EasyChair Preprint

№ 10242

A Light-Weighted Machine Learning Based ECU Identification for Automative CAN Security

Jini Li, Man Zhang and Yu Lai

May 22, 2023

A light-weighted machine learning based ECU identification for automative CAN security

Jini Li
School of Electronics and Communication Engineering
Guangzhou University
Guangzhou, China
Email:jini556@163.com

Man Zhang *
School of Electronics and Communication Engineering
Guangzhou University
Guangzhou, China
Email:manzhang401@gzhu.edu.cn

Yu Lai
School of Electronics and Communication Engineering
Guangzhou University
Guangzhou, China
Email:flame@e.gzhu.edu.cn

***Abstract*—The rise of artificial intelligence brings information security challenges for intelligent connected vehicles. Securing the CAN is crucial to ensuring the overall security of the in-vehicle network. Traditional cryptography technology faces challenges of low computational efficiency and excessive data load when identifying ECU. This paper proposes a light-weighted machine learning based identification algorithm that leverages the physical characteristics of ECU. By analyzing the CAN voltage signals in the time and frequency domains, reducing the data load and choosing a suitable classification model, this method achieves high accuracy, high efficiency and low load for safety identification in-vehicle networks. The experimental results on the data sets of both actual vehicles and CAN bus prototypes have verified the rationality and feasibility of the method.**

***Index Terms*—CAN; ECU identification; physical characteristics; light-weighted; machine learning**

I. Introduction

Intelligent Connected Vehicle(ICVs) are growing rapidly and also suffer from security threats[1]. The Controller Area Network (CAN), which connects various sensors and Electronic Control Units(ECU), is the de facto communication standard for in-vehicle network[2][3]. Since the CAN protocol is designed under the resource constrains in-vehicle environment, it does not support message origin authentication. So it lacks encryption and authentication protocols[4][5][6].

Traditional cryptographic authentication schemes based on Message Authentication Code (MAC)[7] are no longer applicable to CAN frame. Different ECUs show inherent variations in signaling behavior due to hardware and manufacturing inconsistencies. Therefore, the unique physical characteristics based on clock offset could be used as ECU fingerprinting. Based on this, researchers analysed the physical characteristics of the CAN signals transmitted from the ECU as its authentication ID[8]. Since the physical characteristics extraction based on clock offset is easily falsified[9][10], researchers collected different data segments in CAN message frames as training samples and build machine learning models to classify and identify ECUs for intrusion detection. Most of these methods do not take sufficient account of lightweight requirements, or do not select a more suitable machine learning algorithm given the limited resources available.

*Corresponding author.

To implement an ECU fingerprinting technology in CAN networks using low-cost and resource-limited hardware, we propose a light-weighted machine learning identification method, which provides authentication capability for CAN networks. Firstly, dozens of features are extracted from voltage data of CAN signals in the time and frequency domains. The data preprocessing and feature selection are conducted on those features. Finally, two kinds of machine learning classification models are trained and compared based on experiment results. By means of proper feature reduction and a suitable classification model, the proposed method achieves high efficiency and low data load for ECU identification in CAN. In brief, the contributions of our work can be summarized as follows:

- The data load of the identification algorithm is greatly reduced. The amount of training data is reduced by only adopting the dominant segment. Furthermore, the original dozens of features extracted from the dominant segment are filtrated to a few by recursive feature elimination.
- Aiming to the light-weighted requirement, the more suitable machine learning model is chosen in steps. Two kinds of models are initially adopted considering the limited resource constraints. Then experimental comparisons are conducted between them on classification precision and performance.
- Comprehensive experimental validation is performed on a public dataset from real vehicles and a CAN bus Prototype respectively. Both experimental scenarios show the rationality and feasibility of our method.

The rest of the paper is organized as follows. Section II contains the basic knowledge and related work. Section III describes the fingerprint identification method in detail. The experimental dataset and results are presented in Section IV. Finally, Section V concludes the paper.

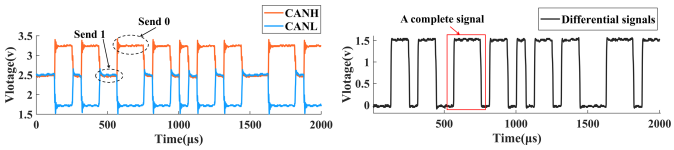
II. Background And Related Works

In this section we introduce some brief background on Controller Area Network and the related works.

