Indonesian Journal of Electrical Engineering and Computer Science
Vol. 30, No. 1, April 2023, pp. 276~288
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v30.i1.pp276-288

A machine learning approach for driver identification

Md. Abbas Ali Khan[1,2], Mohammad Hanif Ali[2], Fazlul Haque[1], Md. Tarek Habib[1]
[1]Department of Computer Science and Engineering, Faculty of Science and Information Technology (FSIT), Dafffodil International University (DIU), Dhaka, Bangladesh
[2]Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh

Article Info

Article history:

Received Jun 29, 2022
Revised Nov 10, 2022
Accepted Nov 18, 2022

Keywords:

CAN-Bus
Driver identification
Machine learning
OBD-II
Pattern analysis

ABSTRACT

Driver identification is a momentous field of modern decorated vehicles in the perspective of the controller area network (CAN-Bus). Many conventional systems are used to identify the driver. One step ahead, most of the researchers use sensor data of CAN-Bus but there are some difficulties because of the variation of a protocol of different models of vehicle. We aim to identify the driver through supervised learning algorithms based on driving behavior analysis. To identify the driver, a driver verification technique is proposed that evaluate driving pattern using the measurement of CAN sensor data. In this paper on-board diagnostic (OBD-II) is used to capture the data from CAN-Bus sensor and the sensors are listed under SAE J1979 statement. According to the service of OBD-II drive identification is possible. However, we have gained two types of accuracy on a full data set with 10 drivers and a partial data set with two drivers. The accuracy is good with less number of drivers compared to a higher number of drivers. We have achieved statistically significant results in terms of accuracy in contrast to the baseline algorithm.

Corresponding Author:

Md. Abbas Ali Khan
Department Computer Science and Engineering, Faculty of Science and Information Technology (FSIT)
Dafffodil International University (DIU)
Savar, Dhaka, Bangladesh
Email: abbas.cse@diu.edu.bd

1. INTRODUCTION

Every driver has their driving style, therefore the driver can be classified according to exploration through the driving pattern analysis. It is to be considered as a fingerprint of the driver's manner like acceleration, speed, and braking habits that vary from driver to driver. Driver fingerprinting could lead to important privacy compromises [1].

Today we cannot consider just a vehicle as a modern car, as it is a fully decorated smart device with various functions like multimedia, security system, and different sensors [2]. At most three sensors named fuel level, coolant temperature, and oil pressure were furnished last century until the 70th year. The sensors were very simple because the driver was informed regarding the features of the engine and the amount of fuel through the magnetoelectric and light display devices [2]. Nowadays, cars are equipped with many microcomputers. Information technology is developing rapidly and cars are connected to the internet. Using state-of-the-art technology in real-time all the microcomputers are communicated to each other through controller area network (CAN-Bus) [3]. As a result, the drivers feel secure and joyful during their trips and all other equipment is functioning properly.

To make a car more efficient a good number of technologies are used in the modern engine. To improve the engine performance direct injection technology was introduced in the modern car [4]. According to a survey, the researcher predicted that the number of sales of connected cars will reach 76.3 million in the

next 2023 [5]. Soon technology-based connected cars will make a digital platform where a multitude of sensors will take place like radar, light detection and ranging (LIDAR), cameras, ultrasonic sensors, and vehicle motion sensors [5]. Through state-of-the-art technology, modern engines use less fuel and besides get more power [6]. Most of the cars have partnered with other components which are highly technology-based, such as traffic lights, garage doors, and services [7]. Cars on the dashboard have green lights that indicate the drivers' efficient driving. It's improving driving style and fuel consumption. Not only on the driving style, there is a discount policy on insurance services but also real-time monitoring, maintenance, pathfinding, driving style development, and also consumption of fuel [8].

The more technology the car is based on, the more intelligent the thieves are. In the modern era, various modern techniques are used to steal a car key by the attacker. Vulnerabilities of connected cars will increase the auto-theft which is one of the threats [9]. Top-of-the-range vehicles are targeted by thieves who simply drive off after bypassing security devices by hacking on-board computers [10]. One technique involves breaking into the vehicle and plugging a laptop into the hidden diagnostic socket [10]. Penny [11] introduced the man-in-the-middle attack or relay attack, to do this radio signals are passed between two devices. Pekaric *et al*. [12] described other attacks such as GPS spoofing and message injection attacks. BMW Group [13] seamlessly integrates mobile devices, smart home technology, and vehicle's intelligent interfaces into a complete driver's environment. Even in 2021, they introduced a remote door unlock system through a signal to the driver's door to unlock [13]. The threats being discovered will be realized and the security of connected cars will become more important as more cars are connected to the internet.

Previous researchers introduced biometric authentication as one of the significant tools based on the physical characteristics of the driver like a fingerprint, face or voice detection, eye shell scanning, and also behavioral characteristics. Recognizing/analyzing the driver's driving pattern is a salient feature to develop the security of a car. Data-mining techniques are widely used by earlier researchers to detect such a novel attack. Because each driver has their driving style, data mining is also a prominent method to detect car theft (due to unexpected driving styles). As we say that the basis of telemetric data the features of the driver's driving pattern are reflected.

CAN-Bus is likely a nervous system used to allow configuration, data logging, and communication among electronic control units (ECU) e.g. ECU is like a part of the body and interconnected through CAN, by which information sensed by one part can be shared with another [14]. Up to 70 ECUs have a modern car e.g. the engine control unit, airbags, audio system, acceleration, and fuel unit. [15].

