IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 29, 2021, accepted June 30, 2021, date of publication July 9, 2021, date of current version July 20, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3095962

Comparative Performance Evaluation of Intrusion Detection Based on Machine Learning in In-Vehicle Controller Area Network Bus

TAREK MOULAHI[1,2], SALAH ZIDI[3,4], ABDULATIF ALABDULATIF[5], (Member, IEEE), AND MOHAMMED ATIQUZZAMAN[6], (Senior Member, IEEE)

[1]Department of Information Technology, College of Computer, Qassim University, Buraydah 52571, Saudi Arabia
[2]FSTSB, Kairouan University, Kairouan 3100, Tunisia
[3]Department of Management Information System, College of Business and Economics, Qassim University, Buraydah 52571, Saudi Arabia
[4]Hatem Bettaher Laboratory, IRESCOMATH, Gabes University, Gabes 6029, Tunisia
[5]Department of Computer Science, College of Computer, Qassim University, Buraydah 52571, Saudi Arabia
[6]School of Computer Science, The University of Oklahoma, Norman, OK 73019, USA

Corresponding author: Abdulatif Alabdulatif (ab.alabdulatif@qu.edu.sa)

This work was supported by the Deanship of Scientific Research, Qassim University.

ABSTRACT Communication between the nodes in a vehicle is performed using many protocols. The most common of these is known as the Controller Area Network (CAN). The functionality of the CAN protocol is based on sending messages from one node to all others throughout a bus. Messages are sent without either source or destination addresses. Consequently, it is simple for an attacker to inject malicious messages. This may lead to some nodes malfunctioning or total system failure, which can affect the safety of the driver as well as the vehicle. Detecting intrusions is a challenging problem in the context of using CAN bus for in-vehicle communication. Most existing work focuses on the physical aspects without taking into consideration the data itself. Machine Learning (ML) tools, especially classification techniques, have been widely used to address similar problems. In this paper, we use and compare several ML techniques to deal with the problem of detecting intrusions in in-vehicle communication. An experimental study is performed using a real dataset extracted from a KIA Soul car. Compared to previous work, which focuses on detecting intrusions based on the physical aspect, this paper aims to concentrate on the application of data analysis and statistical learning techniques. Furthermore, the paper provides a comparative study of the most common ML techniques. The results show that the techniques under consideration in this paper outperform other techniques that have been used previously.

INDEX TERMS CAN bus, data classification, intrusion detection, in-vehicle communication, machine learning.

I. INTRODUCTION

Recently, a considerable amount of research has focused on vehicle communication technology, such as smart vehicles, Vehicular ad hoc Networks (VANET) [1], [2], and Intelligent Transportation Systems (ITS). Vehicles are necessary for daily life, and they are becoming more electronically equipped and are on longer simple mechanical machines. Electronic Control Units (ECUs) are used in vehicles to monitor and control different components. ECUs are connected through buses managed by several protocols [3], [4]. A vehicle bus is an intravehicular communication network that does not have a host computer. A bus is used to link a set of ECUs to simplify the task of exchanging messages as well as diagnostics. Intravehicular networks have many advantages [5], including (1) reducing the cable budget, which is the third most costly system after the engine and the chassis; (2) minimizing the packaging space by using fewer connections for more electrical and electronic features, allowing a reduction in vehicle size; (3) meeting higher bandwidth demands that can manage the large number of ECUs, with some vehicles containing up to 70 ECUs with 2500 internal signals [5]; and (4) making communication more reliable because bus-based communication is more robust than the traditional point-to-point communication in older vehicles.

The associate editor coordinating the review of this manuscript and approving it for publication was Gustavo Callico.

Currently, the most widely used protocols for in-vehicle communication are [3]:

- Local Interconnection Networks (LIN),
- Controller Area Networks (CAN),
- FlexRay,
- Ethernet,
- Media-Oriented Systems Transport.

All these protocols are based on bus communication, and each of them has certain advantages and weakness compared to the others. Among these protocols, we have chosen the CAN bus protocol, developed by Bosch in 1985 [6]. This protocol is used in the majority of vehicles today. Approximately 500 million CAN chips are used in vehicles [5]. In addition, a recent study predicted that the CAN bus will maintain its prosperity for the next decade [5]. The CAN bus is the leading technology due to its low cost compared to other protocols, the maximum bit rate for high-speed CAN is 1 $Mbit/s$ by specification, and its acceptable fault tolerance behavior relative to the other intravehicular communication protocols mentioned earlier.

Despite its advantages, CAN bus suffers from many vulnerabilities. The main problem is that a CAN lacks any kind of security mechanism because it was not considered in its design [7]. Attacks on a CAN bus can come from outside, particularly from the On-Board Diagnostics (OBD) [8], or from other wireless interfaces, such as cellular links, Wi-Fi, and Bluetooth [5], [9]. Figure. 1 illustrates a combination of attack types, attack surfaces, and vulnerable assets.

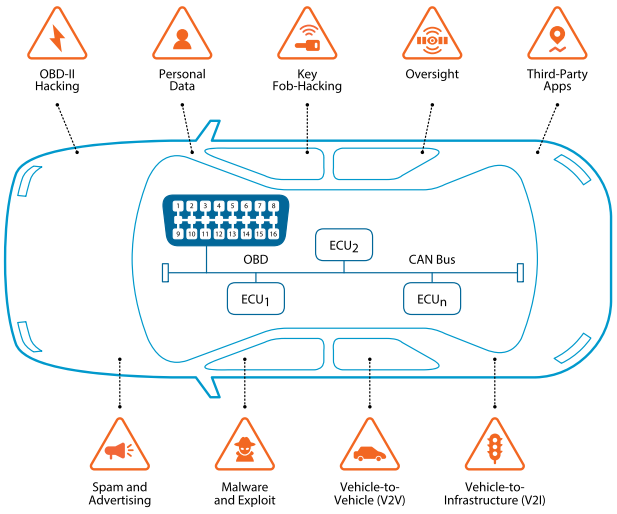

FIGURE 1. Modern cars are exposed to various types of attacks on the CAN bus from external devices connected to the car, particularly from OBD.

The first type of attack includes frame falsifying, sniffing, and relay attacks, which can be addressed by encryption and improving authentication. The second type includes impersonation, Denial of Service (DoS), and fuzzy attacks, which must be treated by developing an Intrusion Detection System (IDS) to distinguish between normal behavior and an attack.

Most of the previous research dealing with security problems in the CAN protocol have been concentrated on physical aspects, such as limiting physical access or using cryptography to protect CAN transmission [10]. However, there is still a need to achieve better IDS. Indeed, the limitation of physical access will affect the effectiveness of transmission in CAN bus. Cryptography is not always suitable with such a lightweight system. This will be discussed in detail in the related work section.