A. Background on Controller Area Network

The CAN bus enables electronic control units to exchange data at rates of up to 1 Mbps over twisted pair cables[3]. The CAN protocol uses differential signals of high-level CAN-H and low-level CAN-L to make CAN noise-resistant and fault tolerant. Fig.1 shows the CAN bus signal logic. The voltage levels of the two differential lines (CAN-H and CAN-L) are approximately 2.5 V. When transmitting dominant bits, the voltage level on CAN-L is about 1.5V and the voltage level on CAN-H is about 3.5V. The unique physical characteristics of CAN signals arise from minor variations in voltage levels in the line, which are caused by small, uncontrollable variations in the ECU manufacturing process, as well as by the influence of the transmission medium itself.

Fig.2 represents the structure of a standard frame of CAN data.The CAN data frames are available in both standard and extended frame formats, and the ECU can determine if it is interested in the received message based on the CANID. In the arbitration phase, the ECU with the smaller ID value gets the priority to send the message. During the voltage signal collection phase, we will remove the SOF, ACK and EOF fields that may be affected by multiple ECUs and keep the ID field to mark the frame.



(a) Voltage signal of CANH and CANL (b) Voltage differential signal

Fig. 1. Part of data frame signal at the physical layer.

SOF	ID	RTR	IDE	R0	DLC	Data	CRC	ACK	EOF
1bit	11bit	1bit	1bit	1bit	4bit	0to64bit	16bit	2bit	7bit

Collected voltage signal

SOF: Start of Frame
ID: Match with an ECU
RTR: Remote Transmission Request
IDE: Identifer Extension Bit
R0: Reserved bit 0
DLC: Data Length Code
CRC: Cyclic Redundancy Check
ACK: Acknowledgement Character
EOF: End of Frame

Fig. 2. Data frame of CAN.

B. Related Works

ECU identification based on physical layer voltage signals is hot research in recent years. Murvay was one of the first[11], who used voltage samples for CAN intrusion detection and sender identification. Choi et al[12] extracted the physical features of the differential signal from the extended frames, and they also analysed all signals after the ID domain in a frame of messages[13]. The voltage samples in Scission[14] were divided into three groups, and time domain statistical characteristics were used as features. The subsequent work of Scission[15] used a reduced amount of acquired signals and a reduced sampling rate to achieve light-weighted. And [19] extracted transient and steady-state parameters of the electrical signal. In [16][17][20], several feature values were extracted and inputted a convolutional neural networks for training. Viden[18] used voltage profiles of CAN frames and ACK voltage thresholds to identify ECUs and detect hostile ECUs.In addition, [21] used a reinforcement learning approach. And in [23], the authors calculated the voltage difference or similarity between ECUs by studying the Marxist distance. The authors of [24] extracted the voltage characteristics between two points on a CAN communication channel to train a random forest model. [25] separated singular pulses from the data to perform ECU classification using three different machine learning models.

Among the above research works different aspects have been analyzed such as extraction methods of physical features and classification models. However, considering the requirement for light-weighted, existing methods neither sufficiently reduce the data load for training and classifying, nor adequately examine the classification model, and all methods have been validated on only few real vehicles or CAN bus prototype. Recently, [22] has provided a very informative dataset of ten vehicles which we have been experimenting on.

III. Our Method

In this section, an ECU identification method based on the physical features of CAN signals is present in details. The specific steps of the identification method are shown in Fig.3.

A. Electrical CAN Signal Sampling and Processing

In step 1, the CAN voltage signals from different ECUs are sampled. Since the CAN has CAN-H and CAN-L channels, we have to collect the electrical signals on these two channels. As explained in Section II, the CAN protocol uses differential signals to ensure noise immunity and fault tolerance. To avoid being affected by electromagnetic interference or other variations, after sampling the output voltages from CAN-H and CAN-L on the CAN bus, we need to subtract the CAN-H and CAN-L signals to obtain the differential signal during step 2. In addition, some invalid data and noise may be collected and need to be filtered out. When identifying the physical characteristics of the electrical signal, we focus on the dominant state part and the part of the signal in which the state is changed from recessive to dominant or vice versa, and these parts of the signal are referred to as the rising edge and falling edge parts. This helps to improve both robustness and accuracy of the system.

As shown in Fig.4, we divide a complete signal into three parts: rising edge, falling edge and dominant state. For our purposes, we sample the mean value of the dominant state part in the middle of the signal as the threshold. The threshold is used as a segmentation point to split the signal. It is possible to make important physical characteristics more observable by considering the individual segments.