Normally, multi-sensor data is made up of in vehicle's CAN data. The in-vehicle CAN data such as steering wheel, vehicle speed, engine speed, and amount of fuel. Several researchers previously proposed a driver identification method based on in-vehicle CAN-Bus data. But direct connectivity is difficult to get data, so on-board diagnostics (ODB-II) is used. (OBD-II, ISO 15765) are a self-diagnostic and reporting capability that e.g. mechanics use to identify car issues, OBD-II specifies diagnostic trouble codes (DTCs) and real-time data (e.g. speed, revolution per minute (RPM)), which can be recorded via OBD-II loggers from CAN-Bus. Though such data is difficult to get, every moment data is passing, we need the parameter identifier (PID) number of each specific feature to correctly extract. It is non-public and it is made up based on the company. Many authors described the problem of CAN-Bus data for identifying the driver [16], [17].

In this paper, we aim to identify driver behavior through telemetric data using machine learning algorithms. We analyze the data in terms of training, testing, and validation to get model accuracy that helps us with driver identification.

2. LITERATURE REIVEW

Wakita *et al*. [18] uses telemetric data to investigate driver identification and identification accuracy decreases by 15% compared to the method. They use the role of non-public parameters in identifying the driver. Previous work had been done by using car driving simulated [18] data. Investigated the driver's behavior when he follows another car. The features mentioned below are used to observe such as accelerator pedal, car speed, brake pedal, and distance to the next car. Gaussian mixture model (GMM) is used to achieve 81% accuracy with 12 drivers and 73% of 30 drivers [18]. Zhang *et al.* [19] analyzes overtaking style for each driver, uses accelerator, and steering data and the accuracy is 85% for about 20 drivers through the hidden Markov model (HMM). Some other authors used smartphones to capture driver data. Sensors in the smartphone are- GPS, accelerometer, magnetometer, and gyroscope [20]-[22]. This data is used for driver profiling and other tasks. An Author used an inertial sensor and algorithm was SVM, k-means methods and got 60% accuracy between two drivers.

In another research, Carfora *et al*. [8] and Ullah and Kim [16] acquire data from in-vehicle CAN-Bus via OBD-II. Azadani and Boukerche [23] uses in-vehicle CAN-Bus sensor data and the accuracy was 94.27% for the human computer interaction (HCI)-lab dataset. He uses two datasets naming HCI-lab and HCRL and measured the performance. Kwak *et al*. [9] got 99% accuracy from 51 features of 10 drivers and used decision

tree (DT), k-nearest neighbors (kNN), random forest (RF), and multilayer perceptron (MPL). Choi *et al*. [24] find out the driving detection and driver recognition using both gaussian mixture model (GMM) and HMM methods, which are used for analyzing vehicle CAN-Bus data. Kedar-Dongakar dan Das [25] recognized the driver classification based on the energy optimization of a vehicle. Based on driving style three types of drivers are classified as aggressive, moderate, and conservative. The author considers the following features for his research work such as vehicle speed, acceleration, torque, acceleration pedal, steering wheel angle, and brake pedal pressure.

Several kinds of research have been going on neural network and deep learning algorithms for a few years back and draw a good impact on driver behavior identification works. Xun *et al*. [26] introduced convolutional neural network (CNN) and got 99% accuracy for 10 drivers. For advanced driver assistance systems (ADAS), this attribute can be an efficient factor to ensure the security and protection of the vehicle. Additionally, it extends the ADAS capabilities by creating different profiles for the drivers, which helps every driver according to his own driving style and improve the ADAS fidelity [27].

3. ARCHITECTURE OF THE INTENDENT SYSTEM

The architecture of the intended system is proposed to identify the authorized driver as shown in Figure 1. Modern vehicles are connected to the internet through IEEE 802 standard [28], [29] which transfers the driver data. The analysis module analyzes the data. If the driving pattern is not matched with the accredited driver then the driver identification cell detects and sends a message to the owner of the vehicle through the Wi-Fi module used to send information via the server [30].

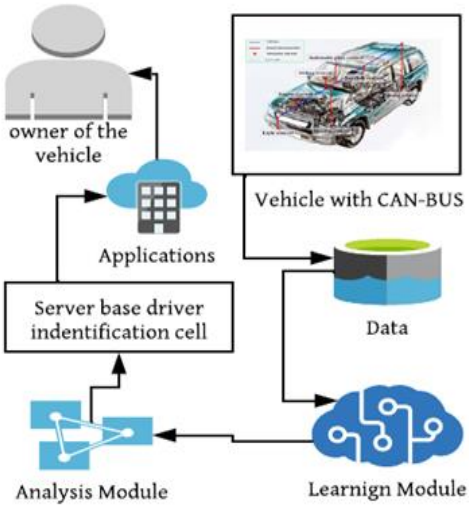

Figure 1. Architecture of the intended system for driver identification

4. METHOD

4.1. Dataset preparation

In this connection we need data of the trips for driver identification. Our model is considered an Ocslab driving dataset [31]. This date is used for driver classification and personalization based on pattern analysis. KIA motors corporation vehicles in South Korea were performed to collect the data and the experiment has been done since July 28, 2015. Total 10 drivers labeled "A" to "J" are included in the trips and cover 23 km length, completing two round trips from 8.00 PM to 11.00 PM. Three types (such as city road, freeway, and parking lot) of road are there with their own characteristics. There are a total 94,401 records with 51 dimensions (51 features) and Table 1 depicts the Ocslab dataset.