Over the last decade, Artificial Intelligence (AI) tools have generated interesting and effective results in solving complex problems that resemble ours, such as automatic system diagnostics and identification [11], fault detection in wireless sensor networks [12]–[16], and certain security problems in other fields. Thus, ML techniques, as the most interesting approach in the field of AI, can be very effective for the detection of intrusions. There are three ML models for prediction: (1) the regression model, (2) the classification model, and (3) the clustering model. For real-time or predictive intrusion detection, classification-based model or clustering-based model are applied where the former is used in the case of a supervised problem and the later is considered in the case of a non-supervised problem.

The objective of this paper is to comparatively study intrusion detection systems based on different ML models. For that, Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), and MultiLayer Perceptron (MLP) have been used to enhance other models applied recently with the same dataset. Unlike previous studies, we perform intrusion detection by attack type on the KIA Soul dataset as one of the comparison criteria where we consider three type of attacks, including DoS, impersonation, and fuzzy attacks.

Security was not considered when bus-based CANs were designed in the 1980s [9]. However, most modern vehicles use bus-based CANs, which is a non-secure network that can be hacked by injecting faulty messages. Consequently, attacks can cause accidents, possibly resulting in injury or death. This makes protecting CAN bus-based network a high priority in order to ensure the safety of drivers and passengers. While existing research works have used ML models to deal with this challenging problem, they appear to be insufficient and can be enhanced by using other ML models. This motivates us to explore the capabilities of other advanced ML techniques, such as SVM, DT, RF, and MLP to overcome the existing security concerns with in-vehicle CAN buses.

The main objectives of the paper are as follows:

- Develop intrusion detection-based ML in In-Vehicle controller area network bus through applying various ML techniques in the context of in-vehicle CAN bus networks as an IDS.
- Conduct a comparative performance evaluation of applied ML for intrusion detection in an in-vehicle CAN bus using a set of classifiers on a real dataset that includes messages transmitted using a CAN bus extracted from a KIA Soul car [6].

- Detect both the intrusion and the attack type: DoS, impersonation or fuzzy attack.

To the best of our knowledge, this is the first time that RF, DT, SVM, and MLP are applied with the KIA Soul dataset. The results of our experimental study show that RF not only outperforms SVM and DT but also the other classifiers (Hierarchical Temporal Memory (HTM), Recurrent Neural Networks (RNN), and Hidden Markov Models (HMM)) previously used in the same context.

The rest of this paper is organized as follows: Section 2 outlines the related work. In Section 3, a review of the classifiers used for intrusion detection is given. The experimental study and a discussion of the results are presented in Section 4. Finally, Section 5 concludes the paper.

II. RELATED WORK

Protecting communication inside vehicles is very important since it affects the safety of vehicles as well as that of their drivers and passengers. Achieving this task in the CAN protocol is challenging due to the shortcomings of CANs, which are vulnerable to many types of attack, including DoS, impersonation, and fuzzy attacks. This makes developing an IDS for this type of network an attractive problem for the research community. Indeed, much research has been undertaken to deal with this problem. In the following, we discuss the most relevant research investigating IDS for intravehicular communication.

In [6], the authors proposed using an analysis of the offset ratio and the time interval between the request and the response; i.e., working on a remote frame and data frame to create an IDS. Analysis of the response performance of ECUs helps to decide if a behavior is an attack (i.e., intrusion detection) or a normal behavior. The authors treated three types of attacks: DoS, fuzzy, and impersonation attacks in CAN-based networks. Some results showed that this approach is very encouraging. However, a metric like accuracy of attack detection is not given to determine whether or not the proposed approach achieved the best detection performance.

Groza and Murvay [8] proposed a bloom filtering-based IDS. A bloom filter is a probabilistic structure for testing whether an item belongs in a set. There are no false negatives with this filter, providing a 100% recall rate. The authors used this filtering method based on frame identifiers and part of the data fields to test frame periodicity, as it facilitates the detection of frame modification attacks or possible replays. The authors tested their contribution with a CAN bus; however, this approach can also be used with other types of in-vehicle communication. The disadvantage of this approach is that the authors that the compare their method with other methods. Furthermore, they included an important overload on ECU, which could affect their time response.

Tariq *et al.* [17] used RNNs and heuristics to detect attacks, employing the same dataset as [6] used in their study. The detection dealt with three types of attacks: DoS, replay, and fuzzy attacks. The authors used both neural networks and network traffic signatures. The accuracy of intrusion detection was high; however, these authors did not did not propose a technique for unseen attacks.

Neural networks are also used for intrusion detection in CANs in [18]. This study reported good results despite some weaknesses. For example, the detection of replay attacks was not adequate due to the high degree of similarity between genuine frames and injected frames, which makes the time stamp very useful in this case. Globally, the use of neural networks as IDS in CANs is promising and provides satisfactory results while still providing CAN bus communication safety.

A Deep Neural Network (DNN) was used in a novel technique for intrusion detection in CANs [19]. The authors used deep learning techniques to distinguish between normal behavior and attacks. The comparison between DNN-based IDS and standard neural networks shows that a DNN is better in terms of improving detection accuracy with a real-time response.

Wu *et al.* [20] proposed a novel intrusion detection method based on the information entropy method. This approach uses sliding windows with fixed numbers of messages. The authors show that optimization of the decision conditions and enhancement of the sliding windows help to improve intrusion detection accuracy while decreasing the false positive rate. Furthermore, the effectiveness of the proposed method was demonstrated in an experimental study providing real-time responses to intrusion with important detection precision. Despite promising results, the authors did not consider the impact of the vehicle operation state on information entropy.

Wang *et al.* [21] used the benefits of hierarchical temporal memory (HTM) to define a distributed anomaly IDS in a CAN-based in-vehicle network. The proposed technique predicts data flow depending on previous state learning in real time. Through an experimental study, the authors showed that HTM outperforms other detection models based on neural networks and HMMs in terms of detection accuracy.

A practical security architecture for a CAN-FD (which is designed to deal with the CAN bandwidth limitations)-based network is defined in [22]. The effectiveness of the proposed architecture was tested on three kinds of microcontrollers. This technique could be considered for use in vehicles manufactured in the future.

Despite the fact that a considerable amount of research has been focused on developing an IDS in CAN-based networks, there is still a need to achieve better systems. Most of the previous work has examined the behavior of exchanged frames or uses the data in the frames only superficially. In addition, traditional classification techniques are not used. The aim of this paper is to mine the data within the exchanged frames deeply and take advantage of the benefits of different classifier methods to define a smart IDS for CANs that is able to detect attacks in real time in order to protect vehicles as well as their drivers and passengers.

III. CLASSIFICATION MODELS FOR INTRUSION DETECTION SYSTEMS

We have applied three ML techniques for intrusion detection. Intrusion detection is a supervised classification problem, as we can use a known dataset containing labeled data. The four approaches tested to solve this problem are SVM, DT, RF, and MLP.

In this section, the problem statement is outlined. Next, the four classification techniques used and the evaluation criteria are defined. Finally, the experimental results are given.

A. PROBLEM STATEMENT

Many research studies have dealt with the problem of intrusion detection using experimental approaches and published datasets [6]. In this study, a set of classification techniques are used for intrusion detection in same dataset. The dataset contains three types of attacks, DoS, fuzzy, and impersonation attacks. This dataset was created by injecting messages through the OBD-II port in real CAN traffic belonging to a KIA Soul car.

