Systematic Intrusion Detection Technique for an In-Vehicle Network Based on Time-Series Feature Extraction

Hiroki Suda, Masanori Natsui, and Takahiro Hanyu
Reserch Institute of Electrical Communication, Tohoku University
2-1-1 Katahira, Aoba-ku, Sendai 980-8577, Japan
Phone/Fax: +81-22-217-5508
E-mail: hiroki.suda.t2@dc.tohoku.ac.jp, {natsui, hanyu}@riec.tohoku.ac.jp

***Abstract*— In this paper, we propose a systematic intrusion detection algorithm based on time-series feature extraction for an in-vehicle network. Since packet-type valid data are transmitted inside an in-vehicle network periodically, illegal data due to unauthorized intrusion attack can be easily and uniformly detected by using periodical time-series feature of valid data, where recurrent neural network is a key tool to efficiently extract their time-series feature. In fact, through an evaluation using data acquired from actual vehicles, we show that the proposed method can detect typical intrusion attack patterns such as data modification attack and injection attack.**

Keywords—car security; controller area network; intrusion detection system; deep learning; recurrent neural network

I. Introduction

As demand for IoT (Internet of Things) increases, various services such as electricity, gas, water, railroad, aircraft, automobile and so on are provided and controlled through the network [1]. In contrast, it is urgent to upgrade illegal intrusion detection technology in in-vehicle networks responsible for automobile control [2]. It is known that there are several methods reported against unauthorized intrusion attacks, such as a method of preventing falsification of information part of data by using machine learning technique [3] and a method of detecting illegal packet contamination based on the periodicity of data propagating through the network [4]. However, considering the increasing diversification of attack approach with automotive IoT technology becoming more widespread and sophisticated, it will be difficult to deal with the conventional approaches based on correspondence to individual cases. Therefore, along with the spread of IoT technology for automobiles and other advanced automotive support systems, it is urgent to establish an in-vehicle network intrusion detection technique that can detect and block unauthorized intrusion quickly and flexibly in various ways.

In this paper, we propose an intrusion detection technique for an in-vehicle network that can systematically detect various attack methods. As a basic concept of that, we show the effectiveness of algorithms using not only individual data but also the periodicity of data existing in the in-vehicle network by performing time-series feature extraction. Concretely, we design the algorithm using recurrent neural network (RNN) which is one of the deep learning algorithms that can learn the temporal features of the data time series. Through computer simulation evaluation experiments on data acquired from vehicles, we show that the proposed technique can detect typical intrusion attack patterns such as data modification attack and injection attack systematically, and achieve higher detection rate than conventional threshold-based methods.

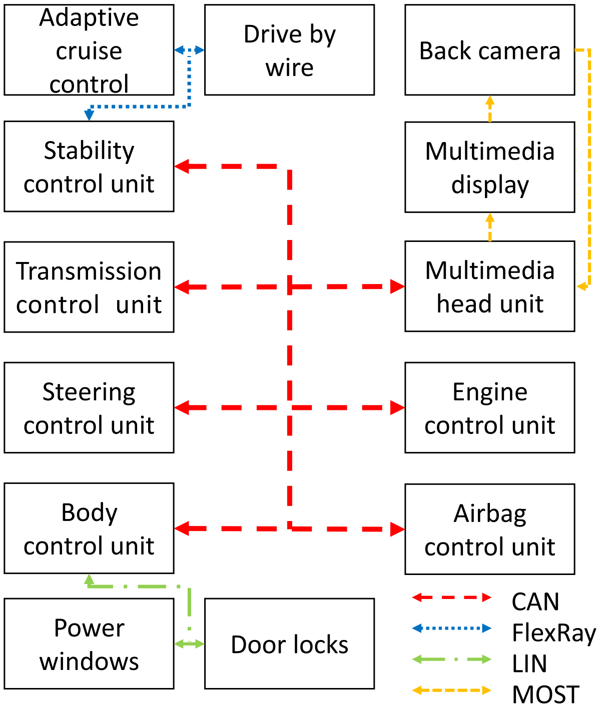
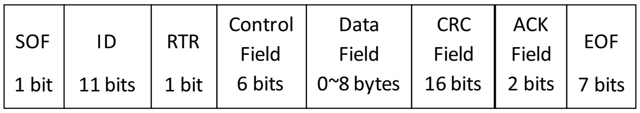


Fig. 1. Configuration example of an in-vehicle network.

