	Fuzzy Attack	0.8714	0	0	0.0006	-
	Gear Spoofing Attack	0.8222	0	0	0.0004	-
	RPM Spoofing Attack	0.7774	0	0	0.0003	-
LDAN	DoS Attack (Known)	0.9806	0.9099	0.9756	0.0184	0.9416
	Fuzzy Attack	0.8717	0	0	0.0002	-
	Gear Spoofing Attack	0.8224	0	0	0.0001	-
	RPM Spoofing Attack	0.7775	0	0	0.0002	-
E-GAN	DoS Attack (Known)	0.9806	0.9099	0.9756	0.0184	0.9416
	Fuzzy Attack	0.8717	0	0	0.0002	-
	Gear Spoofing Attack	0.8224	0	0	0.0001	-
	RPM Spoofing Attack	0.7774	0	0	0.0003	-
HyDL-IDS	DoS Attack (Known)	0.9936	0.9819	0.9781	0.0034	0.9800
	Fuzzy Attack	0.8715	0.0612	0.0002	0.0005	0.0005
	Gear Spoofing Attack	0.8221	0	0	0.0001	-
	RPM Spoofing Attack	0.7769	0.1042	0.0005	0.0011	0.0009
CANet	DoS Attack (Known)	0.9993	0.9992	0.9966	0.0014	0.9979
	Fuzzy Attack	0.8717	0	0	0.0002	-
	Gear Spoofing Attack	0.8223	0	0	0.0001	-
	RPM Spoofing Attack	0.7774	0	0	0.0003	-
Rec-CNN	DoS Attack (Known)	0.9803	0.9097	0.9740	0.0185	0.9408
	Fuzzy Attack	0.8714	0	0	0.0006	-
	Gear Spoofing Attack	0.8221	0	0	0.0005	-
	RPM Spoofing Attack	0.7774	0	0	0.0003	-

In addition, we suggest to take predicting time and memory consumption metrics together for a comprehensive evaluation of models' running performance. From this aspect, we think that HyDL-IDS, CANet and Rec-CNN have a higher running performance due to their relatively less predicting time and memory consumption, lower than 0.5 ms per message and 70 MB respectively. On the other hand, LDAN is also considered as having a good running performance for the reason that it has the lowest memory usage, 43.7 MB, and relatively short predicting time, 0.928 ms per message.

B. Detection Performance Evaluation

According to the experimental setup described in Section III, we tried to replicate each of the models described above and got the results shown in Table VI. Since IDS is an active protective measure, it is sensitive to false positive, which could interrupt the normal operation of systems, and needs high detection accuracy, influencing the effectiveness of IDS. The detailed evaluation and specific analysis based on experimental results are illustrated from two aspects presented below.