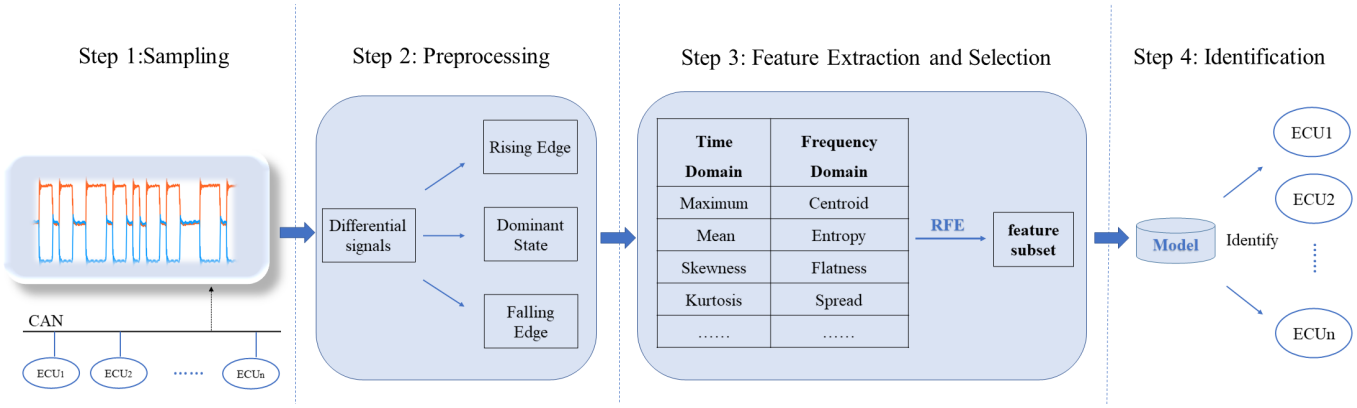


Fig. 3. Overview of the ECU identification.

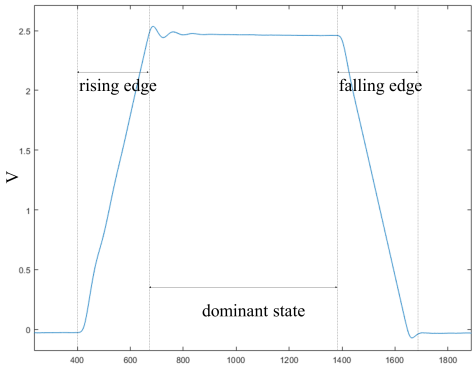


Fig. 4. Three segments of a complete: rising edge, dominant state, and falling edge.

TABLE I
TIME DOMAIN FEATURES

Feature	Description
Maximum	$max = \max(x(i))_{i=1...N}$
Minimum	$min = \min(x(i))_{i=1...N}$
Mean	$\mu = \frac{1}{N}\sum_{i=1}^{N} x(i)$
Peak	$x_p = \max\|x(i)\|$
Mean Absolute	$\|\bar{x}\| = \frac{1}{N}\sum_{i=1}^{N}\|x_i\|$
Variancem	$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x(i)-\mu)^2$
Standard Deviation	$sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x(i)-\mu)^2}$
Kurtosis	$\kappa_{ij} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x_{ij}(i)-\mu_{ij}}{\sigma_{ij}}\right)^4 - 3$
Skewness	$\rho_{ij} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x_{ij}(i)-\mu_{ij}}{\sigma_{ij}}\right)^3$
Root Mean Square	$x_{rms} = \sqrt{\frac{1}{N}\sum_{n=1}^{N} x^2(n)}$
Form Factor	$W = \frac{x_{rms}}{\|\bar{x}\|}$
Peak Factor	$C = \frac{x_p}{x_{rms}}$
Impulse Factor	$I = \frac{x_p}{\|\bar{x}\|}$
Crest Factor	$L = \frac{x_p}{x_r}$

B. Feature Extraction and Selection

After the preprocessing of the signal, we extracted 29 features in time and frequency domains from each segment of the rising edge, falling edge, and dominant state. This means a total feature set of 87 features and three feature subsets are collected for each signal. Feature extraction is an attribute reduction process that enables fewer and more meaningful attributes to describe the data and greatly reduces the data load, further improving model quality as well as the speed and efficiency of machine learning algorithms. The signal is not only time-dependent but also can be converted into information such as frequency and phase. To extract the rich features, we use Fourier transform to convert the time domain signal to the frequency domain signal. Table I and II list the multidimensional features in the time and frequency domains used. For this purpose, we use the Matlab toolbox to extract features.

In the feature selection phase, we use the recursive feature elimination(RFE) method for feature selection, which can select features recursively by considering smaller and smaller sets of features. Since they preserve the original representation of the variables, the accuracy of the classifier is not reduced after removing these features and selecting only a subset of informative features, and the processing can be accelerated by reducing the complexity and dimensionality of the feature space. Therefore, feature selection can further reduce the data load and make the identification model more compliant with light-weighted standards.