II. In-Vehicle Network Overview

Figure 1 shows the configuration of the in-vehicle network. The in-vehicle network is responsible for data transmission and reception between ECUs (electronic control units) mounted in the vehicle [5]. As described below, there are various

2378-2226/18/$31.00 ©2018 IEEE
DOI 10.1109/ISMVL.2018.00018



SOF 1 bit	ID 11 bits	RTR 1 bit	Control Field 6 bits	Data Field 0~8 bytes	CRC Field 16 bits	ACK Field 2 bits	EOF 7 bits

Fig. 2. Format of CAN 2.0 data frame.

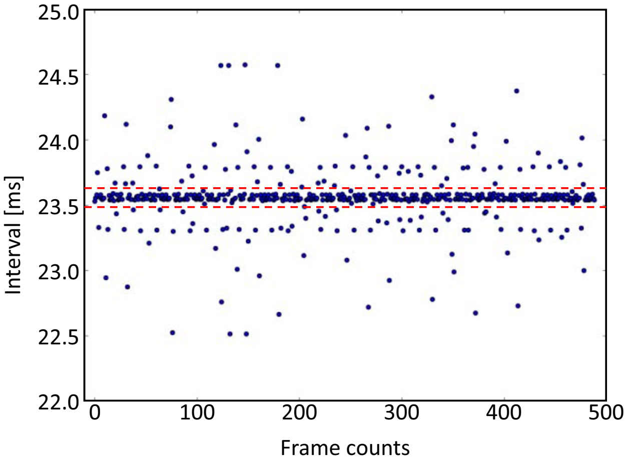


Fig. 3. Periodicity of CAN frame in single ID. Most of the frames are in the dashed line, which indicates the periodicity of the CAN frame, while there are some frames that interval is out of range.

communication protocols of the in-vehicle network with different transmission speed and cost, and it is selectively used according to the characteristics of each application [6, 7].

- Controller area network (CAN): A network forming the core of automobile control such as power train and steering
- FlexRay: High-speed and high-cost next generation protocol used for multimedia and X-by-wire
- Local interconnect network (LIN): A low cost mounted network which is used to the application that does not require a high speed data transfer such as the control of door mirrors, reclining seats, etc.
- Media oriented systems transport (MOST): A network for multimedia application such as audio and car navigation systems

In this paper, we focus on controller area network (CAN), which is the core protocol of in-vehicle network such as powertrain and steering. Figure 2 shows a frame structure of data propagating in a CAN called a CAN frame. The roles of each field are as follows [8]:

- SOF: Represents the start of a frame. It is used for synchronization between the ECUs,
- ID: Represents the destination of the frame. It is used for determining the priority of communication,

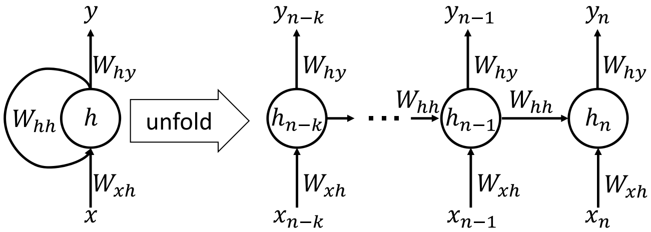


Fig. 4. Recurrent neural network. Since the output of the hidden layer is used for input of the hidden layer of the next step, it is possible to determine the current output based on past inputs.