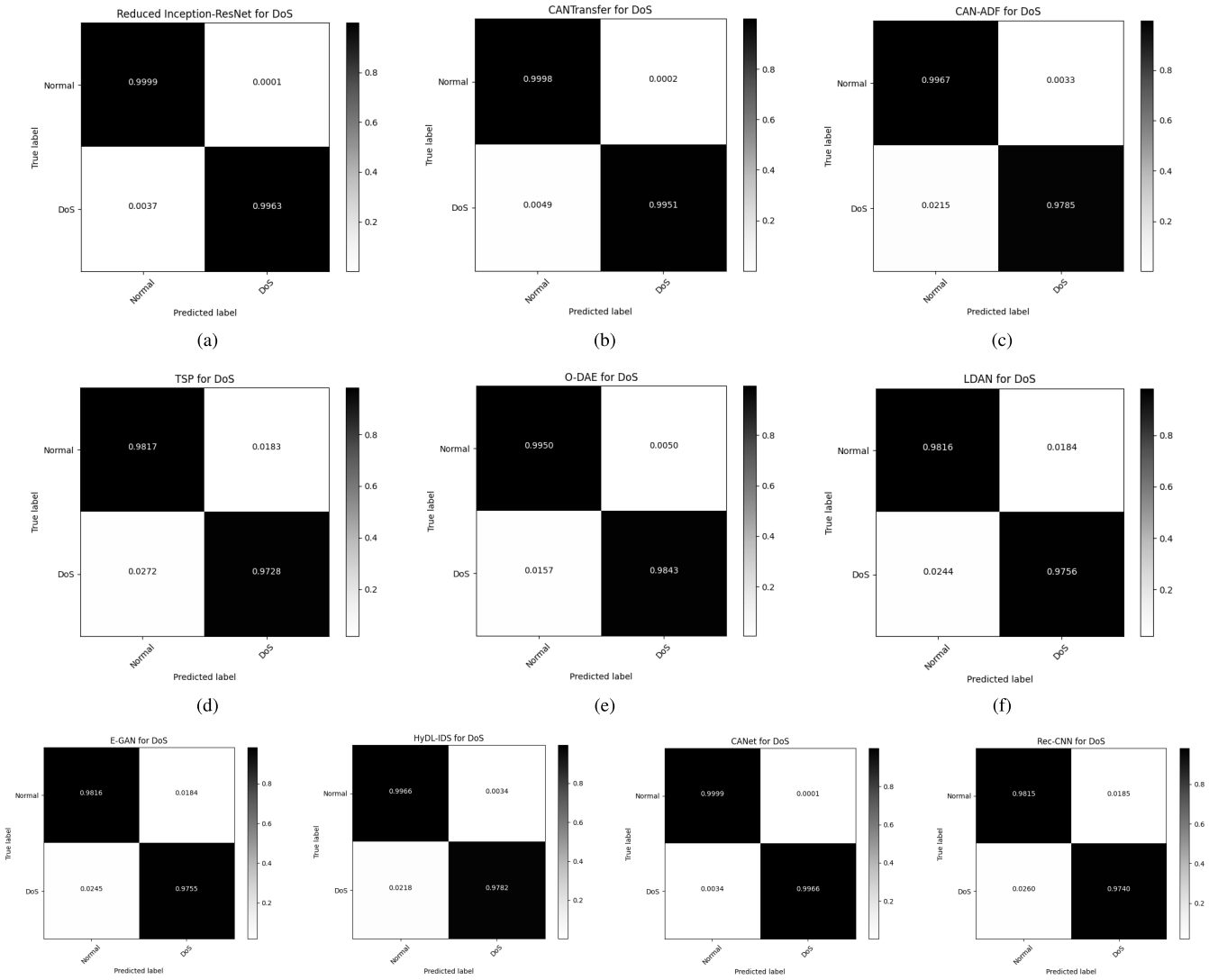


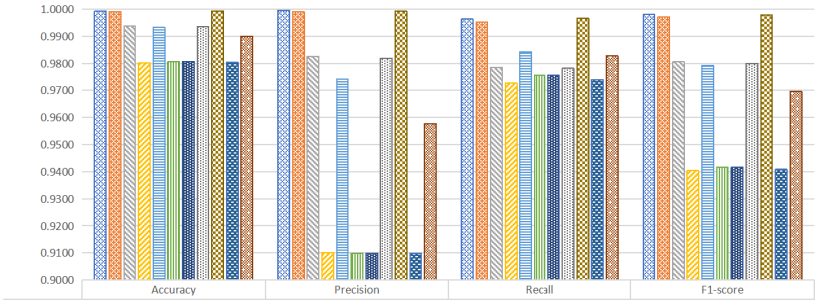
Fig. 5. The confusion matrices of ten models for DoS attack inference.

1) The Ability of Detecting Unknown Attacks: One purpose of our study is to find out whether the representative intrusion detection methods have ability of detecting unknown attacks, which is an inferential capability of detecting simple attacks, such as DoS, to detecting more complex attacks, like fuzzy and impersonation attacks. Thus, only DoS attack samples are adopted as training set, while other types of attacks are regarded as unknown attacks for models.

