NETTED: Non-Standard CAN ID Extraction System Based on Deep Neural Networks

Beomjun Kim, Yunju Baek
Department of Electrical and Computer Engineering
Pusan National University
Busan, Republic of Korea
e-mail: beomjun.kim@eslab.re.kr, yunju@pusan.ac.kr

***Abstract*—With increased interest in smart driving technology, interest in understanding the current state of the vehicle using CAN data has also increased. Standardized CAN information such as vehicle speed and engine RPM information can be easily acquired and used. However, most of the CAN data related to the driving information such as the angle of the steering wheel is not standardized. To solve this problem, this paper proposes and implements a system for extracting non-standard CAN IDs based on deep neural networks. The system can estimate the current state of the steering wheel and brake pedal by learning data collected from GPS and IMU. Then, the appropriate CAN ID and byte position are extracted based on the estimated current state information. Experimental results show that the estimation accuracy of the steering wheel and brake pedal are 87.63% and 94.81%, respectively. Also, it is confirmed that CAN data area is reduced by about 90% through the pre-filtering process. Finally, the proposed system extracts two desired CAN IDs.**

Keywords-controller area network; deep neural network; reverse engineering; embedded system; driving information

I. INTRODUCTION

As the number of vehicles continues to increase and the road environment becomes complicated, interest in smart vehicle technology for enhancing driver safety and convenience is increasing. Typically, study on smart vehicle technology includes autonomous vehicles, advanced driver-assistance systems (ADAS), and automatic driver maneuver recognition.

ADAS can help the driver to drive conveniently and safely. Examples of ADAS include the ability to sound a beep when the driver is drowsy or to indicate when a collision with a preceding vehicle is expected. It also recognizes and informs the driver when the lane departs unintentionally without turning on the turn signal lamp.

Driver maneuver recognition identifies the physical movement of a vehicle as the driver drives the vehicle on the road. When the driver's maneuver is recognized, the system can determine whether the vehicle is in a straight-ahead, lane-changing, left-turn, right-turn or U-turn state. If the vehicle can distinguish the driver's maneuver, information about the surrounding environment can be considered together to prevent accidents caused by inexperience in driving.

Among these studies, there are many studies about controller area network (CAN) for determine the current state of the vehicle or the behavior of the driver. For example, there are studies to recognize the lane change or turn of the vehicle on the road [1-4], and to recognize the drowsiness driving [5, 6] or recognize the driver's fatigue [7]. There are also studies to evaluate whether the driving style is safe [8-10] or to distinguish the driver [11, 12]. However, most of the useful CAN data is difficult to use because the analysis method is not disclosed. Further, since the meaning of the CAN ID used varies depending on the manufacturer or model of the vehicle, the ID acquired from one vehicle cannot be used in another vehicle. Only some data are standardized and published. Therefore, it is difficult for general users to obtain the vehicle status information without cooperation of the vehicle manufacturer.

There are also studies that use additional sensors to obtain information about the vehicle to avoid this problem. For example, there is a study to measure the fatigue by continuously acquiring the steering angle of the vehicle by attaching the sensor to the steering wheel [13]. Also, there are studies that adopt sensors of smartphone to extract driving behavior of vehicle [14, 15]. However, there is a limitation in estimating the current state of the vehicle based on additional sensors, and it is dangerous to apply inaccurate information to the safety-related applications of the driver.

To interpret CAN information not disclosed by the vehicle manufacturer, there are studies that reverse engineer vehicle information through CAN network. At the beginning of the study, there were methods to distinguish ID and data by direct communication by separating ECU from vehicle [16-18]. However, the method of direct communication with the ECU takes a long time due to the disassembly and assembly of the vehicle. In addition, since it is difficult to disassemble all types of vehicles, there is an attempt to analyze the CAN network using only sensor data and CAN information obtained from the OBD port of the vehicle [19]. However, this study also has limitations in extracting only vehicle speed or wheel speed. The most commonly used information for driver maneuver recognition or driver classification is the steering wheel and brake pedal information. Therefore, to activate various smart automobile technologies, it is necessary to collect CAN data and extract useful information not defined as standard.

In this paper, we propose a system that learns deep neural network based on sensor data and extracts non-standard

978-1-5386-9273-8/18/$31.00

CAN ID from collected CAN data to solve this problem. The collection device attached to the vehicle is configured to sniff CAN data from the OBD port of the vehicle while collecting sensor information of IMU and GPS. We also propose a method called NETTED, which extracts non-standard ID using the collected data.