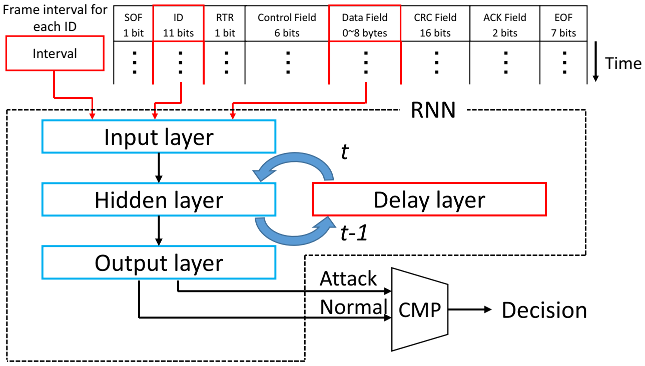


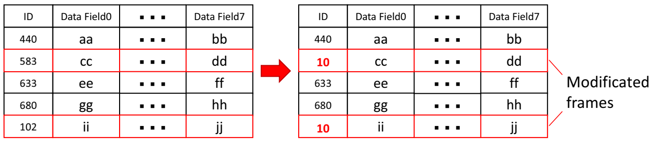
Fig. 5. Proposed technique. The frame interval, ID value, and Data Field value of each frame are used as input. The algorithm compares the similarity between input frame and normal frame, or input frame and attack frame, and determines the output.

- RTR: Identifies whether the frame is a remote frame (a frame of data transmission request) or a data frame (a response to the request),
- Control Field: Identifies whether the frame is a standard frame or an extended frame, and the data length of the Data Field,
- Data Field: Indicates data to send,
- CRC Field: Contains data used for cyclic redundancy check of SOF, ID, Control Field and Data Field,
- ACK Field: Indicates whether the frame is transmitted successfully or not,
- EOF: Represents the end of the frame.

Among them, the ID field representing the destination of the frame and the Data Field representing the data to be transmitted are the main targets of data falsification in the attack. Specifically, attacks that cause malfunctions due to abnormal data transmission to a specific ECU are made by changing the ID field. In addition, by changing the Data Field part, an attack inducing an accident caused by a difference between the display

Table 1. Basic configuration of RNN.

Input layer	10 nodes (Interval value × 1, ID value × 1, Data Field value × 8)
Hidden layer	100 nodes (LSTM)
Output layer	2 nodes (Attack or Normal)
Optimization algorithm	Adaptive Moment Estimation (Adam) [14]



ID	Data Field0	▪ ▪ ▪	Data Field7
440	aa	▪ ▪ ▪	bb
583	cc	▪ ▪ ▪	dd
633	ee	▪ ▪ ▪	ff
680	gg	▪ ▪ ▪	hh
102	ii	▪ ▪ ▪	jj

ID	Data Field0	▪ ▪ ▪	Data Field7
440	aa	▪ ▪ ▪	bb
10	cc	▪ ▪ ▪	dd
633	ee	▪ ▪ ▪	ff
680	gg	▪ ▪ ▪	hh
10	ii	▪ ▪ ▪	jj

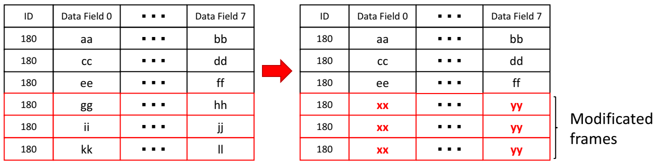
Fig. 6. Attack scenario (1 – a): the value of ID is changed to a small value with a certain probability.

part (e.g. speed meter) and the actual situation (car operation speed) is assumed.

Meanwhile, there is also a technique called flood attack that transmits a large number of attack frames on the in-vehicle network. This attack interferes with normal frame transmission between ECUs by transmitting a large number of frames with small IDs at shorter cycles than usual. This attack utilizes the fact that CAN adopts a network based on a common bus and that a frame having a smaller ID value takes precedence in data transmission and reception between ECUs.