The data is prepared as shown in Table 1, describing the list of features. The results of applying RF, SVM, and DT will be compared to the latest research studies [21] investigating the same dataset.

Three types of attacks are treated:

- DoS attack:
  This attack occurs when messages with high priority are injected into the CAN bus. The aim of this attack is to occupy the bus with packets carrying identifiers with high priority. This attack is done by the injection of packet 0×000 CAN ID in a short cycle inside the traffic.
- Impersonation attack:
  This attack occurs when an attacker creates an impersonating node for answering remote frames. Thus, data frames will be broadcast periodically by the impersonating node to respond as a target node for remote frames. This attack is performed by inserting packets coming from impersonating node, with an arbitration ID = "0×164".
- Fuzzy attack:
  This attack happens when packets of randomly spoofed identifiers with arbitrary data are injected by an adversary. Consequently, many functional packets will be received by all nodes, which may result in unintended vehicle responses. Hence, fuzzy attacks can completely prevent any bus communication or the transmission of certain frames through charging the CAN bus, as happens in a DoS attack. To conduct a fuzzy attack, packets are injected with spoofed random CAN ID and DATA values.
  Most known DoS attacks on CANs do not merely delay legit frame transmission but completely prevent any bus communication or the transmission of certain frames.

B. SUPPORT VECTOR MACHINES (SVM) CLASSIFIER

SVM [23]–[25] is a statistical learning technique. It consists of a determination of decision boundaries. It is a supervised classification technique using a set of labeled examples and is based on the calculation of a learning model that can be generalized. As shown in Figure. 2, SVMs can efficiently perform non-linear as well as linear classification. For the non-linear model, this technique uses kernel functions.

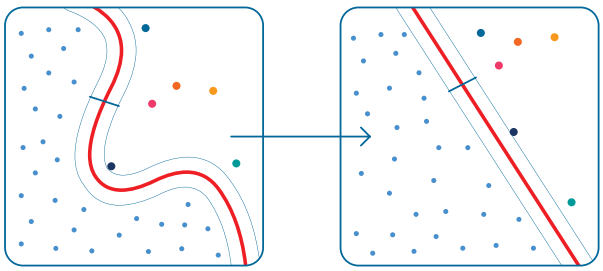

FIGURE 2. The SVM classification model is a supervised technique aiming to find a decision boundary based on a set of labeled samples by calculating a learning model that can be generalized.

C. DECISION TREES (DT) CLASSIFIER

A DT is a decision support tool based on the representation of the choices in the graphical form of a tree with the different classification decisions placed in sheets [26]. This technique uses a hierarchical representation of the data structure in the form of decision sequences (tests) for the result prediction class. Each observation, which must be assigned to a class, is described by a set of variables that are tested in the tree nodes. Tests are performed in internal nodes, and decisions are made in leaf nodes.

To explain the principle of this tool, we consider the classification problem. Each element x of the database is represented by a multidimensional vector $(x_1, x_2, \ldots x_n)$ corresponding to the set of descriptive variables of the point. Each internal node of the tree corresponds to a test performed on one of the variables xi. Once the tree has been built, classifying a new candidate is done by going down the tree, from the root to one of the leaves (which encodes the decision or class). At each level of the descent, we pass an intermediate node where a variable xi is tested to decide which path (or subtree) to choose to continue the descent. To build the tree, the learning base points are all placed in the root node. One of the variables describing the points is the class of the point (the "ground truth"); this variable is called the "target variable". The target variable can be categorical (classification problem) or a real value (regression problem). Each node is cut (split operation), giving rise to several descending nodes. An element of the learning base located in a node will be found in only one of its descendants.

- The tree is built by recursive partition (see Figure. 5) of each node according to the attribute value tested in each iteration (top-down induction). The optimized criterion is the homogeneity of the descendants compared to the target variable. The variable that is tested in a node will be the one that maximizes this homogeneity.

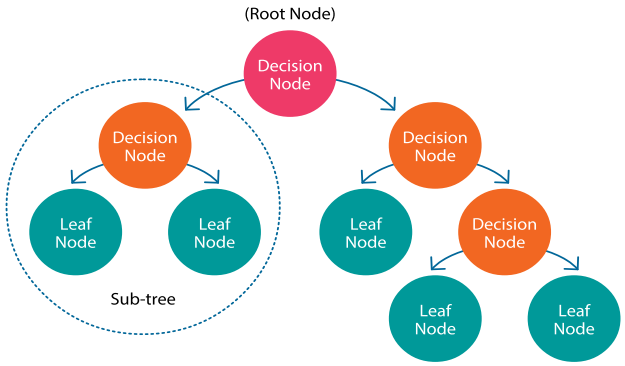

FIGURE 3. DT is a decision tool based on the representation of the choices in the graphical form of a tree with the different classification decisions, where the learning base points are all placed in the root node.

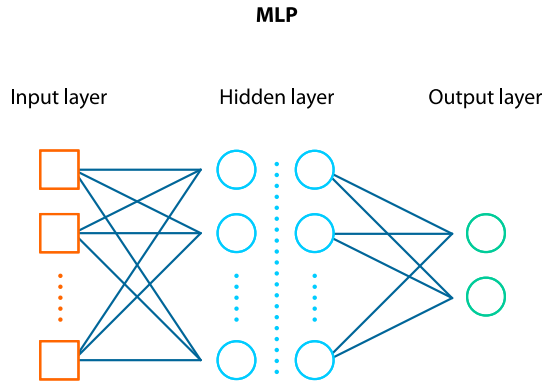

FIGURE 5. A simple illustration of MLP as a neural network learning approach.

- The process stops when the elements of a node have the same value for the target variable (homogeneity).

D. RANDOM FOREST (RF) CLASSIFIER

Figure. 4 shows how RF is used in the context of intrusion detection. RF is based on creating multiple decision trees and determining the class of each DT [27]. The final class is defined using majority voting.

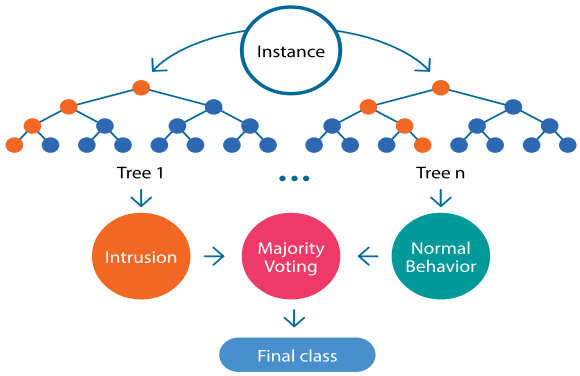

FIGURE 4. RF decision is a classification method based on constructing multiple DTs in the training phase, after which the final decision is based on a majority voting system among the trees.