The remainder of this paper is organized as follows. Section II briefly describes the operation and design of the proposed system. Section III explains NETTED, a technique for extracting the CAN ID used by the proposed system. This technique estimates driving information based on the sensor data, matches it with the refined CAN data, and finally extracts the CAN ID. Finally, we evaluate the performance of the proposed system through experiments and conclude.

II. System Design

We designed a system that can extract non-standard information from collected data from vehicles. The system can extract the CAN ID and byte position of the steering angle and brake pedal, which can be differentiated by vehicle manufacturer and vehicle type, through NETTED (Non-standard information Extraction Technique with Tailoring ECU Data). The extraction process of the proposed system is divided into two parts. The first part is to estimate the driving information of the vehicle using sensor data. Before performing the inference process, the system learns the driving information by using the label generated based on the reference video acquired from the camera with sensor data as an input. The system estimates the driving information of the vehicle by using the sensor data coming after the learning. The second part extracts the ID using the estimated driving information and the data collected from the CAN bus. The CAN data acquired on the CAN bus is classified by ID, and then the data area having a low probability of being the desired information is removed through a pre-filtering process. After confirming the ID list with high relevance to the driving information, the system performs data trimming process and arranges the list in the order of the id having high relevance. The operation structure of the proposed system is shown in Figure 1.

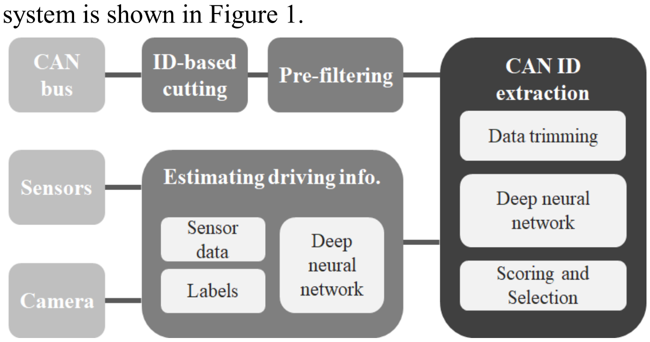

Figure 1. Structure of proposed system

III. NETTED

A. Sensor-Based Estimation of Driving Information

The proposed system estimates two driving information based on sensor data. As the driver drives, these driving information change very quickly and often. The ability to estimate the driving information based on the sensor data can reduce the load of generating the manual label while watching the video for all collected data. The angle of the steering wheel is estimated using the IMU, and the brake pedal is estimated using the speed of the GPS.

The process for estimating the steering wheel is as follows. Using the video of the front camera, a manual label of steering wheel is generated by determining whether the traveling direction of the vehicle is forward, left, or right. Acceleration values and gyro values collected from IMU were applied to Butterworth filter to remove noise. After that, we used the collected data to estimate the driving information by learning the deep neural network that has the filtered data as the input and the label for the steering wheel as the output.

Similarly, to estimate the brake pedal, it was labeled whether the driver depressed the brake pedal. Through the learning of the deep neural network, by inputting the speed that can be obtained through GPS, it is possible to obtain driving information about whether or not the brake pedal is depressed.

B. Pre-filtering

In the proposed system, a prefiltering process is performed to remove the data area which is not likely to be related to the information to be extracted. For example, the CAN data about the steering wheel changes continuously and linearly in the course of driving. Thus, when the value does not change or has a very small number of states, and the nonlinearly changing data is highly unrelated to the steering wheel.

The CAN data of brake pedal also indicates how far the driver has stepped on the pedal or indicates whether the pedal is depressed. This can be used to exclude data whose values do not change at all. In addition, there is a high probability that the data having two or three states or changing linearly are related to the brake pedal. Therefore, data not corresponding to this case can be excluded. In this pre-filtering process, the data area for finding the target value can be greatly reduced.

C. Tailoring Scheme for ECU Data

In this paper, ID-based cutting and Data trimming are methods for tailoring data provided by ECUs via CAN network. ID-based cutting is the separation of collected CAN data by CAN ID. This process facilitates the pre-filtering process and is essential for the process of learning and scoring by ID in the non-standard information extraction process based on the deep learning that will be performed later.