As a countermeasure against this attack, Ref. [2] has proposed a method that determines data transmission with a period shorter than the threshold value of the data transmission cycle defined in advance for each ID value as an attack. This method is based on the fact that in normal CAN communication, frames having a common ID value are transmitted at a substantially constant period as shown in Fig. 3. In this method, it is necessary to set an appropriate threshold value in consideration of period fluctuation caused by sharing the same bus among a large number of ECUs. If a strict threshold value is set, communication efficiency decreases by misjudging normal frames as attacks. On the other hand, if a loose threshold value is set, the detection failure of the attack frame occurs, resulting in decreased reliability. Therefore, in this method, there is a trade-off between communication efficiency and reliability, making it difficult to achieve both requirements simultaneously.

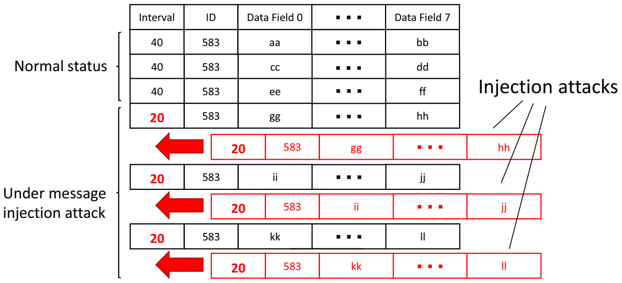
As described above, intrusion attack methods targeting in-vehicle networks can be roughly classified into (1) those targeting the contents of transmitted and received data and (2) those targeting temporal trends of transmitted and received data. In consideration of the increasing diversification of intrusion attack methods on the in-vehicle network with the spread of in-vehicle IoT technology, it is urgent to propose a method that can handle various attacks including the above in a unified manner and detect it flexibly and quickly.



ID	Data Field 0	▪ ▪ ▪	Data Field 7
180	aa	▪ ▪ ▪	bb
180	cc	▪ ▪ ▪	dd
180	ee	▪ ▪ ▪	ff
180	gg	▪ ▪ ▪	hh
180	ii	▪ ▪ ▪	jj
180	kk	▪ ▪ ▪	ll

ID	Data Field 0	▪ ▪ ▪	Data Field 7
180	aa	▪ ▪ ▪	bb
180	cc	▪ ▪ ▪	dd
180	ee	▪ ▪ ▪	ff
180	xx	▪ ▪ ▪	yy
180	xx	▪ ▪ ▪	yy
180	xx	▪ ▪ ▪	yy

Fig. 7. Attack scenario (1 – b): the value of Data Field is changed to another value with a certain probability



Interval	ID	Data Field 0	▪ ▪ ▪	Data Field 7
40	583	aa	▪ ▪ ▪	bb
40	583	cc	▪ ▪ ▪	dd
40	583	ee	▪ ▪ ▪	ff
20	583	gg	▪ ▪ ▪	hh
20	583	gg	▪ ▪ ▪	hh
20	583	ii	▪ ▪ ▪	jj
20	583	ii	▪ ▪ ▪	jj
20	583	kk	▪ ▪ ▪	ll
20	583	kk	▪ ▪ ▪	ll

Fig. 8. Attack scenario (2 – a): frames with short intervals are inserted.

III. Intrusion Detection Technique Using Recurrent Neural Network

In this paper, we propose a method to detect illegally altered or mixed frames based on the temporal characteristics of CAN frame, specifically (1) information of each frame and (2) frame periodicity. Specifically, we propose an intrusion detection method based on RNN (Recurrent neural network) which is one of machine learning algorithms as a basic algorithm.

Neural network (NN) is known as a mathematical model aiming at expressing the characteristics found in cranial nervous systems of living organisms by computer simulation [9]. The NN is composed of an input layer, some hidden layers, an output layer, and wires connecting them. Each layer is composed of a set of nodes. An input signal given to the input layer propagates to the hidden layer after superimposing the weight given to each wire. In the hidden layer, the propagated signals are summed for each unit, and the output of each unit is determined based on an activation function. Similarly, the output from the hidden layer is propagated to the output layer after superimposing the weights, and the output as NN is determined through similar processing. By giving input / output samples and applying so-called supervised learning by back propagation, the weight of each wire is updated and NN can operate with the desired function.