RF uses bootstrap aggregating applied to a learning tree. It operates on a training set, for example, $X = x_1, x_2, \ldots, x_n$, having $Y = y_1, y_2, \ldots, y_n$ as responses. RF is executed by looping B times. In each iteration, it chooses a sample with changes n training examples X_b, Y_b from X, Y. Next, RF trains a classification tree f_b on X_b, Y_b. Finally, after finishing the loop, a majority vote is applied to determine the right class.

If C_b is the class prediction of the b^{th} RD tree, the final class will be:

$$\hat{C}_{rf}^{B} = majorityVoting\{\hat{C}_b\}_1^B \tag{1}$$

E. MULIT LAYER PERCEPTRON

The Multi-Layer Perceptron (MLP) is a neural network learning approach. It is a feedforward learning algorithm with several layers of nodes, including an input layer, an output layer, and some hidden layers. This supervised learning technique uses a nonlinear activation function in each neuron. By applying the back propagation training, MLP is able to solve several multidimensional classification problems. It can distinguish non-linearly separable data. With a large number of layers, it can be considered as a type of deep learning technique.

IV. PROPOSED MODEL AND EXPERIMENTAL STUDY

This section describes the evaluation criteria, followed by the results of using ML as an IDS.

A. APPLIED MODEL

The overall architecture of the used model is described in Figure. 6, including the detail of the model workflow. The KIA Soul dataset CAN bus has been extracted from a shared repository. Then, the process of labelling has been performed by executing prepossessing according to the dataset description given in [6]. Then, a set of ML tools has been applied using Python. Finally, the results are presented by attack types. Furthermore, an overall comparison has been made with other ML models executed in other works with the same dataset.

B. EVALUATION CRITERIA

In this paragraph, we define the list of criteria that have been used to evaluate the RF results:

Precision, which is defined by the following equation (2):

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

Recall, which is defined by the following equation (3):

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

The $f1{-}score$ combines the precision and the recall given by the equation (4):

$$f1{-}score = \frac{2 \times precision \times recall}{precision + recall} \tag{4}$$

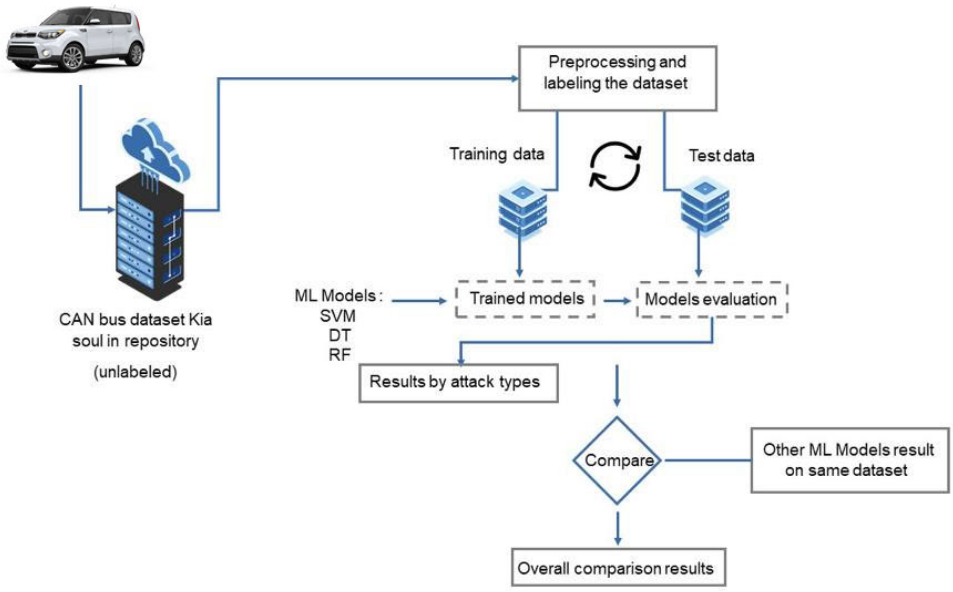

FIGURE 6. Overall architecture of the proposed model. The KIA soul dataset is extracted from a shared repository, and then a prepossessing labeling is performed. Then, advanced ML algorithms, including, SVM, DT, and RF, are applied for intrusion detection, and an overall comparison is made among the ML models.

Finally, accuracy is the most significant parameter representing the success of a classification method, as follows (5):

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{5}$$

where:

- *TP*: True positive: True intrusion that is detected correctly,
- *TN*: True negative: True intrusion that is not detected,
- *FP*: False positive: Normal behavior that is considered an attack,
- *FN*: False negative: Normal behavior that is not considered an attack.

C. DATASET

We have used a dataset which include DoS attack, fuzzy attack and impersonation attack. This dataset were constructed by logging CAN traffic via the OBD-II port from a real vehicle while message injection attacks were performing. The in-vehicle data was extracted from KIA SOUL.

- DoS Attack: Injecting messages of '0×000' CAN ID in a short cycle.
- Fuzzy Attack: Injecting messages of spoofed random CAN ID and DATA values.
- Impersonation Attack: Injecting messages of Impersonating node, arbitration ID = '0×164'.

This dataset with 47519 examples contains 2201 DoS attacks, 313 fuzzy attacks, 824 impersonation attacks. all the 16 features are presented in the table 1.

D. RESULTS AND DISCUSSION

Table 1 provides the feature list describing the prepared dataset [6], which includes three types of attacks: DoS, impersonation, and fuzzy attacks. A Python program was executed on a machine with 8GB RAM and an i7 processor. In the following, two kinds of comparison are given. The first comparison is based on attack type, and the second is an overall comparison with well-known methods.

TABLE 1. Feature list of the vectors in the KIA soul dataset, which contains essential information about the frames transmitted in the CAN bus.

Feature	Significance and description
$time$	Time stamp
$time_{remote}$	Last remote frame time stamp
id	Frame id
$id1$	Previous frame id
$id2$	Id of previous of previous frame
$id3$	Id of previous of previous of previous frame
rtr	If the frame is a remote frame or not (1 or 0)
dlc	Size of data filed in the frame (0:8)
$d0$	First byte of data
$d1$	Second byte of data
$d2$	Third byte of data
$d3$	Fourth byte of data
$d4$	Fifth byte of data
$d5$	Sixth byte of data
$d6$	Seventh byte of data
$d7$	Eighth byte of data

1) COMPARISON BASED ON ATTACK TYPE

As mentioned previously, we consider three type of attacks, which are DoS, impersonation, and fuzzy attacks. In Table 2, a comparison based on attack type is given.

Figures. 7, 9 and 8 show classifier results in terms of precision, recall, and f1-score for impersonation, DoS and fuzzy attacks, respectively. We found that the best result for the four classifiers is linked to detecting impersonation attacks.

TABLE 2. Comparison based on attack type: RF outperforms DT in terms of fuzzy and impersonation attack.