In real driving condition each driver drove their own style, in-vehicle CAN-Bus data were collected with OBD-II and CarbbigsP (OBD-II scanner). Not all data is possible to get because there are some limitations of OBD-II identifiers and sensors such as it cannot provide body control status or airbag status even wheel angle rotation status. OBD-II has a limited set of identifier [32] provided by the manufacturer, Table 2 shows some list of parameter IDs (PIDs) of service/mode (Hex) 01. There are 10 diagnostic services described in the

latest OBD-II standard SAE J1979 [32]. Few numbers of services are shown in Table 3. Driver's driving statistical features are exposed. Figure 3 shows the time series pattern of the in-vehicle's CAN data in the real-time driving situation of drivers A and D, where data fluctuation is visible in RPM.

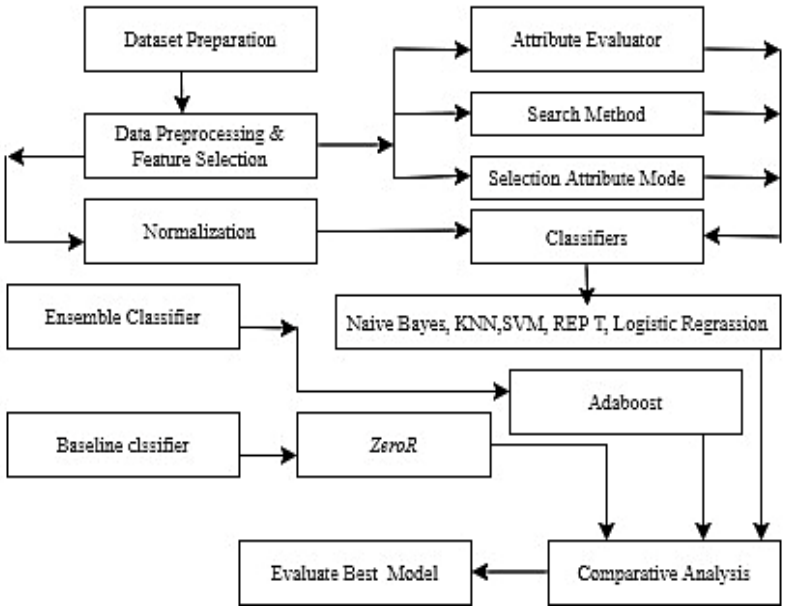

Figure 2. Steps of the proposed system

Table 1. Driving dataset of Ocslab with type and feature

Type	Features
Engine	Engine torque, Engine coolant temperature, Maximum indicated engine torque, Activation of Air Compressor, ……, Friction torque
Fuel	Long term fuel trim Bank1, Intake air pressure, Accelerator pedal value, ……., Fuel consumption
Transmission	Transmission oil temperature, Wheel velocity, front, left-hand, Wheel velocity, front, right-hand Wheel velocity, rear, left-hand, …., Torque converter Speed

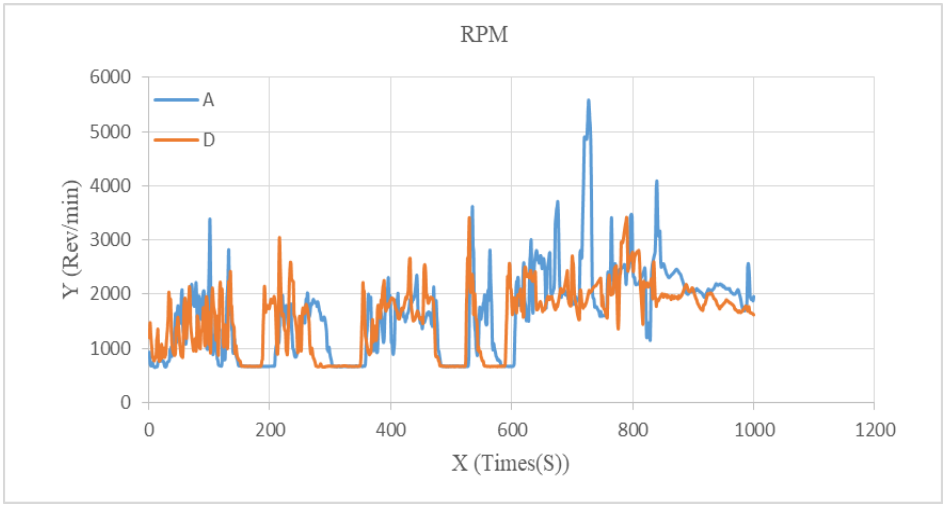

Figure 3. The revolutions per minute (RPM) of driver A and D

Table 2. List some OBD-II parameters

Service/Mode (Hex)	PID (hex)	Data byte returned	Description	Min Value	Max Value	Units
01	03	2	Fuel system status	-	-	-
	04	1	Calculated engine load	0	100	%
	0C	2	Engine speed	0	16,383.75	rpm
	OD	1	Vehicle speed	0	255	km/h
	--	---	----	--	--	--
	68	3	Intake air temperature sensors	-40	215	0C