Data trimming can be used in two ways. The first method is post-learning trimming that confirms associativity by initializing certain bytes to zero in the learned network. The second method is pre-learning trimming that identifies associations by leaving only certain bytes prior to learning. By fixing the data used in the network estimating the driving information, it is possible to check the influence on the performance of the network according to the position.

D. Extraction of Non-standard Information

Extraction of non-standard information is performed in a deep neural network that has CAN data separated by ID-based cutting as inputs and sensor-based driving information as outputs. The following process is performed to extract the data most relevant to the target information from the data that passed the pre-filtering process.

First, a deep neural network is learned by using data having only one CAN ID through an ID-based cutting process. By learning multiple networks, the estimation accuracies for each ID is collected. Extracts a list of the top 10 CAN IDs having the highest estimation accuracy among all IDs that pass the pre-filtering process. Since the data with the highest estimation accuracy is the most likely to be the target data, information for scoring is collected for these data. Then, a post-learning trimming process is performed on the input data, and a change in the estimation accuracy is stored. Most of the data will decrease in estimation accuracy and will vary a lot depending on which byte is trimmed. If data trimming is performed after learning, the accuracy of several bytes will vary greatly due to the characteristics of the deep neural network. For more detailed scoring, pre-learning trimming is performed on the candidates with similar scores. The system also stores the estimation accuracy when performing pre-learning trimming. Finally, a final score is calculated for candidate groups generated using CAN IDs and byte location. The score of the i-th ID is a result of weighted sum of accuracy, the differences in pre-learning trimming and post-learning trimming.

IV. IMPLEMENTATION OF THE SYSTEM

We built a data acquisition device based on Nvidia's Jetson TX2 to implement the system. IMU, GPS, and CAN transceiver were added to the developer's hardware, and a program for logging data was created. The IMU was used to collect real time data at 125 Hz and the GPS speed was collected every second. In addition, the USB camera connected to the device has taken video for use as reference data. The source code that processes the sensor data and constructs the deep neural network is all based on python. Deep neural networks were constructed using Keras with Tensorflow as a backend and their performance was evaluated. The photographs and components of the implemented system are shown in Figure 2. Neural networks use a simple structure to operate on embedded devices. LSTM was used for state inference using sensor data, and DNN was used for evaluation of each CAN data. Figure 3 shows the behavior of a neural network.

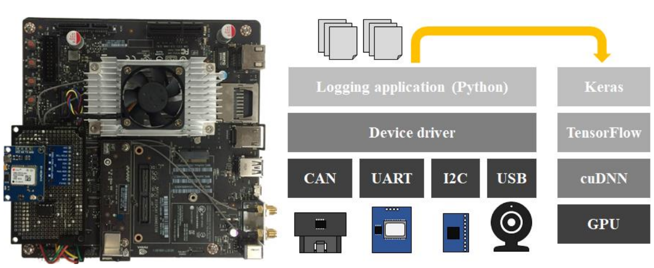

Figure 2. Data collection device of the proposed system

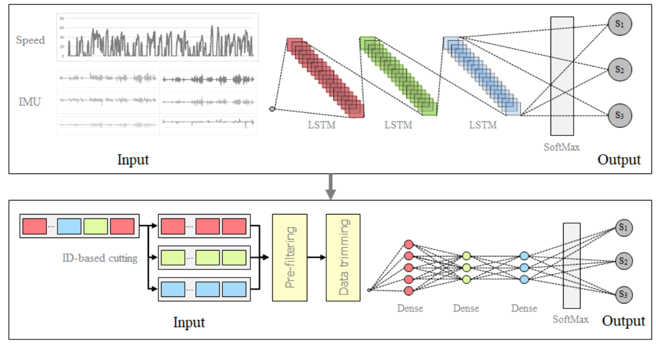

Figure 3. CAN ID extraction process using neural networks

V. EXPERIMENTS AND RESULTS

A. Dataset

We collected the data through the actual vehicle and evaluated the performance of the implemented system based on these data. Collected data in this paper were obtained from Renault Samsung SM6 (2017), Hyundai Grandeur HG (2015) and Kia Carnival (2017). The dataset used in the experiment was collected by repeatedly driving the course shown in Figure 4. As shown in the figure, it consists of courses like actual driving including left turn, right turn, and U-turn. Data were collected from three vehicles. The information collected was composed of video, sensor data, and CAN data. The total length of the collected datasets is about 3 hours, and the number of CAN data collected while driving is about 16.2 million. In addition, the total driving distance is 54 km since 9 courses of about 6 km were collected during the collection process.