On the other hand, the RNN has a recursive structure as shown in Fig.4 [10]. In this structure, the input signal x is superimposed with W_{xh} which is the weight between the input layer and the hidden layer, applied to the activation function, and then propagated to the hidden layer h. The output of the hidden layer is superimposed with the weight W_{hh} and is used as a part

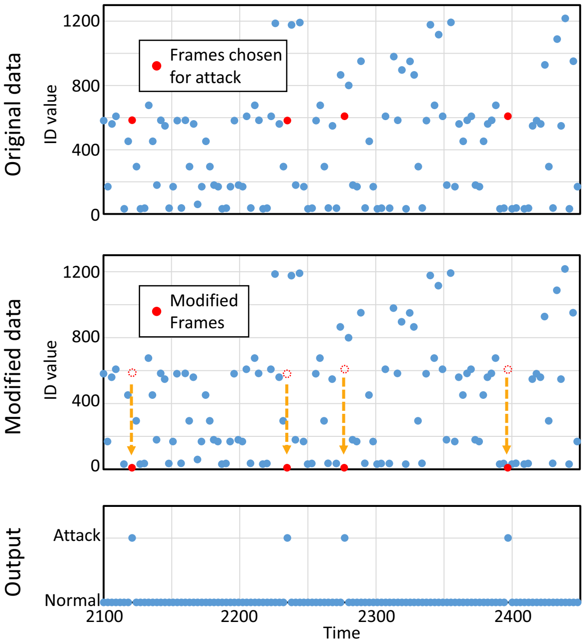


Fig. 9. The result of detecting frames with abnormal ID values.

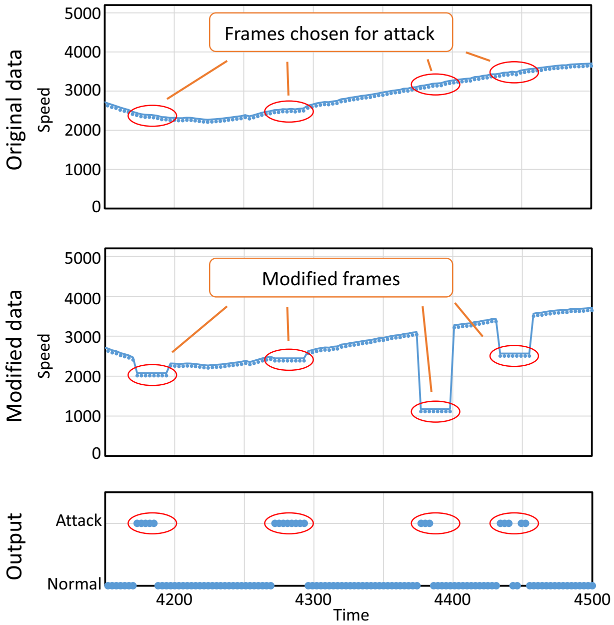


Fig. 10. The result of detecting frames with abnormal Data Field values.

of the input of the hidden layer at next time step. Therefore, the output of the hidden layer is determined not only by the input signal x_n at the current time step but also by the feedback signals based on past input signals x_{n-1}, x_{n-2}, ..., and x_{n-k}. In the same way, the output from the hidden layer is propagated to the output layer after superimposing the weight W_{hy}, and the output y is determined.

With this structure, the RNN can obtain the output considering not only the input at a certain time but also the relation with the input of the previous time. That is, the RNN can acquire the periodic features of the input time series information through supervised learning, and can flexibly cope with fluctuations in the cycle. Utilizing this property, RNN is widely used in speech signal analysis [11], context estimation [12], and so on.