	Attack	Precision	Recall	F1-score	Support
	Normal	1.000	1.000	1.000	550
DT	Fuzzy	0.233	0.230	0.232	78
	DoS	0.768	0.757	0.762	206
	Impers.	0.990	0.990	0.990	11046
	Accuracy : 0.981				11880
RF	Fuzzy	0.266	0.051	0.086	78
	DoS	0.796	0.742	0.768	206
	Impers.	0.988	0.995	0.992	11046
	Accuracy: 0.985				11880
SVM	Fuzzy	0.000	0.000	0.000	78
	DoS	0.859	0.296	0.440	206
	Impers.	0.972	0.998	0.985	11046
	Accuracy: 0.972				11880
MLP	Fuzzy	0.000	0.000	0.000	78
	DoS	0.000	0.000	0.000	206
	Impers.	0.979	1.000	0.987	11046
	Accuracy: 0.961				11880

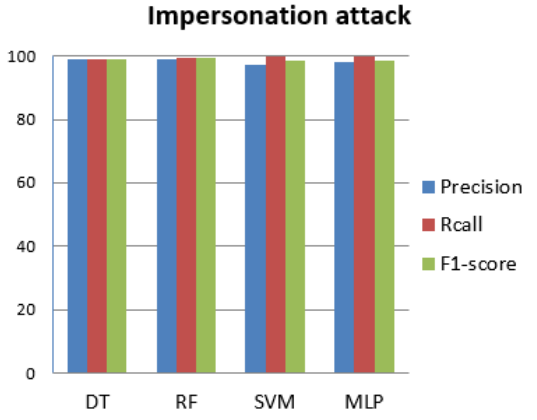

FIGURE 7. Impersonation attack detection results: DT, RF, MLP, and SVM show good performances. The good results can be explained by the high support for impersonation attacks in the dataset.

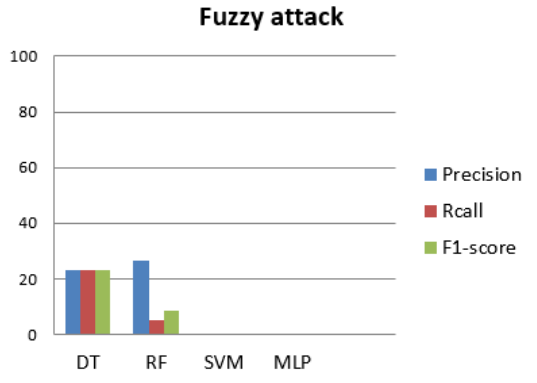

FIGURE 8. Fuzzy attack detection results: DT and RF show weak results, while SVM and MLP are not detecting this type of attack.

Meanwhile, the detection of fuzzy attacks is very low. SVM shows the worst performance with fuzzy attacks.

As we can see, the results are poor for fuzzy and DoS attacks. This can be explained by the insufficient number of examples of these attacks in the dataset.

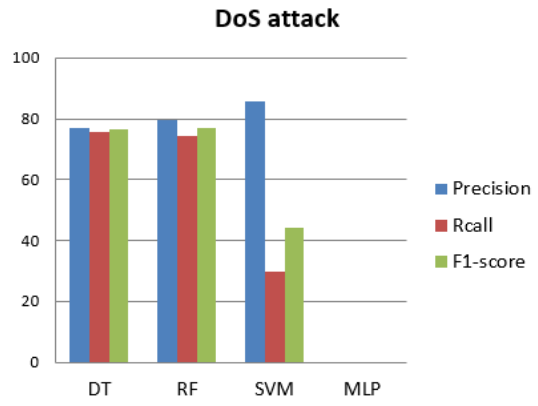

FIGURE 9. DoS attack detection results. SVM has the highest precision. In terms of combined precision, recall, and F1-score, DT and RF outperform SVM. MLP is not detecting DoS attacks.

We can see that RF outperforms DT, MLP and SVM with impersonation or fuzzy attacks. However, DT performs slightly better than RF and far from SVM and MLP. The worst performance is given by SVM as well as MLP with Fuzzy attacks. The best performance is given with impersonation attacks due to the support included in the dataset 11046. Meanwhile, the worst performance of the three classifiers is with fuzzy attacks, which is explained by the low support.

DT performs better than the other methods when DoS attacks occur. SVM has the worst performance with fuzzy attacks and worse performance compared to DT and RF. For fuzzy attack detection, SVM shows the worst results by no detection at all. In addition, the detection for this attack based on DT and RF is relatively weak. This fact can be explained by the low support of this attack in the dataset.

2) OVERALL COMPARISON

In this subsection, RF, DT, MLP, and SVM results will be compared to those of three other techniques: HTM, RNN, and HMM. The results of these three methods are directly taken from [21], where they were obtained from the same dataset.

Table 3 shows the accuracy results for the applied techniques, including RF, SVM, MLP and DT, which contains the values for accuracy, precision, recall, training time, and testing time for the four classifiers (SVM, RF, MLP, and DT) used to detect intrusion. Figure 10 shows a comparison of the precision between the best-known ML techniques (SVM, RF, DT, MLP, RNN, HTM, and HMM). It is clear that the precision of RF, SVM, MLP and DT is better than that of RNN and HMM, but it is slightly worse than HTM. Additionally, in this section we have prepared for each attack a specific database that contains only some examples of concerned attacks in normal cases. The results confirm the explanations of the previous results. It is clear that the attacks that have fewer examples in the base are the least recognized by the learning techniques. Indeed, the learning rates have been improved since the total number of examples in each base (per attack)

TABLE 3. Overall comparison between RF, DT, SVM, and MLP performance results using python and executed on an i7 PC with 8GB of RAM.

Classifier	RF	SVM	DT	MLP
Precision (%)	98.5269	97.2895	98.1902	95.2800
Recall (%)	98.1214	96.5583	98.1782	97.6100
Accuracy (%)	98.5269	97.2895	98.1902	97.6100
Training Time (s)	460.627	460.383	460.719	460.710
Testing Time (s)	14.933	14.919	14.935	14.925

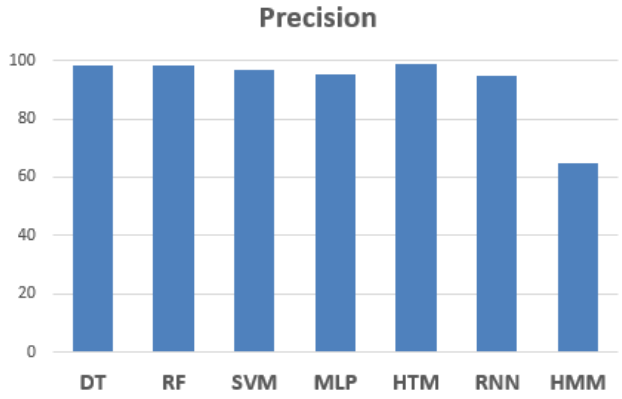

FIGURE 10. Precision comparison between the techniques used (RF, DT, MLP, and SVM) and methods previously used with the same dataset (HTM, RNN, and HMM).

has decreased. So, with only two classes the recognition improves.

Figure. 11 shows the recall factors of the seven methods. The RF, SVM, MLP, and DT classifiers outperform the other techniques (RNN, HMM, and HTM).