As shown in Table VI, we can see that the accuracy of each method, regardless the type of attacks, is at an acceptable high level. For example, Reduced Inception-ResNet has 0.9993, 0.8730, 0.8223 and 0.7774 of accuracy for DoS, fuzzy, gear spoofing and RPM spoofing attack respectively, which seems to be an acceptable result. However, combining with other metrics, we find that precision and recall for unknown attacks, including fuzzy, gear spoofing and RPM spoofing attacks, are all 0, illustrating that there is no true positive of any type of unknown attack. The relatively high accuracy in results is mainly caused by imbalanced datasets, so accuracy metric has low reference value here.

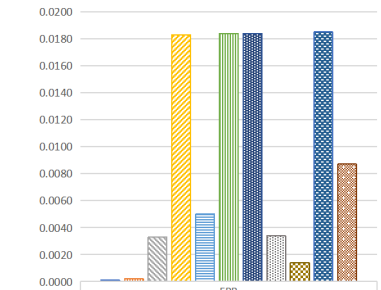
Referring to metrics except for accuracy, we can discover that only CANTransfer, CAN-ADF and HyDL-IDS can barely detect unknown attacks, while others are not. In fact, CANTransfer gets the ability to detect unknown fuzzy attack due to the use of 1-shot transfer learning technology. It improves the performance of detecting fuzzy attack from 0 to 0.9794 of precision, 0.0309 of recall and 0.0599 of F1-score.

It is worth noting that CAN-ADF and HyDL-IDS get a bit of ability to detect fuzzy and RPM spoofing attacks with similar performance. The reason is that CAN-ADF has combined heuristic algorithm with deep learning models, defining a rule for detecting unknown attacks. Moreover, another possible reason is that both CAN-ADF and HyDL-IDS have taken deep learning structures for extracting spatial and temporal features of CAN messages. Furthermore, the result shows that no one model can detect gear spoofing attack as unknown attack, which indicates that gear spoofing attack may have the most complex mechanism among all four types of attacks.



	Accuracy	Precision	Recall	F1-score
Reduced Inception-ResNet	0.9993	0.9995	0.9963	0.9980
CANTransfer	0.9991	0.9990	0.9951	0.9971
CAN-ADF	0.9938	0.9826	0.9785	0.9805
TSP	0.9802	0.9100	0.9728	0.9403
O-DAE	0.9933	0.9742	0.9843	0.9792
LDAN	0.9806	0.9099	0.9756	0.9416
E-GAN	0.9806	0.9099	0.9756	0.9416
HyDL-IDS	0.9936	0.9819	0.9781	0.9800
CANet	0.9993	0.9992	0.9966	0.9979
Rec-CNN	0.9803	0.9097	0.9740	0.9408
Average	0.9900	0.9576	0.9827	0.9697

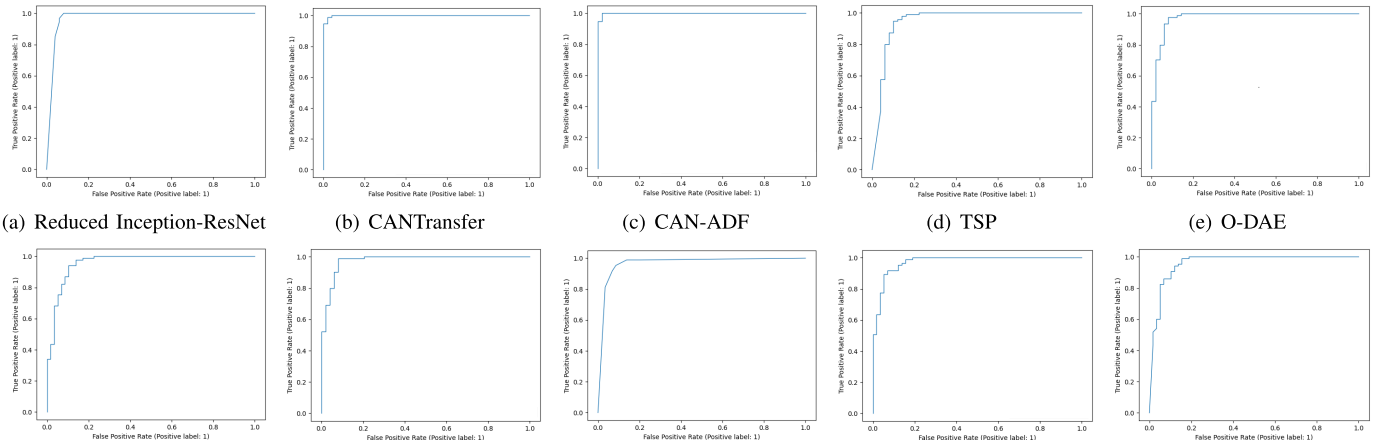
(a) Accuracy, Precision, Recall and F1-score