Table 3. List of some OBD-II services/mode (hex)

Service/Mode (Hex)	Description
01	Show current data
02	Show freeze frame data
……..	……..
09	Request vehicle information
0A	Permanent Diagnostic Trouble Codes

4.2. Data prepossessing

There are 51 features used in our work in the dataset. Transform the collected data to our classification model for analysis we follow- feature selection, data normalization, and data processing through the sliding window technique. Example data show in (1). Where *d*columns correspond to *the d* variable and *N* rows correspond to *N* instances.

$$X = \begin{bmatrix} X_1^1 & X_2^1 \dots & X_d^1 \\ X_1^2 & X_2^2 \dots & X_d^2 \\ \vdots & \vdots & \vdots \\ X_1^N & X_2^N \dots & X_d^N \end{bmatrix} \quad (1)$$

4.2.1. Feature selection

We discard the following kind of features from the dataset for ameliorative achievement and accuracy of the model. We have considered CoorelationAttributeEval as an attribute evaluator, Ranker is used for the search method and also select cross-validation 10 and seed 1 while selecting the attribute mode.

- Homogeneous feature=$A_{h.}$
- Irrelevant feature=$B_{i.}$
- Superfluous feature and=$C_{s.}$
- Mostly Correlated feature=$D_{c.}$

Engine_torque and correction_of_engine_torque features are identical as well as engine_coolant_ temperature is a redundant feature. Hence, the selection of features referred to in [2] was performed, from the original dataset of 51 selecting 15 features. Table 4 shows the selected feature with statistical significance of mean and standard deviation in (2) and (3) respectively.

$$\bar{x} = \frac{\sum_{i=1}^{N} x_i}{N} \quad (2)$$

$$\sigma = \sqrt{\frac{\sum(x_i - \mu)^2}{N}} \quad (3)$$

4.2.2. Data normalization

As we see, different scales of data exist in the dataset. So we are to normalize the data according to the min-max approach. Normalization is essential for some machine learning algorithms like *k*-Nearest Neighbor (k-NN) and SVM. The normalization formulas for integrating data scales are shown in (4). In Figure 4 and 5 show the Ocslab data set before and earlier normalization respectively.

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (4)$$

Here, min means minimum value of a feature and max refers to the maximum value respectively

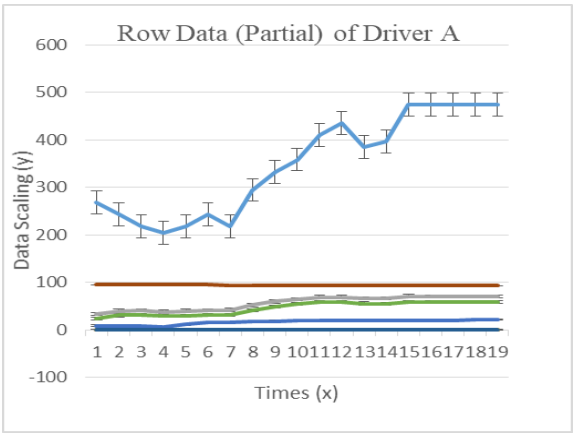

Figure 4. Shows the original dataset of Ocslab

Table 4. Selected 15 feature with mean and standard deviation

Feature	Vehicle Data Type	Mean	Standard deviation	Previous work	Classifiers
Long term fuel trim bank1	Fuel	2.843	1.363	[9]	DT, KNN, RF, MLP
Intake air pressure		36.85	27.95		
Accelerator pedal value		3.719	8.506	[18], [25], [33], [34], [35]	GMM, SMG, MM, GMM, MLP, SM, FNN,
Fuel consumption		757	761.13		
Maximum indicated engine torque	Engine	67.5	9.5		
Engine torque’		23.75	14.73	[23]	SVM, RF, NB, KNN
Calculated load value		41.30	18.38		
Friction torque		13.7	2.27		
Activation of air compressor		0.89	0.31		
Engine coolant temperature	Transmission	84.24	6.12		
Transmission oil temperature		80.21	10.5	[9]	DT, KNN, RF, MLP
Wheel velocity front left-hand		30.11	26.48		
Wheel velocity front right-hand		29.36	26.22		
Wheel velocity rear left-hand		29.20	26.10		
Torque converter speed’		1259.15	766.51	[9]	DT, KNN, RF, MLP