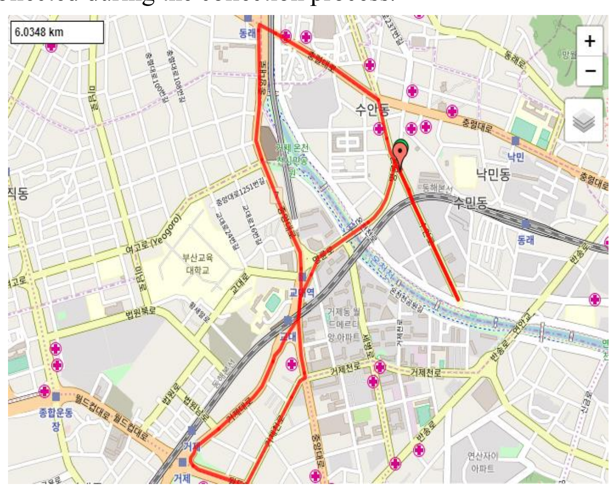

Figure 4. Driving route for dataset collection (Busan, South Korea)

B. Estimation Accuracy of Driving Information

First experiments were performed to evaluate the performance of estimating the driving information based on the sensor data. Figure 5-(a) shows the system performance when the IMU is used to distinguish the steering wheel state from the left, center, and right. Accuracy was 91.22%, 86.84%, and 84.83% for each vehicle. Using this technique, by inferring the driving information based on the sensor, it is possible to reduce the number of manual labeling by a person. Therefore, we estimated the steering wheel for the remaining two vehicles by using the same network as the first vehicle. The position and angle of the data acquisition device vary slightly depending on the vehicle, and the performance is slightly reduced because the shaking pattern appears different during driving.

Next, the performance of estimating the brake pedal based on the GPS speed is shown in Figure 5-(b). When the window size was 8, the estimated success rates of the vehicles were 94.67%, 94.57%, and 95.19%, respectively.

Likewise, we estimated the performance of the remaining two vehicles using the network learned from the first vehicle. The speed information of the GPS is not affected by the type of the vehicle or the location of the data collecting device, so that the performance is not decreased.

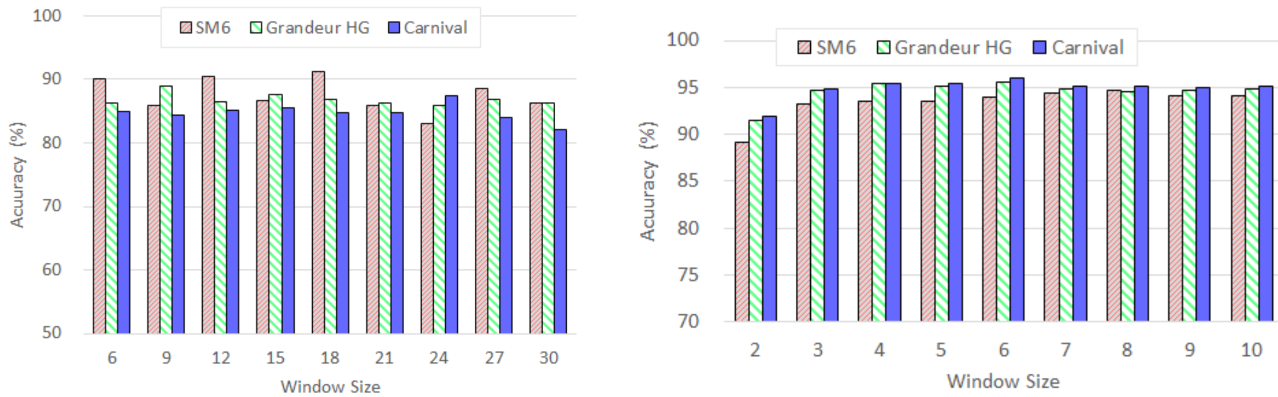

Figure 5. Performance of the estimating: (a) IMU-based steering wheel, (b) brake pedal depression based on GPS speed

C. Effect of Pre-filtering

To check the effectiveness of the pre-filtering process, we have confirmed how much the data area shrinks after execution. The performance of this experiment is shown in Figure 6.