	FPR
Reduced Inception-ResNet	0.0001
CANTransfer	0.0002
CAN-ADF	0.0033
TSP	0.0183
O-DAE	0.0050
LDAN	0.0184
E-GAN	0.0184
HyDL-IDS	0.0034
CANet	0.0014
Rec-CNN	0.0185
Average	0.0087

(b) FPR

Fig. 6. The detection performance of all methods against known DoS attack (the average value of each metric is shown in the last row of the table and at the far right bar of each group).



(a) Reduced Inception-ResNet (b) CANTransfer (c) CAN-ADF (d) TSP (e) O-DAE

(f) LDAN (g) E-GAN (h) HyDL-IDS (i) CANet (j) Rec-CNN

Fig. 7. The ROC curves of ten models for DoS attack inference.

2) The Comparison of Detecting Known DoS Attacks: Based on our quantitative experiment, the results of known DoS attack is used to evaluate the detection performance of all models and make a horizontal comparison between each model. Normalized confusion matrices, as shown in Fig. 5, is taken and shows that all models mentioned in our study can predict the labels of normal message and DoS attacks correctly. In addition, accuracy, precision, recall, FPR and F1-score are calculated and shown in Fig. 6, where the results can be compared intuitively. As the state-of-the-art intrusion detection technologies, all methods described in this study have acceptable performance, with high accuracy (all over 0.98), high precision (all over 0.90), high recall (all over 0.97), high F1-score (all over 0.94) and low FPR (all below 0.02).

For a further analysis and comparison, ROC curves are taken to show more details in Fig. 7. The results demonstrate that all models have satisfactory detection performance dealing with imbalanced datasets. It also indicates that CANTransfer and CAN-ADF perform better under imbalanced datasets. The reason is that CANTransfer utilizes transfer learning and CAN-ADF utilizes rule-based framework, both of which are beneficial to avoid overfitting training and enhance generalization ability, hence the imbalanced datasets will have less influence on these two models.

Moreover, we classify these ten models into three levels according to the detection performance under DoS attack. Among all, Reduce Inception-ResNet, CANet and CANTransfer are the top 3 models with F1-score over 0.99 and FPR no more than 0.002. Therefore, these three models belong to the first level for their excellent detection performance. As the result of having detection performance scores around average values, CAN-ADF, O-DAE and HyDL-IDS are classified into the second level. They also have considerable satisfactory detection performance but are not so perfect as the top three with scores difference from them about 0.02. The last level includes LDAN, E-GAN, Rec-CNN and TSP, whose detection

performance on DoS attacks has a relatively big gap between them and the first two levels.