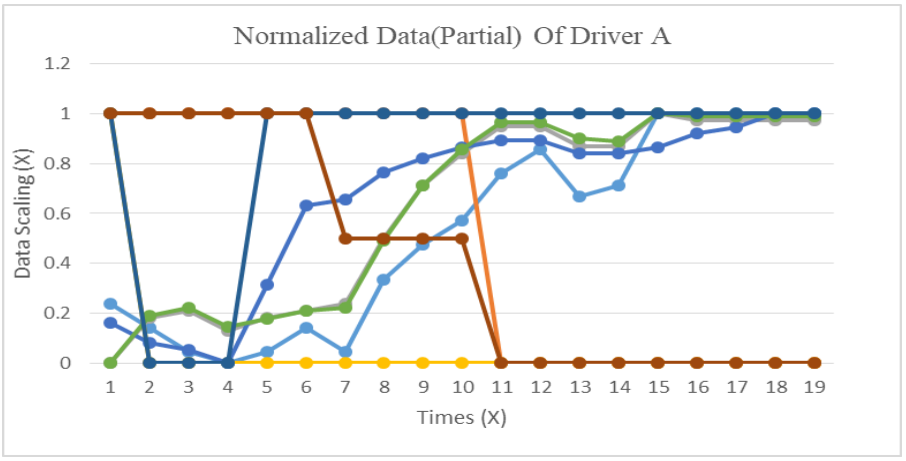

Figure 5. Shows the normalized dataset of the Ocslab

4.3. Description of the classifiers

We have considered supervised machine learning classifiers for performing the metrics named *k*NN, SVM, logistic regression, and reduced error pruning (REP) tree. The *k*-NN is an instance-based traditional machine learning algorithm. Both classification and regression cases *k*-NN can be used and select the number of neighbors through distance calculation of the query points. As shown in (6) is used to calculate the Euclidian distance between two points.

$$D=\sqrt{\sum_{i=1}^{n}(X_i - Y_i)^2} \quad (6)$$

Support vector machine (SVM) used for classification and regression problems [36]. The goal is to find a hyperplane in an N-dimensional space and separately classify the query data point. There is a decision boundary called hyperplane that is used to differentiate the classes. It also creates a margin separator with the nearest observations and it performs better if maximizes the margin. The equation represents the loss function that indicates maximize the margin.

$$C\,(x, y), where\ y = f(x) = \{0,\ if\ y * f(x) \geq 1, 1 - y * f(x), else\ 1 \quad (7)$$

Logistic regression predicts whether something is true or false. Instead of fitting a line to the data, it fits an "S" shaped "logistic function" and the curve goes from 0 to 1. The following equation is used to calculate the function, also called the sigmoid function.

$$S(x) = \frac{1}{1+e^{-x}} \quad (8)$$

Naïve Bayes is a classifier based on Bayes' theorem. It assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Naïve Bayes model is easy to build a large dataset and outperform with sophisticated classification method. The way Naïve Bayes is used to calculate the posterior probability, shows in (9).

$$P(c|x) = \frac{P(\boldsymbol{X}|\boldsymbol{C})P(c)}{P(x)} \quad (9)$$

Reduced error pruning (REP) Tree is a classification technique, from a given dataset it generates decision tree. It is seemed to be the extension of the C4.5 by improving the pruning phase. A distinct pruning dataset is used by the method and create multiple trees in different iteration. Finally select the best one. As measure, mean squared error is used for prediction the model by the tree [37]. To find the mean squared error used (10).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2 \quad (10)$$

4.4. Performance metrics

The dataset was represented as $X \epsilon R^{Nx*Mx}$ and we selected 15 features from the original dataset of 51 features. The new dataset are express as: $Xi=X - \Sigma (A_h+B_i+C_s+D_c)$. We have considered supervised machine learning classifiers to identify driving behavior. Previous researcher has done some work with the classifiers of e.g. Decision Tree (DT), *k*-NN, random forest (RF), MLP [9], and SVM, RF, *k*-NN [23].

In OBD-II the features which are publicly available and also in the Ocslab dataset, we used for preparing confusing metrics. Most of the researchers find accuracy to identify the driver behavior and a few number researchers use precision and f-score [2].

To measure the performance, we have used four indicators named accuracy, precision, F-Measure, and Recall. The computation and evaluation performance of the classifiers have occurred through the confusion metric. Once the model is generated then the classifier is tested by using a test dataset to check the model accuracy. Precision indicates how close or dispersed the measurement is to each other. It measures the number of correct positive predictors made. The recall is a metric that quantifies the number of correct positive predictions made out of all positive predictions that could have been made.

The number of *FP*'s, *FN*'s, *TP*'s, and *TN*'s cannot be calculated directly from this matrix. The values of *FP*'s, *FN*'s, *TP*'s, and *TN*'s for class $\boldsymbol{i}$ ($1 \leq i \leq n$) are determined as per [38].

$$TP_i = a_{ii}. \quad (11)$$

$$FP_i = \sum_{\substack{j=1, \\ j \neq i}}^{n} a_{ji}. \quad (12)$$

$$FN_i = \sum_{\substack{j=1, \\ j \neq i}}^{n} a_{ij}. \quad (13)$$

$$TN_i = \sum_{\substack{j=1, \\ j \neq i}}^{n} \sum_{\substack{k=1, \\ k \neq i}}^{n} a_{jk} \quad (14)$$

The final confusion matrix, which has dimension 2×2, comprises the average values of the n confusion matrices for all classes. For a binary, i.e. two-class problem, a confusion matrix gives the number of false positives (*FP*'s), false negatives (*FN*'s), true positives (*TP*'s), and true negatives (*TN*'s). From this confusion matrix, accuracy, precision, recall and *F*1-score are calculated in the following (15)-(18).