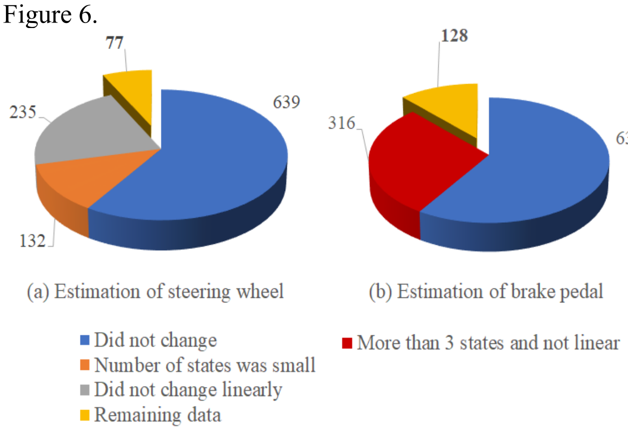

Figure 6. The number of excluded and remaining data by performing a prefiltering process

First, the data that were excluded because the values did not change during the estimation of the steering wheel were 639, which accounted for 59.0%. Since the number of states was small, the number of excluded data was 132, which accounted for 12.19%. In addition, 235 cases that did not change linearly accounted for 21.7%. In sum, the data excluded through pre-filtering in the estimation of the steering wheel is 92.89% of the total.

In the estimation process of the brake pedal, 639 pieces of data are also excluded because the data is not changed. In addition, 316 cases were excluded because the number of states was more than 3 and not linear, accounting for 29.18%. In the process of estimating the brake pedal, 88.18% of the total data is excluded through pre-filtering.

Since it is possible to exclude unimportant data areas, it has been confirmed that the pre-filtering can reduce the size of the candidate data area. Since it is not scored if it does not pass pre-filtering, it is possible to prevent the inappropriate ID and byte position from being selected as the extraction result.

D. Extraction Performance of Non-standard Information

Through experiments, we confirmed that the proposed system works as expected by checking the performance of non-standard information extraction. Excluding IDs from pre-filtering out of the list with high estimation accuracy, on average, fewer than 10 ID and byte position pairs remain. When trying to extract the IDs for the steering wheel from the three vehicles, the correct IDs were the highest score in all cases. Since pre-learning trimming takes longer than post-learning trimming, pre-learning trimming is not performed for candidates that score lower than other candidates even if they receive a good score in pre-learning trimming process. In order to evaluate the performance of the system, it was confirmed that the extracted ID and byte position pairs were correctly extracted by monitoring the vehicle while driving.

VI. Conclusion

In this paper, we propose and implement a non-standard CAN ID extraction system that is difficult to use because it differs depending on vehicle manufacturer and vehicle model. The proposed system estimates driving information by using sensor data and CAN data collected from actual vehicle and extracts the final ID by checking the association between the corresponding information and CAN data through deep learning. We have confirmed through the performance evaluation that the system estimates the driving information appropriately. In addition, we implemented a pre-filtering method that greatly reduces CAN data candidates by reflecting the characteristics of the information to be extracted. Also, we confirmed the system can successfully extract the target ID for all collected datasets.

ACKNOWLEDGMENT

This work was supported by the Center for Integrated Smart Sensors funded by the Ministry of Science and ICT as Global Frontier Project" (CISS-2011-0031863)

REFERENCES

[1] A. Sathyanarayana, S. O. Sadjadi, and J. H. Hansen, “Leveraging Sensor Information from Portable Devices towards Automatic Driving Maneuver Recognition,” 15th International IEEE Conference on Intelligent Transportation Systems (ITSC 2012), Sep. 2012, pp. 660-665, doi:10.1109/ITSC.2012.6338717.

[2] V. Leonhardt and G. Wanielik, “Feature Evaluation for Lane Change Prediction based on Driving Situation and Driver Behavior,” 20th International Conference on Information Fusion (ICIF 2017), August 2017, pp. 1-7, doi: 10.23919/ICIF.2017.8009848.

[3] Y. Zheng and J. H. Hansen, “Lane-Change Detection From Steering Signal Using Spectral Segmentation and Learning-Based Classification,” IEEE Transactions on Intelligent Vehicles, vol. 2, March 2017, pp. 14-24, doi:10.1109/TIV.2017.2708600.

[4] V. Leonhardt and G. Wanielik, “Recognition of Lane Change Intentions Fusing Features of Driving Situation, Driver Behavior, and Vehicle Movement by Means of Neural Networks,” Advanced Microsystems for Automotive Applications, August 2017, pp. 59-69, doi:10.1007/978-3-319-66972-4_6.