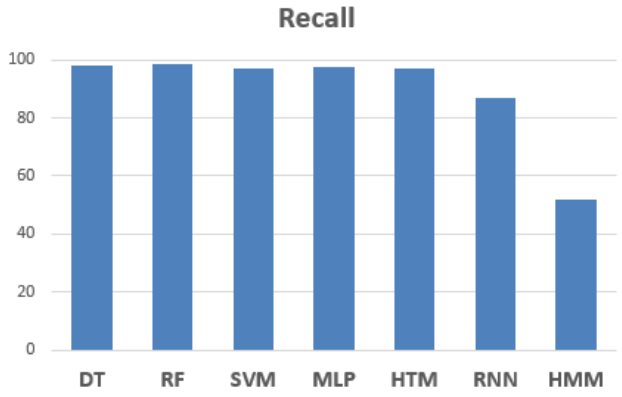

FIGURE 11. Recall comparison between the techniques used (RF, DT, MLP, and SVM) and the methods previously used on same dataset (HTM, RNN, and HMM).

The most important comparison is that of accuracy. Figure 12 shows that the four classifiers used in this study, RF, SVM, MLP, and DT, outperform other techniques. RF exceeds HTM by 1 : 3%, RNN by 12 : 2%, and almost doubles the performance of HMM. DT also outperforms other techniques by the same rate, while SVM exceeds HTM by 1.2%, RNN 12.1%, and also almost doubles HMM.

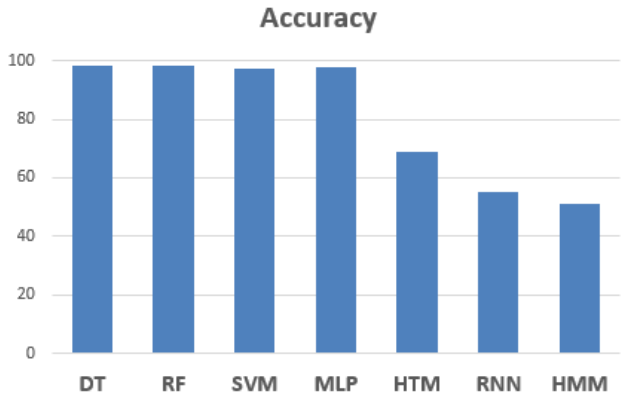

FIGURE 12. Accuracy comparison between the techniques used (RF, DT, MLP and SVM) and methods previously used on the same dataset (HTM, RNN, and HMM).

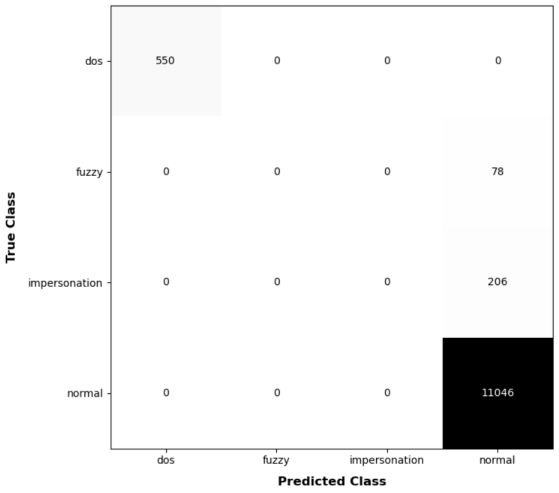

FIGURE 13. MLP confusion matrix.

In the next part, we present the confusion matrix of all techniques in Fig. 13, 14, 15 and 16.

The different results of each attack show that the number of attacks can influence the learning results. it can even be determining above a certain number. This is logical, as any learning model can only be generalized on the basis of a number of examples. it reminds us of the overfitting and underfitting problems.

Another type of comparison between the performance of different techniques can be made according to the percentage difference, as represented generally by equation 6:

$$PD = 100 \times \frac{|\Delta V|}{\frac{\sum V}{2}} \tag{6}$$

In our case general equation 7 can be used as follows:

$$PD(x, y) = 100 \times \frac{|x-y|}{\frac{x+y}{2}} \tag{7}$$

Table 4 describes the percentage distance between RF and HTM, RNN, HMM, SVM, MLP, and DT. When the other

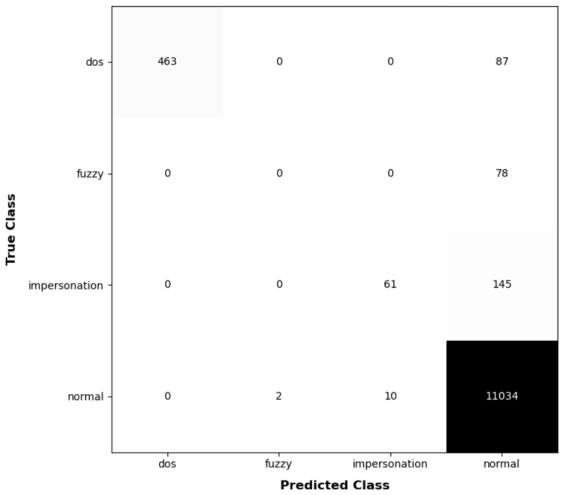

FIGURE 14. SVM confusion matrix.

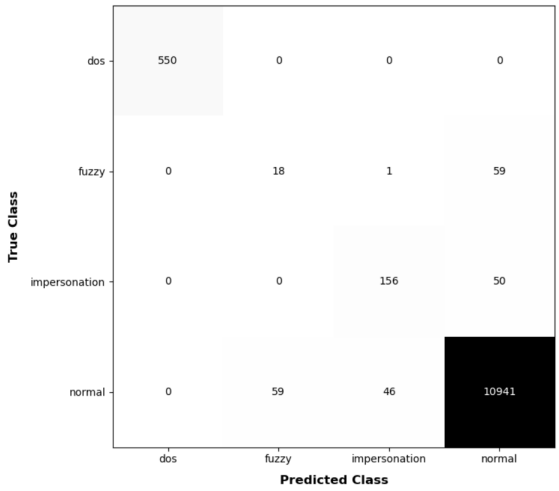

FIGURE 15. DT confusion matrix.

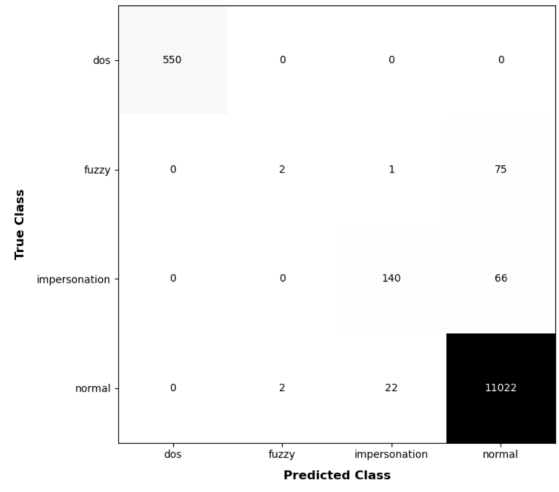

FIGURE 16. RF confusion matrix.

methods outperform RF, the values are underlined. Bold values represent outperformance of RF by more than 35% compared to the other classifiers, while italic values indicate

TABLE 4. Percentage distance-based comparison between RF and HTM, RNN, HMM, SVM, DT, and MLP.