V. Discussion and Future Work

Having combined running performance and detection performance, we re-evaluated these representative intrusion detection methods based on the experimental results. Although Reduced Inception-ResNet has the best detection performance and Rec-CNN has the best running performance, they lack good comprehensive performance to be selected as a comparative baseline for in-vehicle environment. By comprehensive analysis, CANet is the best to be selected as comparative baseline, which has high-level running performance as well as the first-level detection performance. In addition, considering the ability of detecting unknown attacks, HyDL-IDS may be also a good alternative baseline as the result of its high-level running and detection performance and its existing ability for unknown attacks detecting. From this aspect, we can also find that considering both spatial and temporal features of attack data may be helpful to learn more about internal data characteristics and improve the generalization ability of models, and then models can obtain some ability to detect unknown attacks.

According to the results of our experiments, we find that the existing intrusion detection methods based on deep learning have made the detection performance on CAN traffic reach a very high degree. For example, Reduced Inception-ResNet, CANTransfer and CANet all achieve close to 100% detection scores. However, the challenges of deep-learning-based IDSs are the ability to detect unknown and sophisticated attacks and the ability to reduce their time and resource consumption for embedded systems. These factors will determine whether a model is suitable to be applied in the resource-constrained in-vehicle network environment in reality and whether the model can ensure the safety and security of the CAN bus.

Therefore, the enhancement of models' detection ability for unknown attacks, the running speed, as well as the reduction of resource consumption are suggested for future research in the field of in-vehicle intrusion detection technology based on deep learning. Also, we suggest designing feature extraction architecture with shallow layers based on both time and space into intrusion detection models and using transfer learning to improve the generalization ability of models. In this way, the models can be lightweight enough for an in-vehicle environment and may get the ability to detect unknown attacks.

VI. Conclusion

With the development and popularity of V2X communication, the connectivity between vehicles and powerful networks become much more than ever, causing more chances for attackers to take over a car by CAN. Therefore, many intrusion detection methods are designed to ensure the security of IVNs, especially deep-learning-based methods, having much more capabilities and better performance than traditional algorithms. However, studies about deep-learning-based intrusion detection methods in a quantitative and fair horizontal performance comparison analysis is insufficient. To get valuable conclusions about the selection of proper baseline method under embedded in-vehicle environment, this paper focuses on investigating and discussing ten representative state-of-the-art deep-learning-based IDSs: Reduced Inception-ResNet, CANTransfer, CAN-ADF, TSP, O-DAE, LDAN, E-GAN, HyDL-IDS, CANet and Rec-CNN. By setting quantitative experiment, this paper discusses the ability to detect unknown attacks, the running performance and detection performance of these intrusion detection methods. The result shows that CANet and HyDL-IDS may be suitable to be selected as baseline methods for their great comprehensive performance. Also, we provide significant suggestion and valuable guidance for the development direction of in-vehicle intrusion detection method about reducing time delay and resource consumption and improving the ability of detecting unknown attacks.

TABLE VII
The Abbreviation

Abbreviation	Full Name
ANN	Artificial Neural Network
AUC	Area Under the Curve
CAN	Controller Area Network
CNN	Convolutional Neural Network
ConvLSTM	Convolution Long Short Term Memory
DAN	Dynamic Autoencoder Network
DNN	Deep Neural Network
DoS	Denial of Service
DT	Decision Trees
EBO	Ecogeography-based Optimization
ECU	Electronic Control Unit
E-GAN	Enhanced Generative Adversarial Network
FN	False Negative
FNR	False Negative Rate
FP	False Positive
FPR	False Positive Rate
GAN	Generative Adversarial Network
IDS	Intrusion Detection System
IVN	In-Vehicle Network
KNN	K-Nearest Neighbors
LDAN	Lightweight Dynamic Autoencoder Network
LSTM	Long Short Term Memory
MLP	Multi-Layer Perception
NB	Naive Bayes
OCSVM	One-Class Support Vector Machine
O-DAE	Optimized Deep Denoising Autoencoder
OTIDS	Offset Ratio and Time Interval based Intrusion Detection System
RNN	Recurrent Neural Network
SVM	Support Vector Machine
TN	True Negative
TNR	True Negative Rate
TP	True Positive
TPR	True Positive Rate
TSP	Time Series Prediction
V2X	Vehicle-to-Everything

Appendix

See Table VII.