C. Classification Model Generation

We input the extracted features into the machine learning model for training. Since any CAN frame sent from the same ECU will have the hardware characteristics of the sender node, the identification can be considered as a classification problem. In this phase, we trained and tested machine learning models using several classification algorithms in Scikit-Learn. Considering the lightweight perspective, we compared four typical machine learning models as shown in the table III. The K-Nearest Neighbor(KNN) algorithm is computationally expensive and requires a large amount of memory. Artificial

TABLE II
FREQUENCY DOMAIN FEATURES

Feature	Description
Centroid	$c = (\sum_{k=1}^{N} f_k s(k))/(\sum_{k=1}^{N} s(k))$
Crest	$cr = (\max(s(k))_{k=1\ldots N})/(\frac{1}{N-1}\sum_{k=1}^{N} s(k))$
Peak	$s_p = \max\lvert s(k)\rvert$
Mean	$\mu_s = \frac{1}{N}\sum_{k=1}^{N} s(k)$
Decrease	$s_d = (\sum_{k=2}^{N} \frac{s(k)-s(1)}{k-1})/(\sum_{k=2}^{N} s(k))$
Entropy	$s_e = (-\sum_{k=1}^{N} s(k)\log(s(k)))/\log(N-1)$
Flatness	$S_f = ((\prod_{k=1}^{N} s(k))^{\frac{1}{N-1}})/(\frac{1}{N-1}\sum_{k=1}^{N} s(k))$
Arithmetic Mean	$\mu_a = \frac{1}{N-1}\sum_{k=1}^{N} s(k)$
Geometric Mean	$\mu_g = (\prod_{k=1}^{N} s(k))^{\frac{1}{N-1}}$
Flux	$\mathrm{flux}(t) = \left(\sum_{k=1}^{N} \lvert s_k(t) - s_k(t-1)\rvert^P\right)^{1/p}$
Kurtosis	$\kappa_s = \left(\sum_{i=1}^{N} (y_m(i) - C_s)^4 y_m(i)\right)/\sigma_s^4 - 3$
Rolloff Point	$i, \sum_{k=1}^{i} s(k) = \kappa \sum_{k=1}^{N} s(k)$
Skewnes	$\rho_s = (\sum_{k=1}^{N} f(k)s(k))/\sigma_s^3$
Slope	$s_p = (\sum_{k=1}^{N} (f_k - \mu_f)(s(k) - \mu_S))/(\sum_{k=1}^{N} (f_k - \mu_f)^2)$
Spread	$\sigma_s = \sqrt{\left(\sum_{k=1}^{N} (f_k - c)^2 s(k)\right) / \sum_{k=1}^{N} (s(k))}$

Neural Network(ANN) has high memory usage due to its requirement of lots of parameters, long learning time and high computational cost. Neither of these two models is suitable for the light-weighted requirement. Logistic Regression (LR) is simple to implement, computationally small, fast in classifying, and requires fewer storage resources. Random Forest(RF) can be parallelized to speed up computation and handle large datasets, making training faster and more accurate for imbalanced datasets. These two algorithms are more in line with the light-weighted requirement. Therefore, we choose logistic regression and random forest for comparison experiments to train the optimal machine learning model.

TABLE III
PERFORMANCE COMPARISON OF CLASSIFICATION MODELS

Model	Memory Usage	Computing Speed	Training Speed
RF	–[a]	–	◯
LR	◯ [b]	◯	–
KNN	×[c]	×	–
ANN	×	×	×

[a] "–" means average.
[b] "◯" means suitable.
[c] "×" means not suitable.

The generation of machine learning models is divided into two major phases, namely training and testing. In the training phase, each sample is labeled with the ECU ID. Physical features are extracted from the CAN voltage signal as the fingerprint of the ECU to set up the feature set, and the best feature subset is selected for model training by feature selection. For different datasets, cross-validation is applied to adjust the model parameters to improve the generalization ability and computational efficiency of the model. The completed model after training is a multi-classification classifier. In the testing phase, the fingerprints and IDs are extracted using the new CAN information as test data, and then the trained machine learning model is used for classification. Finally, we obtained the predicted ID and the actual ID for each sample, the predicted ID and the actual ID can be used to evaluate the accuracy of the model identification. We compared the machine learning models by evaluating the accuracy and false positive rate and the prediction time.

IV. EXPERIMENTAL RESULTS

This section introduces our evaluation of ECU identification method on actual vehicles and CAN bus prototype. A series of experiments have proved that our method has high accuracy and low false detection rate, which accord with the light-weighted standard.