[5] K. Vasudevan, A. P. Das, B. Sandhya and P. Subith, “Driver Drowsiness Monitoring by Learning Vehicle Telemetry Data,” 10th International Conference on Human System Interactions (HIS 2017), July 2017, pp. 270-276, doi:10.1109/HSI.2017.8005044.

[6] S. Moon, M. Min, J. Nam, J. Park, D. Lee amd D. Kim, “Drowsy Driving Warning System Based on GS1 Standards with Machine Learning,” IEEE International Congress on Big Data, June 2017, pp. 289-296, doi:10.1109/BigDataCongress.2017.44.

[7] Y. Zhang, M. Chen, N. Guizani, D. Wu, and V. C. Leung, “SOVCAN: Safety-Oriented Vehicular Controller Area Network,” IEEE Communications Magazine, vol. 55, August 2017, pp. 94-99, doi:10.1109/MCOM.2017.1601185.

[8] S. H. Chen, J. S. Pan, and K. Lu, “Driving Behavior Analysis Based on Vehicle OBD Information and AdaBoost Algorithms,” International MultiConference of Engineers and Computer Scientists (IMECS 2015), March 2015, pp. 18-20.

[9] J. E. Meseguer, C. K. Toh, C. T. Calafate, J. C. Cano, and P. Manzoni, “Drivingstyles: a mobile platform for driving styles and fuel consumption characterization,” Journal of Communications and Networks, vol. 19, April 2017, pp. 162-168, doi:10.1109/JCN.2017.000025.

[10] F. Martinelli, F. Mercaldo, A. Orlando, V. Nardone, A. Santone, A. K. Sangaiah, “Human behavior characterization for driving style recognition in vehicle system,” Computers and Electrical Engineering, in press.

[11] B. I. Kwak, J. Woo and H. K. Kim, “Know your master: Driver profiling-based anti-theft method,” 14th Annual Conference on Privacy, Security and Trust (PST), Dec. 2016, pp. 211-218, doi:10.1109/PST.2016.7906929.

[12] M. Enev, A. Takakuwa, K. Koscher, and T. Kohno, “Automobile Driver Fingerprinting,” Proceedings on Privacy Enhancing Technologies, vol. 2016, Jan. 2016, pp. 34-50, doi:10.1515/popets-2015-0029.

[13] Z. Li, L. Chen, J. Peng and Y. Wu, “Automatic Detection of Driver Fatigue Using Driving Operation Information for Transportation Safety,” Sensors, vol. 17, May 2017, pp. 1212, doi:10.3390/s17061212.

[14] X. Liu, H. Mei, H. Lu, H. Kuang and X. Ma, “A Vehicle Steering Recognition System Based on Low-Cost Smartphone Sensors,” Sensors, vol. 17, Mar 2017, pp. 633, doi:10.3390/s17030633.

[15] S. Hernández Sánchez, R. Fernández Pozo and L. Hernández Gómez, “Estimating Vehicle Movement Direction from Smartphone Accelerometers Using Deep Neural Networks,” Sensors, vol. 18, August 2018, pp. 2624, doi:10.3390/s18082624.

[16] T. Hermans, J. Denil, P. De Meulenaere and J. Anthonis, “Decoding of data on a CAN powertrain network,” 16th Annual Symposium on Communication and Vehicular Technology in the Benelux, vol. 6, 2009.

[17] K. Koscher, A. Czeskis, F. Roesner, S. Patel, and T. Kohno, “Experimental Security Analysis of a Modern Automobile,” IEEE Symposium on Security and Privacy, May. 2010, pp. 447-462, doi:10.1109/SP.2010.34.

[18] S. Freiberger, M. Albrecht and J. Käufl, “Reverse Engineering Technologies for Remanufacturing of Automotive Systems Communicating via CAN Bus,” Journal of Remanufacturing, Dec. 2011, pp. 6, doi:10.1186/2210-4690-1-6.

[19] T. Huybrechts, Y. Vanommeslaeghe, D. Blontrock, G. Van Barel, and P. Hellinckx, “Automatic Reverse Engineering of CAN Bus Data Using Machine Learning Techniques,” International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, Nov. 2017, pp. 751-761, doi:10.1007/978-3-319-69835-9_71.