Percentage distance (%)	Precision	Recall	Accuracy
RF to HTM	0.891	35.250	1.561
RF to RNN	3.217	70.204	*10.869*
RF to HMM	41.401	89.673	50.359
RF to SVM	1.934	1.121	1.114
RF to DT	0.058	0.070	0.070
RF to MLP	2.937	0.522	0.522

```
-----Random Forest------
......testing......
accuracy:  [0.15867003 0.79555976 0.91677189 0.69412879 0.39103441]
------------------------
-----Neural Network------
......testing......
accuracy:  [0.9760101  0.97611532 0.9760101  0.9760101  0.97611281]
------------------------
-----Decision Tree------
......testing......
accuracy:  [0.97590488 0.93297559 0.90088384 0.91245791 0.42628644]
------------------------
-----SVM------
......testing......
accuracy:  [0.95991162 0.9704335  0.96738215 0.97180135 0.9567505 ]
```

FIGURE 17. Cross validation results.

outperformance of RF by more than 10%. Ordinary values indicate outperformance of RF by less than 10%. In terms of accuracy, the RF classifier outperforms HTM slightly, and it also outperforms RNN by 10.68%. Notably, RF performs better than HMM by more than 50%. This table clearly shows that the RF classifier is more suitable in the context of intrusion detection for CAN-based in-vehicle networks.

SVM, DT, MLP, and RF achieve better results than RNN because statistical learning techniques are often more efficient in multidimensional problems. In our intrusion detection problem, the input data dimension is 16. The most difficult phase for the statistical learning technique is parameterization, and optimal parameters are crucial to the success of this approach. We thoroughly explored the research space before closing the training phase. This yielded results comparable to the neural network techniques.

We noticed a few disadvantages of SVM technique including the long training and testing time. It takes almost 100 times longer than the others techniques (MLP, DT and RF) to train and to test. Parameterization is also difficult for statistical learning techniques, especially for nonlinear learning. For example, it is difficult to find optimal parameters for the kernel function. We also applied cross-validation. you find in figure 16 all the accuracy rate of the different executions (cv = 5) for each learning approach.

V. CONCLUSION AND FUTURE WORK

This paper deals with an important problem: malicious intrusion in communications in vehicles using the CAN bus

protocol. Through an overview of the previous research in this area, we found that most existing studies have examined the behavior of exchanged frames or only superficially used the data contained in the frame without deeply considering the data itself. In addition, these studies do not use traditional classification techniques. For these reasons, in this study, we have proposed the use of the RF, SVM, MLP, and DT classifiers to distinguish between normal and malicious communications. According to the results of the experimental study performed with our dataset, we found that these four machine learning tools outperform the other techniques (HTM, RNN, HMM) in terms of accuracy.

In future work, we will apply non-supervised classification techniques to illustrate the detection performance with some unknown or new intrusions. It will also be important to apply deep learning techniques to large intrusion datasets.

ACKNOWLEDGMENT

The researchers would like to thank the Deanship of Scientific Research, Qassim University for funding the publication of this project.

REFERENCES

[1] R. Hajlaoui, H. Guyennet, and T. Moulahi, "A survey on heuristic-based routing methods in vehicular ad-hoc network: Technical challenges and future trends," *IEEE Sensors J.*, vol. 16, no. 17, pp. 6782–6792, Sep. 2016.

[2] A. Mchergui, T. Moulahi, B. Alaya, and S. Nasri, "A survey and comparative study of QoS aware broadcasting techniques in VANET," *Telecommun. Syst.*, vol. 66, no. 2, pp. 253–281, Oct. 2017.

[3] J. Liu, S. Zhang, W. Sun, and Y. Shi, "In-vehicle network attacks and countermeasures: Challenges and future directions," *IEEE Netw.*, vol. 31, no. 5, pp. 50–58, May 2017, doi: 10.1109/MNET.2017.1600257.

[4] L. Ran, W. Junfeng, W. Haiying, and L. Gechen, "Design method of CAN bus network communication structure for electric vehicle," in *Proc. Int. Forum Strategic Technol.*, Oct. 2010, pp. 326–329.

[5] W. Zeng, M. A. S. Khalid, and S. Chowdhury, "In-vehicle networks outlook: Achievements and challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 1552–1571, 3rd Quart., 2016, doi: 10.1109/COMST.2016.2521642.

[6] 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. 5709–5757.

[7] M. Bozdal, M. Samie, S. Aslam, and I. Jennions, "Evaluation of CAN bus security challenges," *Sensors*, vol. 20, no. 8, p. 2364, Apr. 2020, doi: 10.3390/s20082364.

[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, doi: 10.1109/TIFS.2018.2869351.

[9] M. Bozdal, M. Samie, and I. Jennions, "A survey on CAN bus protocol: Attacks, challenges, and potential solutions," in *Proc. Int. Conf. Comput., Electron. Commun. Eng. (iCCECE)*, Aug. 2018, pp. 201–205.

[10] S. Woo, H. J. Jo, and D. H. Lee, "A practical wireless attack on the connected car and security protocol for in-vehicle CAN," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 993–1006, Apr. 2015, doi: 10.1109/TITS.2014.2351612.

[11] L. Sellami, S. Zidi, and K. Abderrahim, "Self-adaptative multi-kernel algorithm for switched linear systems identification," *Int. J. Model., Identificat. Control*, vol. 31, no. 1, pp. 103–111, 2019, doi: 10.1504/IJMIC.2019.096792.

[12] S. Mahfoudhi, M. Frehat, and T. Moulahi, "Enhancing cloud of things performance by avoiding unnecessary data through artificial intelligence tools," in *Proc. 15th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jun. 2019, pp. 1463–1467, doi: 10.1109/IWCMC.2019.8766790.

[13] M. Panda, B. S. Gouda, and T. Panigrahi, "Fault diagnosis in wireless sensor networks using a neural network constructed by deep learning technique," in *Nature Inspired Computing for Wireless Sensor Networks*. Singapore: Springer, 2020, pp. 77–101.

[14] M. Emperuman and S. Chandrasekaran, "Hybrid continuous density hmm-based ensemble neural networks for sensor fault detection and classification in wireless sensor network," *Sensors*, vol. 20, no. 3, p. 745, Jan. 2020.

[15] M. Kordestani, M. F. Samadi, M. Saif, and K. Khorasani, "A new fault diagnosis of multifunctional spoiler system using integrated artificial neural network and discrete wavelet transform methods," *IEEE Sensors J.*, vol. 18, no. 12, pp. 4990–5001, Jun. 15, 2018.

[16] D. P. Kumar, A. Tarachand, and C. S. R. Annavarapu, "Machine learning algorithms for wireless sensor networks: A survey," *Inf. Fusion*, vol. 49, pp. 1–25, Sep. 2019.