References

[1] H. H. Jeong, Y. C. Shen, J. P. Jeong, and T. T. Oh, "A comprehensive survey on vehicular networking for safe and efficient driving in smart transportation: A focus on systems, protocols, and applications," *Veh. Commun.*, vol. 31, Oct. 2021, Art. no. 100349.

[2] J. P. Jeong et al., "A comprehensive survey on vehicular networks for smart roads: A focus on IP-based approaches," *Veh. Commun.*, vol. 29, Jan. 2021, Art. no. 100334.
[3] E. Aliwa, O. Rana, C. Perera, and P. Burnap, "Cyberattacks and countermeasures for in-vehicle networks," *ACM Comput. Surv.*, vol. 54, no. 1, pp. 1–37, Apr. 2021.
[4] G. Loukas, E. Karapistoli, E. Panaousis, P. Sarigiannidis, A. Bezemskij, and T. Vuong, "A taxonomy and survey of cyber-physical intrusion detection approaches for vehicles," *Ad Hoc Netw.*, vol. 84, pp. 124–147, Oct. 2019.
[5] H. Ji, Y. Wang, H. Qin, Y. Wang, and H. Li, "Comparative performance evaluation of intrusion detection methods for in-vehicle networks," *IEEE Access*, vol. 6, pp. 37523–37532, 2018.
[6] T. Moulahi, S. Zidi, A. Alabdulatif, and M. Atiquzzaman, "Comparative performance evaluation of intrusion detection based on machine learning in in-vehicle controller area network bus," *IEEE Access*, vol. 9, pp. 99595–99605, 2021.
[7] D. Swessi and H. Idoudi, "Comparative study of ensemble learning techniques for fuzzy attack detection in in-vehicle networks," in *Advanced Information Networking and Applications*, L. Barolli, F. Hussain, and T. Enokido, Eds. Cham, Switzerland: Springer, 2022, pp. 598–610.
[8] B. Groza and P.-S. Murvay, "Efficient intrusion detection with Bloom filtering in controller area networks," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 4, pp. 1037–1051, Apr. 2019.
[9] Y. Lin, C. Chen, F. Xiao, O. Avatefipour, K. Alsubhi, and A. Yunianta, "An evolutionary deep learning anomaly detection framework for in-vehicle networks—CAN bus," *IEEE Trans. Ind. Appl.*, early access, Jul. 17, 2020, doi: 10.1109/TIA.2020.3009906.
[10] Y. Sun, H. Ochiai, and H. Esaki, "Intrusion detection with segmented federated learning for large-scale multiple LANs," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2020, pp. 1–8.
[11] A. Jolfaei, N. Kumar, M. Chen, and K. Kant, "Guest editorial introduction to the special issue on deep learning models for safe and secure intelligent transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4224–4229, Jul. 2021.
[12] A. Mchergui, T. Moulahi, and S. Zeadally, "Survey on artificial intelligence (AI) techniques for vehicular ad-hoc networks (VANETs)," *Veh. Commun.*, vol. 34, Apr. 2021, Art. no. 100403.
[13] K. Kim, J. S. Kim, S. Jeong, J.-H. Park, and H. K. Kim, "Cybersecurity for autonomous vehicles: Review of attacks and defense," *Comput. Secur.*, vol. 103, Aug. 2021, Art. no. 102150.
[14] W. Wu et al., "A survey of intrusion detection for in-vehicle networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 919–933, Mar. 2020.
[15] S. Sharma and A. Kaul, "A survey on intrusion detection systems and honeypot based proactive security mechanisms in VANETs and VANET cloud," *Veh. Commun.*, vol. 12, pp. 138–164, Apr. 2018.
[16] M. Keramati, "An attack graph based procedure for risk estimation of zero-day attacks," in *Proc. 8th Int. Symp. Telecommun. (IST)*, Sep. 2016, pp. 723–728.
[17] H. Lee, S. H. Jeong, and H. K. Kim, "OTIDS: A novel intrusion detection system for in-vehicle network by using remote frame," in *Proc. 15th Annu. Conf. Privacy, Secur. Trust (PST)*, Aug. 2017, pp. 57–66.
[18] H. M. Song, J. Woo, and H. K. Kim, "In-vehicle network intrusion detection using deep convolutional neural network," *Veh. Commun.*, vol. 21, pp. 1–13, Jan. 2020.
[19] H. Qin, M. Yan, and H. Ji, "Application of controller area network (CAN) bus anomaly detection based on time series prediction," *Veh. Commun.*, vol. 27, Jan. 2021, Art. no. 100291.
[20] G. Xie, L. T. Yang, Y. Yang, H. Luo, R. Li, and M. Alazab, "Threat analysis for automotive can networks: A GAN model-based intrusion detection technique," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4467–4477, Jul. 2021.
[21] R. Zhao et al., "An efficient intrusion detection method based on dynamic autoencoder," *IEEE Wireless Commun. Lett.*, vol. 10, no. 8, pp. 1707–1711, Aug. 2021.
[22] S. Tariq, S. Lee, and S. S. Woo, "CANTransfer: Transfer learning based intrusion detection on a controller area network using convolutional LSTM network," in *Proc. 35th Annu. ACM Symp. Appl. Comput.*, Mar. 2020, pp. 1048–1055.
[23] S. Tariq, S. Lee, H. K. Kim, and S. S. Woo, "CAN-ADF: The controller area network attack detection framework," *Comput. Secur.*, vol. 94, Jul. 2020, Art. no. 101857.
[24] 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, Jun. 2022, Art. no. 100471. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214209622000183
[25] M. Hanselmann, T. Strauss, K. Dormann, and H. Ulmer, "CANet: An unsupervised intrusion detection system for high dimensional CAN bus data," *IEEE Access*, vol. 8, pp. 58194–58205, 2020.
[26] 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, Jun. 2022, Art. no. 100470. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214209622000171
[27] Y. Yang, G. Xie, J. Wang, J. Zhou, Z. Xia, and R. Li, "Intrusion detection for in-vehicle network by using single GAN in connected vehicles," *J. Circuits, Syst. Comput.*, vol. 30, no. 1, Jan. 2021, Art. no. 2150007.