Figure 5 shows a configuration of the CAN intrusion detection technique using RNN. In this technique, the ID value and Data Field value, which are main targets in intrusion attack, are used as input. Also, to learn the time-series features of the CAN frame, the frame interval for each ID is also used as input. By doing so, the proposed system can learn the periodic features of the CAN frame as shown in Fig. 3, taking into consideration the variation of the period.

The output layer has two nodes, and outputs similarities between input frames and normal frames, or input frames and attack frames, respectively as a numerical value. Based on the magnitude comparison of the output values of the two nodes, it is determined whether the input frame is an attack or not. In the next chapter, we evaluate the performance of the proposed technique implemented using the deep learning framework Chainer [13].

IV. Evaluation

In this paper, we consider three kinds of attack patterns shown below as evaluation targets.

1) Modification of ID and Data Field

1-a) Forcing to send frames to the wrong ECU by changing the value of ID field in CAN flames to an illegal value as shown in Fig. 6

1-b) Causing malfunction by changing the value of Data Field in CAN flames as shown in Fig. 7

2) Flood Attack

2-a) Interfering with the communication of the normal frame by occupying the bus in the attack frame as shown in Fig. 8.

In this experiment, we use time series data of the CAN frame acquired from the actual vehicle by the in-vehicle network frame monitor. After applying data modification equivalent to the above attack patterns randomly, the data is used as learning data of the RNN. Table 1 shows the main parameters of RNN used in this experiment.

Experimental results on detection of frames with abnormal ID values (attack pattern 1-a)) are shown in Fig. 9. The graph shows original CAN data acquired from the actual vehicle, modified data in which the values of IDs of some selected frames are modified, and the output. The horizontal axis indicates the transmission time of each frame. The vertical axes of the upper two figures indicate the value of ID of each frame, and that of the bottom figure indicates whether the evaluation result for each frame is attack or normal. We confirmed that the proposed