A. Datasets

We used public datasets from ten actual vehicles in [22] and a dataset sampled from the CAN bus prototype for our experiment. The dataset was split into a training (70%) and test set (30%)for each run.

1) Actual Vehicles: Actual vehicles datasets which included 9 cars and 1 tractor came from a public paper published by Lucian Popa et al [22]. They used Scope 5000 Series devices to collect voltage data. The sample rate was set to 500 MS/s, with a sample interval of 2 nanoseconds. The specific vehicles information as already shown in Table IV. The cars used fall in three different body configurations with manufacturing dates between 2006 and 2021. The datasets provide a reliable basis for the experiment by fully considering the diversity of vehicles. the dataset is publicly released to serve for future investigations and can be retrieved from the ECUPrint project on GitHub and the authors institution server

TABLE IV
INTRODUCTION OF VEHICLES AND DATA COLLECTION

Vehicle	Model year	No. ECUs	Collected bits (voltage)
Honda Civic	2012-2017	6	40,073
Opel Corsa	2006-2014	4	9,187
Hyundai i20	2014-2020	7	17,767
John Deere Tract.	2010-2018	3	4,021
Dacia Duster	2010-2017	3	9,086
Dacia Logan	2012-2019	6	31,579
Hyundai ix35	2009-2015	6	23,104
Ford Fiesta	2017-2020	6	43,861
Ford Kuga	2013-2019	9	28,024
Ford Ecosport	2018-2021	4	22,808

[a] The dataset is publicly released.

2) CAN Bus Prototype: We set up a CAN bus prototype to simulate a CAN bus network. The CAN Bus Prototype consists of four Arduino Uno boards with CAN bus shields connected by twisted-pair cable of the same channel length, as shown in Fig.5. Three of the simulation ECUs were programmed to transmit respectively standard frames with different message IDs but identical data contents, and one received the signals as the receiver. The message IDs are used to label the different ECUs, and the identical data contents are used to assure that only the physical features of the voltage signal are considered in our experiments. In addition, we collected the electrical signal of the message frame at the receiver by random sampling with an oscilloscope TBS1102C and set the sampling rate to 250 MS/s and the number of sampling points to 2000. The 18724 bits of electrical signals were collected from CAN-H and CAN-L respectively.

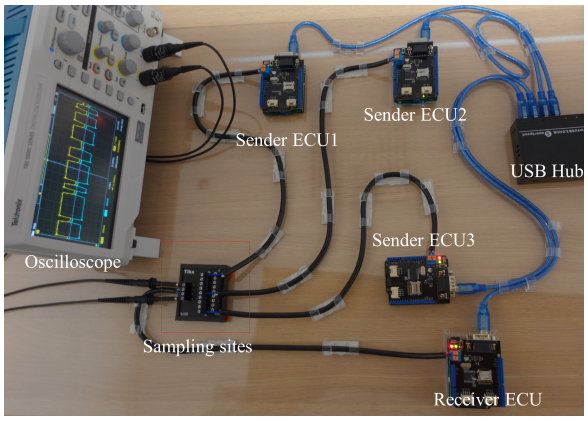


Fig. 5. CAN Bus Prototype

B. Feature Extraction and Selection

We extracted time and frequency domain features from the rising edge, dominant state and falling edge of the differential signals to obtain 87-dimensional total feature sets, rising edge feature subsets, falling edge feature subsets and dominant state feature subsets.

The Linear Discriminant Analysis(LDA) dimensionality reduction technique was used to process the total feature set, and two-dimensional scatter plots are shown in Fig.6. It can be noticed that the projections of the voltage data from the same ECU are generally clustered together when the amount of ECUs is less, conversely, there are obvious distances between the different ECUs.However, when the amount of ECUs is more, the projection points may overlap. We can preliminarily conclude that the physical characteristics of the CAN voltage data could be used to classify different ECUs, but it is necessary to further train the machine learning classification model for identification.

The total feature set and feature subsets of rising edge, falling edge and dominant states were individually put into the machine learning model for testing. The results on random forest and logistic regression are shown in Fig.7. We have found that even using a part of the feature subset, such as the