Kai Wang received the B.S. and Ph.D. degrees from Beijing Jiaotong University. He is currently an Associate Professor with the Faculty of Computing, Harbin Institute of Technology (HIT), China. Before joining HIT, he was a Post-Doctoral Researcher in computer science and technology with Tsinghua University. He has published more than 40 papers in prestigious international journals and conferences, including *IEEE Network*, IEEE SYSTEMS JOURNAL, IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, and *ACM Transactions on Internet Technology*. His current research interests include lightweight and intelligent security technologies, such as deep learning applications on industrial control network or in-vehicle network intrusion detection. He is a Senior Member of the China Computer Federation (CCF).

Aiheng Zhang received the B.S. degree in computer science and technology from the Beijing University of Technology, Beijing, China. She is currently pursuing the master's degree in computer technology with the Harbin Institute of Technology (HIT), China. Her research interests include intelligent and lightweight in-vehicle intrusion detection models.

Haoran Sun received the B.S. degree in computer science and technology from Jilin University, Changchun, China, and the master's degree in computer technology from the Harbin Institute of Technology (HIT), China. He is currently an Engineer with the Big Data Center, State Grid Corporation of China. His research interests include transfer learning and data-sample-generating methods for in-vehicle intrusion detection.

Bailing Wang received the Ph.D. degree from the School of Computer Science and Technology, Harbin Institute of Technology (HIT), China, in 2006. He is currently a Professor with the Faculty of Computing, HIT. He has published more than 80 papers in prestigious international journals and conferences, and has been selected for the China national talent plan. His research interests include information content security, industrial control network security, and V2X security.