[17] S. Tariq, S. Lee, H. K. Kim, and S. S. Woo, "Detecting in-vehicle CAN message attacks using heuristics and RNNs," in *Proc. Int. Workshop Inf. Oper. Technol. Secur. Syst., (IOSec)* in Lecture Notes in Computer Science, vol. 11398, A. P. Fournaris, K. Lampropoulos, and E. Marín-Tordera, Eds. Heraklion, Crete, Greece: Springer, Sep. 2018, pp. 39–45, doi: 10.1007/978-3-030-12085-6_4.

[18] C. Jichici, B. Groza, and P. Murvay, "Examining the use of neural networks for intrusion detection in controller area networks," in *Proc. Int. Conf. Secur. Inf. Technol. Commun.* in Lecture Notes in Computer Science, vol. 11359, J. Lanet and C. Toma, Eds. Bucharest, Romania: Springer, Nov. 2018, pp. 109–125, doi: 10.1007/978-3-030-12942-2_10.

[19] M.-J. Kang and J.-W. Kang, "Intrusion detection system using deep neural network for in-vehicle network security," *PLoS ONE*, vol. 11, no. 6, Jun. 2016, Art. no. e0155781.

[20] W. Wu, Y. Huang, R. Kurachi, G. Zeng, G. Xie, R. Li, and K. Li, "Sliding window optimized information entropy analysis method for intrusion detection on in-vehicle networks," *IEEE Access*, vol. 6, pp. 45233–45245, 2018, doi: 10.1109/ACCESS.2018.2865169.

[21] C. Wang, Z. Zhao, L. Gong, L. Zhu, Z. Liu, and X. Cheng, "A distributed anomaly detection system for in-vehicle network using HTM," *IEEE Access*, vol. 6, pp. 9091–9098, 2018, doi: 10.1109/ACCESS.2018.2799210.

[22] S. Woo, H. J. Jo, I. S. Kim, and D. H. Lee, "A practical security architecture for in-vehicle CAN-FD," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2248–2261, Aug. 2016, doi: 10.1109/TITS.2016.2519464.

[23] S. Zidi, T. Moulahi, and B. Alaya, "Fault detection in wireless sensor networks through SVM classifier," *IEEE Sensors J.*, vol. 18, no. 1, pp. 340–347, Jan. 2018.

[24] T.-K. Dao, T.-T. Nguyen, J.-S. Pan, Y. Qiao, and Q.-A. Lai, "Identification failure data for cluster heads aggregation in WSN based on improving classification of SVM," *IEEE Access*, vol. 8, pp. 61070–61084, 2020.

[25] F. Qu, Q. Jiang, G. Jin, Y. Wei, and Z. Zhang, "Mud pulse signal demodulation based on support vector machines and particle swarm optimization," *J. Petroleum Sci. Eng.*, vol. 193, Oct. 2020, Art. no. 107432.

[26] D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Trans. Syst., Man Cybern.*, vol. 21, no. 3, pp. 660–674, May 1991, doi: 10.1109/21.97458.

[27] G. Biau and E. Scornet, "A random forest guided tour," *Test*, vol. 25, no. 2, pp. 197–227, 2016.

TAREK MOULAHI received the joint Ph.D. degree from the University of Franche-Comté, Besançon, France, in March 2015, and the Sfax National School of Engineering, Tunisia. He is currently an Assistant Professor with Mathematics and Computer Science Departments, Faculty of Science and Technology of Sidi Bouzid (FSTSB), University of Kairouan, Tunisia, and the Department of Information Technology, College of Computer, Qassim University, Saudi Arabia. His research interests include wireless sensor networks, vehicular *ad hoc* networks (VANET) and the Internet of Things (IoT). He received the 2019 IEEE Sensors Council Sensors Journal Best Paper Runner-Up Award.

SALAH ZIDI received the Ph.D. degree from the University of Lille, France, with a focus on regulation and reconfiguration of multimodal transportation systems, in July 2007, and the HDR degree from the University of Lille1, France, in 2017. He is currently an Assistant Professor with the MIS Department, College of Business and Economics, Qassim University, Saudi Arabia, and an Associate Professor with the University of Gabes, Tunisia. His research interests include optimization, artificial intelligence, machine learning, feature extraction, and data analysis for automation systems and complex systems. He received the 2019 IEEE Sensors Council Sensors Journal Best Paper Runner-Up Award.

ABDULATIF ALABDULATIF (Member, IEEE) received the B.Sc. degree in computer science from Qassim University, Saudi Arabia, in 2008, and the M.Sc. and Ph.D. degrees in computer science from RMIT University, Australia, in 2013 and 2018, respectively. He is currently an Assistant Professor with the School of Computer Science and IT, Qassim University. His research interests include applied cryptography, cloud computing, data mining, and remote healthcare.

MOHAMMED ATIQUZZAMAN (Senior Member, IEEE) received the M.S. and Ph.D. degrees in electrical engineering and electronics from The University of Manchester, U.K., in 1984 and 1987, respectively. He currently holds the Edith J. Kinney Gaylord Presidential Professorship with the School of Computer Science, The University of Oklahoma, USA. His research has been funded by the National Science Foundation, the National Aeronautics and Space Administration, the U.S. Air Force, Cisco, and Honeywell. He has coauthored *Performance of TCP/IP Over ATM Networks* and has authored over 300 refereed publications. His current research interests include transport protocols, wireless and mobile networks, *ad hoc* networks, satellite networks, power-aware networking, and optical communications. He received the IEEE Communication Society's Fred W. Ellersick Prize and the NASA Group Achievement Award for outstanding work to further NASA Glenn Research Center's efforts in the area of the advanced communications/air traffic management's fiber optic signal distribution for Aeronautical Communications Project, the 2018 Satellite and Space Communications Technical Recognition Award for valuable contributions to the Satellite and Space Communications Scientific Community from the IEEE, and the 2017 Distinguished Technical Achievement Award from the IEEE Communications Society in recognition of outstanding technical contributions and services in the area of communications switching and routing. He Co-Chaired the IEEE High Performance Switching and Routing Symposium, in 2003 and 2011; the IEEE GLOBECOM and ICC, in 2006, 2007, 2009, 2010, 2012, and 2014; IEEE VTC, in 2013; and the SPIE Quality of Service Over Next Generation Data Networks conferences, from 2001 to 2003. He was the Panels Co-Chair of INFOCOM'05. He has been on the program committee of many conferences, such as INFOCOM, GLOBECOM, ICCCN, ICCIT, and Local Computer Networks. He serves on the review panels at the National Science Foundation. He was the Chair of the IEEE Communication Society Technical Committee on Communications Switching and Routing. He is the Editor-in-Chief of the *Journal of Networks and Computer Applications*, the Founding Editor-in-Chief of *Vehicular Communications*, and served on the Editorial Boards of many journals, including *IEEE Communications Magazine*, the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, the IEEE TRANSACTIONS ON MOBILE COMPUTING, *Real-Time Imaging* journal, the *International Journal of Sensor Networks*, and the *International Journal of Communication Systems*.