$$Precision = \frac{TP}{TP+FP} \times 100\% \quad (15)$$

$$Recall = \frac{TP}{TP+FN} \times 100\% \quad (16)$$

$$F_1\text{- score} = \frac{2 \times precision \times recall}{precision+recall} \times 100\% \quad (17)$$

$$Accuracy = \frac{TP+TN}{(TP+FN)+(FP+TN)} \times 100\% \quad (18)$$

Table 5. Confusion metrics of several classifiers of driver A and D

Classifier	Accuracy of the model	Accuracy by class (Binary)			
		Precision	F1-Score	Recall	Class (Driver)
Naive Bayes	96.15	92.1	95.3	98.8	A
		97.8	98.90	97.9	D
Logistic Regression	98.12	97.4	97.5	98.1	A
		98.8	98.8	99.0	D
kNN	99.99	1.00	1.00	1.00	A
		99.00	1.00	1.00	D
REP Tree	99.95	1.00	99.90	99.90	A
		1.00	1.00	1.00	D
SVM	99.88	98.99	98.87	98.99	A
		99.0	99.0	99.1	D
ZeroR (Baseline)	78.54	-	-	0.0	A
		78.0	88.8	1.0	D
AdaBoost (Ensemble)	99.91	1.00	99.9	99.8	A
		1.00	1.00	1.00	D

Table 6. Confusion metrics of several classifiers of all drivers

Classifier	Accuracy of the model of full dataset	Accuracy by class (multi class)			
		Precision	F1-Score	Recall	Class (Driver)
Naive Bayes	29.00%	41.8%	37.4%	33.8%	A
		33.4%	20.9%	15.2%	D
KNN	76.35%	95.1%	91.6%	88.3%	A
		76.1%	76.1%	76.1%	D
SVM	64.00%	95.5%	96.1%	97.3%	A
		50.6%	54.3%	59.6%	D
REP Tree	97.14%	99.5%	99.4%	99.3%	A
		96.6%	96.7%	96.6%	D
ZeroR (Baseline)	14.03%	-	-	0.00	A
		14.0%	24.6%	1.00%	D
AdaBoost (Ensemble)	20.90%	70.9%	79.1%	89.5%	A
		15.5%	26.9%	1.00%	D