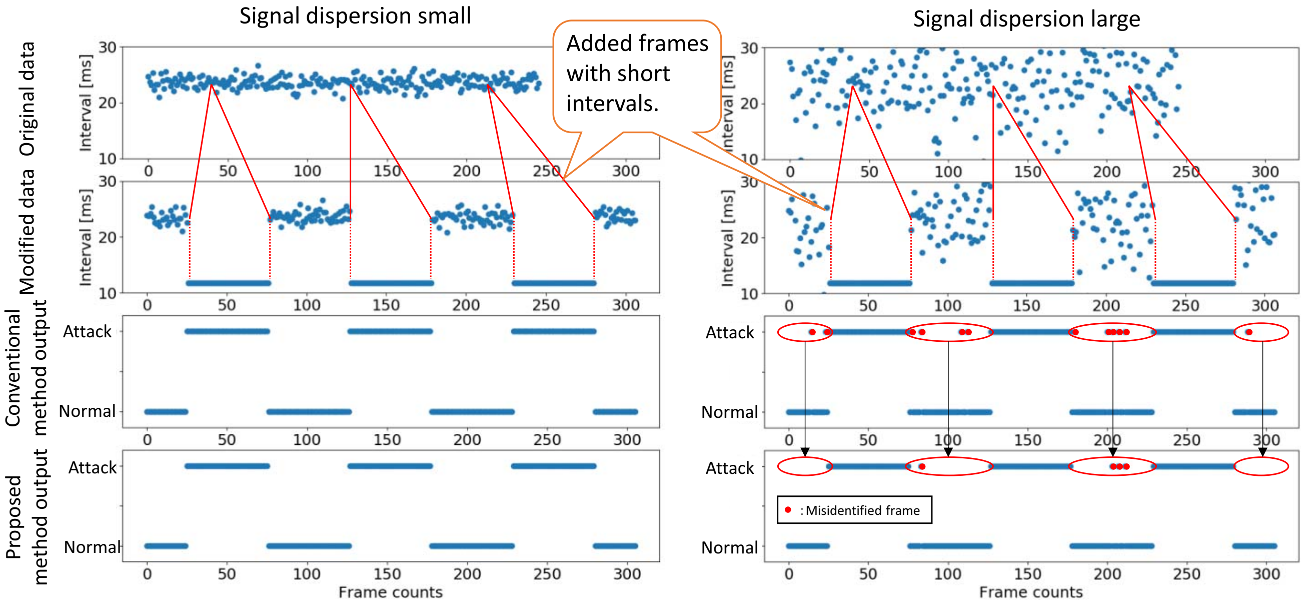


Fig. 11. The result of detecting flood attack. When the variance of the original data is large, the proposed method could achieve higher recognition rate as compared with the conventional method.

technique correctly judged the data as an attack frame when a modified frame was input.

Experimental results on detection of frames with abnormal Data Field values (attack pattern 1-b)) are shown in Fig. 10. As in Fig. 9, each graph indicates original data, modified data, and the output. The vertical axes of the upper two figures indicate the values of Data Fields that corresponds to speed value transmitted to the speedometer, and that of the bottom figure indicates whether the evaluation result for each frame. In this experiment, the frame transferred to the speedometer is targeted, and the Data Field value is modified to a value different from the original value for a certain period of time. Although some omission of detection occurred, it was confirmed that almost accurate detection was possible.

It is worth noting that the proposed technique can detect attack frames even if the Data Field values of the attack frames are within a range that can normally be taken. This result is obtained by determining the output based on the time series features including past inputs, instead of using only current input, which suggests the effectiveness of using RNN as the basic algorithm.

Experimental results on detection of flood attack (attack pattern 2-a)) are shown in Fig. 11. Since the degree of variance of the input signal depends on the congestion situation of CAN, we evaluated both cases when the variance is small (left figure) and when the variance is large (right picture). The figure shows CAN data acquired from the actual car, the modification data, the identification result based on the conventional threshold-based method [4] and the identification result based on the proposed method. The horizontal axis for each graph represents the index of the frames. The vertical axes of the upper two rows represent the time interval between frames, and those of the lower two rows indicate whether the frame was determined to be an attack frame or a normal frame. In this experiment, data with a frame interval shorter than usual was inserted as attack data into the original data, and it was evaluated whether it can be determined by the proposed technique. When the variance of the original data is small, all attacks can be identified with both the conventional method and the proposed method. On the other hand, when the variance of the original data is large, false recognition occurred in both methods, but it was confirmed that the proposed method could achieve higher recognition rate as compared with the conventional method.

Figure 12 shows the detailed performance comparison with the conventional method for flood attacks. The horizontal axis indicates a coefficient of variation, which is a coefficient representing the magnitude of the variance of the input data. The vertical axis indicates the accuracy, that is, the recognition rate of the technique. In the conventional threshold-based method, since the tendency of the accuracy depends on a preset threshold value, different characteristics indicated by several blue lines are shown. Among them, the bold blue line indicates the highest accuracy by a conventional method, and the bold red line indicates the accuracy obtained by the proposed method. We confirmed that the proposed method achieves a sufficiently high accuracy as compared with the conventional method despite that there is no need to set a threshold.

V. Conclusion

In this paper, we have proposed an intrusion detection method based on time-series feature extraction for in-vehicle network which can respond flexibly to various attacks. Specifically, we confirmed that various attacks such as ID modification attack, Data Field modification attack, and flood attack can be detected systematically with same algorithm. Furthermore, for the identification of flood attacks, the