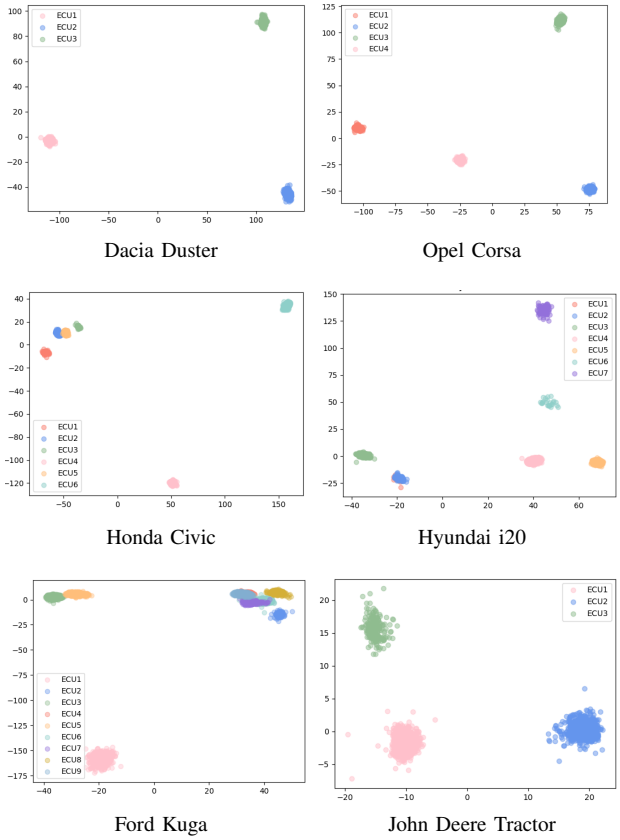


Fig. 6. Two-dimensional scatter plot after feature extraction on five typical real cars with different number of ECUs.

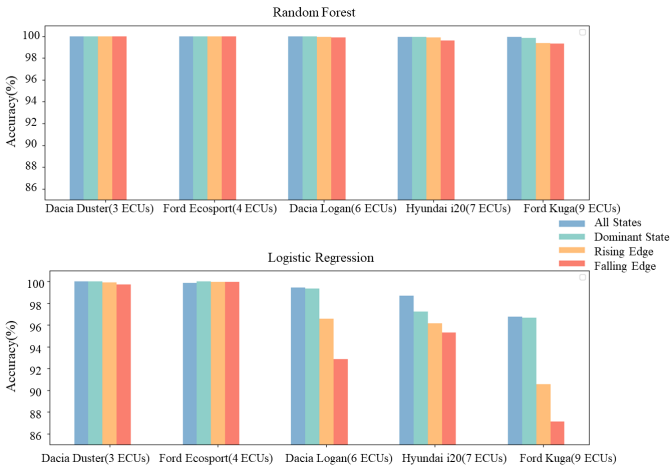
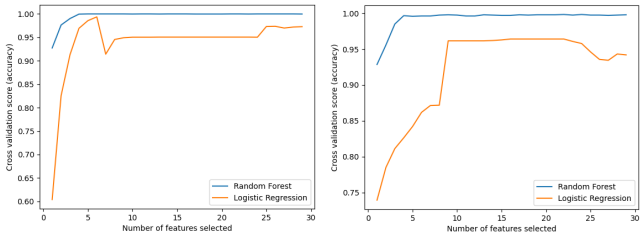


Fig. 7. Accuracy of the three feature subsets and the total feature set in two machine learning models.

dominant state part, still achieves a similarly higher accuracy as using the full feature set. It may be that because the dominant voltage state is actively driven by the transmitter but the recessive state is passively returned to voltage by a resistor, the dominant state may include relatively more electrical characteristics that enable the ECU to be identified.

We chose recursive feature elimination for feature selec-



Ford Fiesta John Deere Tractor

Fig. 8. Feature selection for two types of vehicles.

tion.The Fig.8shows the example of the cross-validation results on the training set.For random forest,the accuracy remains relatively constant as the number of features selected varies,but for logistic regression it affects the accuracy. The number of features selected is automatically adjusted by cross-validation to ensure that the optimal number of features is selected. Most of the vehicles selected the optimal number of features with less than 10. Therefore, by using only a subset of features from the dominant state and feature selection, we enable to achieve high identification accuracy rate while reducing the data load and reach the lightweight effect.

C. Evaluation of ECU identification

In this subsection, the results of the experiment on actual vehicles and on the CAN bus prototype are presented separately.