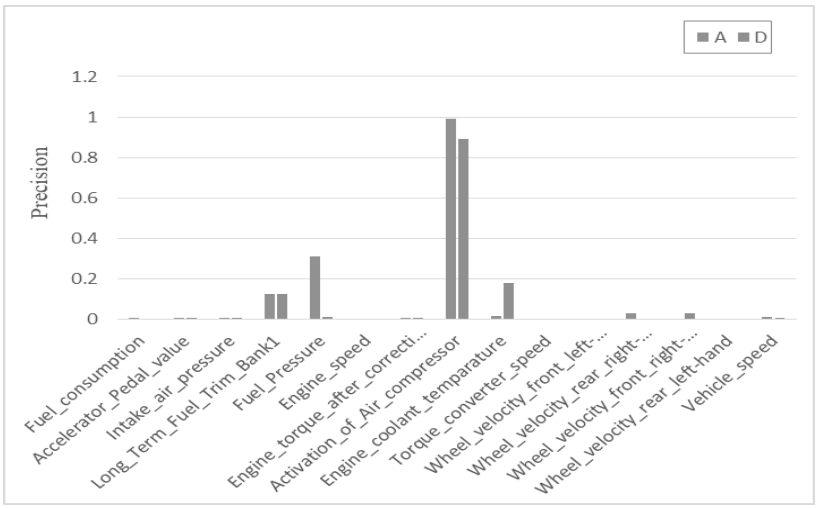

Figure 6. Precision using Naive Bayes model of driver A and D

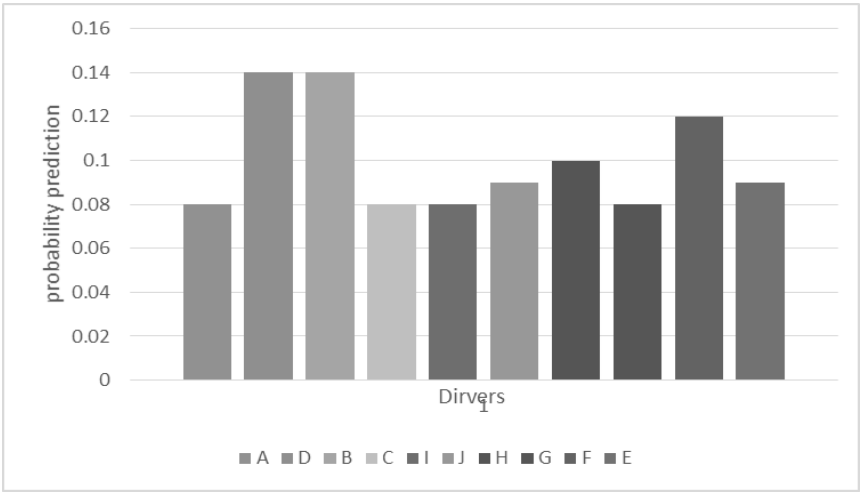

Figure 7. Randomly multi class probability prediction using Naive Bayes

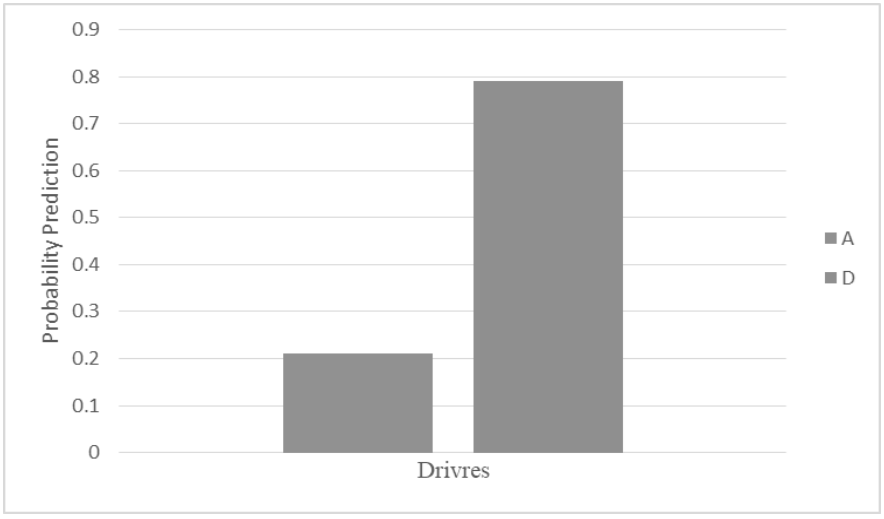

Figure 8. Binary class probability prediction using Naive Bayes

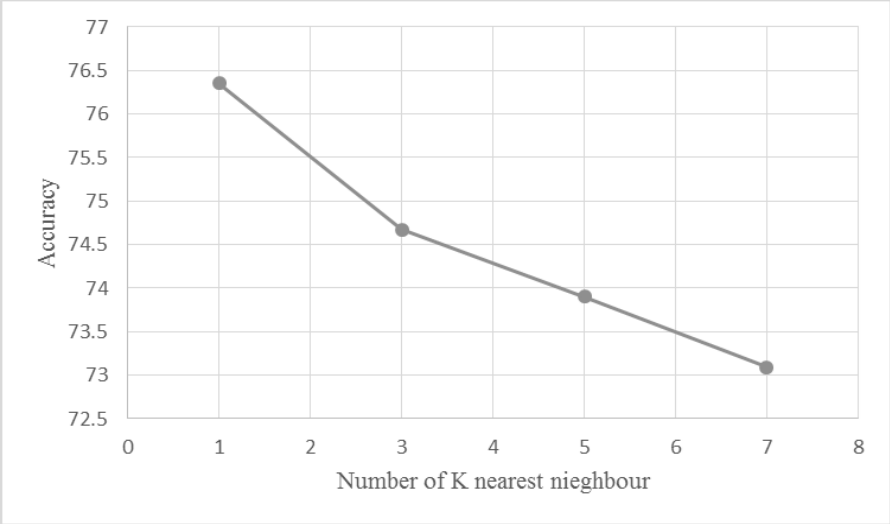

Figure 9. Shows the result of *k*NN algorithm with difference k value

5. RESULTS AND DISCUSSION

To evaluate the generalization of the model we have considered K-fold cross-validation for low bias and a modest variance [39]. We have used 10 folds where each fold contains 9 blocks are used for training and the remaining group taken as a test data set and obtained the mean performance.

For the classification of the driver, we have introduced the prominent supervised algorithm named Naive Bayes, Logistic Regression, *k*-NN, REP Tree, and SVM. Table 5 shows the result of all classifiers through the confusion metric. Among them, kNN shows 99.99% (highest) and Naive Bayes performs 96.15% (lowest) accuracy respectively. Mentionable that we have used 15 features and two drivers among 51 features and 10 drivers respectively from the Ocslab dataset. The results of ensemble classifiers where AdaBoost and voting are given 99.91% and 60.22% accuracy respectively.

Again we have calculated all the drivers' accuracy using the full dataset. Table 6 is representing the accuracy of only two drivers (A, D) and it also represents the baseline accuracy of 14.03%. In this research, we have figured out the ZeroR algorithm to calculate the baseline. Moreover, Adaboost uses an ensemble algorithm and the accuracy is important in this research because the other classifier's accuracy is better than this. The model is statistically significant because the accuracy of *k*-NN is better than baseline accuracy.

From the above discussion of Tables 5 and 6, we have recognized that state-of-the-art algorithms provide the best accuracy when the driver is less for the Ocslab dataset for public OBD-II in service 01. Figure 6 illustrates the precision of an algorithm named Naive Bayes, feature "Accivation_of_Air_Comprssior" points out the height result for drivers A and D. If we calculate all drivers randomly through ZeroR then the baseline accuracy is 14.03% whereas the Naive Bayes shows 29.93% accuracy on the full Ocslab dataset. This comparison indicates statistical significance. In Figures 7 and 8, there is another statistical importance that shows the multi-class and binary class probability prediction respectively. If we precisely classify the driver then we must have to consider less number of drivers like Figure 8 shows the good results for driver D than the driver D in Figure 7. The accuracy increases of drive D from 0.14% to 0.8%, which is more statistically significant.

Tuning the result through the hyperparameter of *k*-NN -1, 3, 5, 7 shown in Figure 9 with batch size 100 and Euclidean distance is used for finding the distance function. Each hyperparameter gives different results, the higher the number of the nearest neighbor, the lower the accuracy. In REP Tree uses the size of the tree is 1,397 with depth and learning rate are 26 and 0.001 accordingly, obtained 99.95% accuracy for drivers A and D.

6. COMPARATIVE PERFORMANCE ANALYSIS

There are six datasets and seven work domains are shown in Table 7. This table shows the comparative analysis of this work with other works already done before. The height accuracy of 99.99% belongs to this work based on the classifier, application domain, and the dataset. The statistically significant is exist because of different classes have different results for the same dataset.