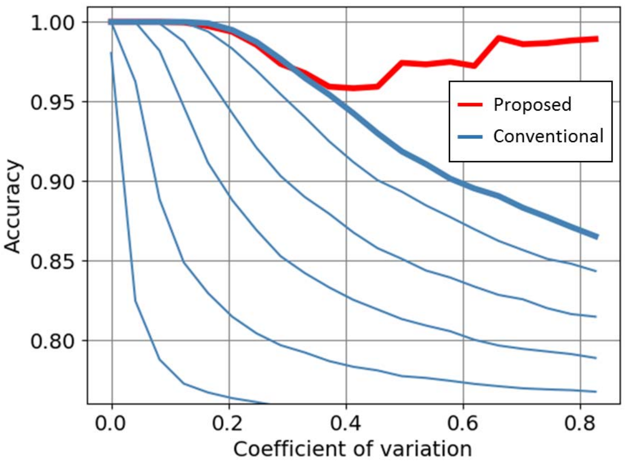


Fig. 12. Performance comparison with conventional methods targeting flood attack. The accuracy of the conventional threshold-based method depends on the preset threshold value, while the proposed method achieves a high accuracy despite that there is no need to set a threshold value.

proposed method flexibly identifies and achieves a high accuracy without having to decide thresholds, as compared with the conventional threshold-based method. In the future, we will consider performance evaluation for more various attack patterns and hardware implementation.

Acknowledgment

The authors thank Prof. S. Nagayama and Associate Prof. H. Inoue of Hiroshima City University for excellent technical assistance. Part of this work was carried out under the Cooperative Research Project Program of the RIEC, Tohoku University, Brainware LSI Project by MEXT, JST OPERA, and JSPS KAKENHI Grant Number 16KT0187.

References

[1] R. Porkodi and V. Bhuvaneswari, "The Internet of Things (IoT) Applications and Communication Enabling Technology Standards: An Overview," 2014 International Conference on Intelligent Computing Applications (ICICA), Mar. 2014.

[2] Jiajia Liu, Shubin Zhang, Wen Sun, and Yongpeng Shi, "In-Vehicle Network Attacks and Countermeasures: Challenges and Future Directions," IEEE Network, vol. 31, no. 5, pp 50-58, Sep. 2017.

[3] Min-Ju Kang and Je-Won Kang, "A Novel Intrusion Detection Method Using Deep Neural Network for In-Vehicle Network Security," 2016 IEEE 83rd Vehicular Technology Conference (VTC Spring), pp. 1-5, May. 2016.

[4] Hyun Min Song, Ha Rang Kim, and Huy Kang Kim, "Intrusion detection system based on the analysis of time intervals of CAN messages for in-vehicle network," 2016 International Conference on Information Networking (ICOIN), pp. 63-68, Jan. 2016.

[5] F. Yu, D.-F. Li, and D. Crolla, "Integrated vehicle dynamics controlstateof-theart review," in Vehicle Power and Propulsion Conference, 2008. VPPC'08. IEEE. IEEE, 2008, pp. 1–6.

[6] Amos Albert and Robert Bosch GmbH, "Comparison of Event-Triggered and Time-Triggered Concepts with Regard to Distributed Control Systems," Embedded world 2004, pp. 235-252, Feb. 2004.

[7] Ulf E. Larson, Dennis, K. Nilsson, and Erland Jonsson, "An approach to specification-based attack detection for in-vehicle networks," 2008 IEEE Intelligent Vehicles Symposium, pp. 220-225, Jun. 2008.

[8] Dennis K. Nilsson, Ulf E. Larson, and Erland Jonsson, "Efficient In-Vehicle Delayed Data Authentication Based on Compound Message Authentication Codes," 2008 IEEE 68th Vehicular Technology Conference (VTC), pp. 1-5, Sep. 2008.

[9] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," Nature, vol. 323, pp.533–536, 1986.

[10] J. L. Elman, "Finding Structure in Rime." Cognitive Science, vol.14, pp.179-211, 1990.

[11] A. Graves, A. Mohamed, and G. Hinton, "Speech Recognition with Deep Recurrent Neural Networks," ICSSP2013, pp.6645-6649, 2013.

[12] G. Mesnil, X. He, L. Deng, and Y. Bengio, "Investigation of Recurrent-Neural-Network Architectures and Learning Methods for Spoken Language Understanding," Interspeech, pp. 3771-3775, 2013.

[13] https://chainer.org/

[14] D. Kingma, and J. Ba, "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980, 2014.