TABLE V
RESULTS OF ACTUAL VEHICLES

Dataset	No. ECUs	Model	Predicted Time(ms)		ACC(%)	FPR(%)
			Feature Selection			
			yes	no		
Dacia Duster	3	RF	0.87	2.39	100	0.00
		LR	0.12	1.73	100	0.00
Ford Kuga	9	RF	3.29	7.04	99.87	0.00
		LR	0.41	3.01	98.25	0.01
Dacia Logan	6	RF	3.86	10.62	100	0.00
		LR	0.26	5.87	99.93	0.00
Ford Ecosport	4	RF	3.75	5.33	100	0.00
		LR	0.13	2.30	100	0.00
Hyundai i20	7	RF	3.26	7.15	99.63	0.01
		LR	0.10	1.65	99.42	0.01
Ford Fiesta	6	RF	2.61	7.21	100	0.00
		LR	0.25	3.33	99.67	0.01
Honda Civic	6	RF	2.15	7.52	100	0.00
		LR	0.23	3.91	98.35	0.01
Hyundai ix35	6	RF	4.28	8.31	99.93	0.00
		LR	0.45	4.45	96.43	0.05
John Deere Tractor	3	RF	0.24	1.86	99.41	0.00
		LR	0.08	1.15	96.88	0.02
Opel Corsa	4	RF	0.81	5.74	100	0.00
		LR	0.12	4.62	100	0.00

1) Actual Vehicles: Table V shows the experimental results for 10 vehicles, including the model prediction time, accuracy(ACC) and false positive rate(FPR). We compared the model prediction time using the subset of dominant state features with feature selection and the total feature set without feature selection. It can be seen that our method can improve the efficiency of the model and ensure high accuracy and low false detection rate.

2) CAN Bus Prototype: Performing our method on CAN bus prototype. Fig.9 shows the results of feature extraction and feature selection.Fig.10 show the random forest and logistic regression confusion matrices. The confusion matrix clearly shows the identification of each ECU. From the two figures, we can find that the random forest performs significantly better than the logistic regression.

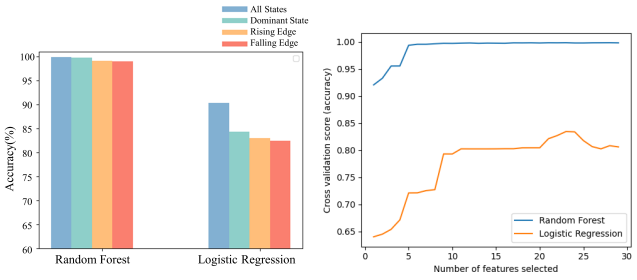
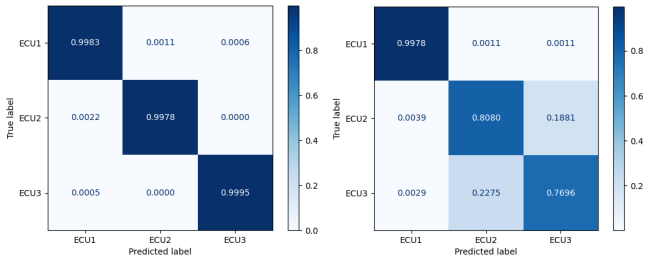


Fig. 9. Feature extraction and feature selection on CAN bus prototype



RF LR

Fig. 10. CAN Bus Prototype Confusion Matrix

From the results of the experiments on actual vehicles and CAN bus prototypes, there is evidence that the accuracy of random forest is higher than that of logistic regression, which indicates that physical characteristics identification is not a linear classification problem. The accuracy of the logistic regression model on the vehicle dataset is mostly good, indicating that the physical characteristics of the ECU devices in the real vehicle environment has greater disparity; however, the accuracy of the logistic regression model decreases significantly when used in the CAN prototype simulation environment (which has less disparity between devices). In contrast, the random forest has high accuracy both in the vehicle and simulation environment. Thus we can conclude that random forest is suitable for physical fingerprinting in terms of accuracy metrics.

V. CONCLUSION

In this paper, we propose a light-weighted machine learning based ECU identification method by extracting physical characteristics of CAN signals in both time and frequency domains. Use only dominant segment physical features and

feature selection. Our method decrease the data load and the computational burden of the CAN bus, enables accurate identification of message senders with high efficiency. Validated the feasibility and effectiveness of the method on actual vehicles datasets and CAN bus prototypes. Provides authentication capabilities for in-vehicle networks. This light-weighted ECU identification method can be used as an effective tool in intrusion detection systems.

REFERENCES

[1] W. Zeng, M. A. Khalid, and S. Chowdhury, "In-vehicle networks outlook: Achievements and challenges," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 1552–1571, 2016.

[2] A. Hafeez, H. Malik, O. Avatefipour, P. R. Rongali, and S. Zehra, "Comparative study of can-bus and flexray protocols for in-vehicle communication," SAE Technical Paper, Tech. Rep., 2017.

[3] M. Di Natale, H. Zeng, P. Giusto, and A. Ghosal, *Understanding and using the controller area network communication protocol: theory and practice*. Springer Science & Business Media, 2012.

[4] M. Bozdal, M. Samie, and I. Jennions, "A survey on can bus protocol: Attacks, challenges, and potential solutions," in *2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE)*. IEEE, 2018, pp. 201–205.

[5] J. den Hartog, N. Zannone *et al.*, "Security and privacy for innovative automotive applications: A survey," *Computer Communications*, vol. 132, pp. 17–41, 2018.

[6] J. Cui, L. S. Liew, G. Sabaliauskaite, and F. Zhou, "A review on safety failures, security attacks, and available countermeasures for autonomous vehicles," *Ad Hoc Networks*, vol. 90, p. 101823, 2019.

[7] D. K. Nilsson, U. E. Larson, and E. Jonsson, "Efficient in-vehicle delayed data authentication based on compound message authentication codes," in *2008 IEEE 68th Vehicular Technology Conference*. IEEE, 2008, pp. 1–5.

[8] K.-T. Cho and K. G. Shin, "Fingerprinting electronic control units for vehicle intrusion detection." in *USENIX Security Symposium*, vol. 40, 2016, pp. 911–27.

[9] S. U. Sagong, X. Ying, A. Clark, L. Bushnell, and R. Poovendran, "Cloaking the clock: Emulating clock skew in controller area networks," in *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 2018, pp. 32–42.

[10] X. Ying, S. U. Sagong, A. Clark, L. Bushnell, and R. Poovendran, "Shape of the cloak: Formal analysis of clock skew-based intrusion detection system in controller area networks," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 9, pp. 2300–2314, 2019.

[11] P.-S. Murvay and B. Groza, "Source identification using signal characteristics in controller area networks," *IEEE Signal Processing Letters*, vol. 21, no. 4, pp. 395–399, 2014.

[12] W. Choi, H. J. Jo, S. Woo, J. Y. Chun, J. Park, and D. H. Lee, "Identifying ecus using inimitable characteristics of signals in controller area networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 6, pp. 4757–4770, 2018.

[13] 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 Transactions on Information Forensics and Security*, vol. 13, no. 8, pp. 2114–2129, 2018.

[14] M. Kneib and C. Huth, "Scission: Signal characteristic-based sender identification and intrusion detection in automotive networks," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 787–800.

[15] M. Kneib, O. Schell, and C. Huth, "Easi: Edge-based sender identification on resource-constrained platforms for automotive networks." in *NDSS*, 2020, pp. 1–16.

[16] O. Avatefipour, "Physical-fingerprinting of electronic control unit (ecu) based on machine learning algorithm for in-vehicle network communication protocol "can-bus"," Ph.D. dissertation, 2017.

[17] Z. Deng, Y. Xun, J. Liu, and Y. Zhao, "A lightweight sender identification scheme based on vehicle physical layer characteristics," in *ICC 2022-IEEE International Conference on Communications*. IEEE, 2022, pp. 3334–3339.

[18] K.-T. Cho and K. G. Shin, "Viden: Attacker identification on in-vehicle networks," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 1109–1123.

[19] A. Hafeez, K. Topolovec, and S. Awad, "Ecu fingerprinting through parametric signal modeling and artificial neural networks for in-vehicle security against spoofing attacks," in *2019 15th International Computer Engineering Conference (ICENCO)*. IEEE, 2019, pp. 29–38.

[20] K. Verma, M. Girdhar, A. Hafeez, and S. S. Awad, "Ecu identification using neural network classification and hyperparameter tuning," in *2022 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2022, pp. 1–6.

[21] L. Xiao, X. Lu, T. Xu, W. Zhuang, and H. Dai, "Reinforcement learning-based physical-layer authentication for controller area networks," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2535–2547, 2021.

[22] L. Popa, B. Groza, C. Jichici, and P.-S. Murvay, "Ecuprint—physical fingerprinting electronic control units on can buses inside cars and sae j1939 compliant vehicles," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1185–1200, 2022.

[23] N. Liu, C. Moreno, M. Dunne, and S. Fischmeister, "vprofile: Voltage-based anomaly detection in controller area networks," in *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2021, pp. 1142–1147.

[24] S. Ahmed, M. Juliato, C. Gutierrez, and M. Sastry, "Two-point voltage fingerprinting: Increasing detectability of ecu masquerading attacks," *arXiv preprint arXiv:2102.10128*, 2021.

[25] S. Bellaire, M. Bayer, A. Hafeez, R. U. D. Refat, and H. Malik, "Fingerprinting ecus to implement vehicular security for passenger safety using machine learning techniques," in *Intelligent Systems and Applications: Proceedings of the 2022 Intelligent Systems Conference (IntelliSys) Volume 3*. Springer, 2022, pp. 16–32.