Table 7. Comparative analysis among this work and the works

Method/Work done	Work domain	Dataset	No. of Class	Application Domain	Classifier	Accuracy
This work	Driver identification	Ocslab	2 (A & D)	Machine Learning	*k*NN	99.99%
Kwak *et al.* [9]	Driver profiling	Ocslab	2 (A & E)	Machine Learning	*k*NN	95.70%
Wakita *et al* [18], [25], [31], [34]	Driver identification	Driver signal data	276	-	GMM	76%
Azadani and Boukerche [23]	Automobile driver fingerprinting	In-Vehicle Data	1 (15 Drivers)	Machine Learning	*k*NN	100%
Zhang *et al.* [40]	Driver behavior identification	Ocslab	2 (B & C)	Deep Learning	LSTM-15	99.82%
Abdennour *et al.* [27]	Driver identification	Vehicular data trace-2	2 (4 Drivers)	Deep Learning	LSTM	99.00%
Choi *et al.* [24]	Classification of Driver Behavior	Vehicle signal	6	Statistical	Hidden Markov Model (HMM)	25.00%
Ullah and Kim *et al.* [16]	Lightweight driver behavior identification	Ocslab	-	Deep Learning	GRU	98.72%
Nishiwaki *et al.* [34]	Driver identificatio	Gas and Brake pedal reading	276	-	GMM	76.00%
Xun *et al.* [26]	Driver fingerprinting	-	10	Deep Leaning	CNN	100%

7. CONCLUSION

Driver identification is our prime aim by using telemetric data in terms of the best accuracy of the classifiers. The CAN-Bus data was collected through OBD-II. Only public PIDs are used for this research work because some non-public PIDs (which are hard to identify) are available in OBD-II. Previous researchers also use the same PIDs for several classifiers. Logistic regression is a prominent supervised learning. We could not build the model by using the full Ocslab dataset, even if it was a 24 hours continuous process. Partial dataset with two drivers we have built the model successfully and achieved good accuracy. We have achieved statistical significance because the baseline classifier is smaller than the others. Moreover, the accuracy of the ensemble and other supervised learning classifiers are almost the same. In *k*NN classifiers, there is a computational complexity e.g. it takes more processing time when we consider the higher nearest neighbor to build the model and provide less accuracy, in contrast, we have found height accuracy with the lower nearest neighbor and less computational complexity. To identify the driver we need around 100% accuracy. In this regard, *k*NN shows a height accuracy of 99.99% among two drivers with 15 features. Whereas for the full data set with 10 drivers the accuracy is 76.36% which is unsuitable for driver identification. We plan to use a compound feature for novel research with the Ocslab dataset in the future.

ACKNOWLEDGEMENTS

We thank KIA Motors Corporation for the Ocslab dataset.

REFERENCES

[1] M. Enev, A. Takakuwa, K. Koscher, and T. Kohno, "Automobile driver fingerprinting," *Proceedings on Privacy Enhancing Technologies*, vol. 2016, no. 1, pp. 34–50, Jan. 2016, doi: 10.1515/popets-2015-0029.

[2] K. Uvarov and A. Ponomarev, "Driver identification with OBD-II public data," in *2021 28th Conference of Open Innovations Association (FRUCT)*, pp. 495–501, Jan. 2021, doi: 10.23919/FRUCT50888.2021.9347648.

[3] S. Kaplan, M. A. Guvensan, A. G. Yavuz, and Y. Karalurt, "Driver behavior analysis for safe driving: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3017–3032, Dec. 2015, doi: 10.1109/TITS.2015.2462084.

[4] J. Hanzl, "Parking information guidance systems and smart technologies application used in urban areas and multi-storey car parks," *Transportation Research Procedia*, vol. 44, no. 1, pp.361-368, Jan. 2020, doi: 10.1016/j.trpro.2020.02.030.

[5] International Data Corporation (IDC), "Worldwide connected vehicle shipments forecast to reach 76 million units by 2023, according to IDC," *Businesswire*, 2019. https://www.businesswire.com/news/home/20190523005089/en/Worldwide-Connected-Vehicle-Shipments-Foreca St-to-Reach-76-Million-Units-by-2023-According-to-IDC (accessed Jan. 20, 2022).

[6] A. O. Hasan, *et al.* "An experimental study of engine characteristics and tailpipe emissions from modern DI diesel engine fuelled with methanol/diesel blends," *Fuel Processing Technology*, vol. 220, pp. 106901, Sep. 2021, doi: 10.1016/j.fuproc.2021.106901.

[7] R. Kemp, J. Schot, and R. Hoogma, "Regime shifts to sustainability through processes of niche formation: the approach of strategic niche management," *Technology analysis & strategic management*, vol. 10, no. 2, pp. 175-198, 1998, doi: 10.1080/09537329808524310.

[8] M. F. Carfora *et al*., "A 'pay-how-you-drive' car insurance approach through cluster analysis," *Soft Computing*, vol. 23, no. 9, pp. 2863–2875, May 2019, doi: 10.1007/s00500-018-3274-y.

[9] B. Il Kwak, J. Y. Woo, and H. K. Kim, "Know your master: Driver profiling-based anti-theft method," in *2016 14th Annual Conference on Privacy, Security and Trust, PST 2016*, Dec. 2016, pp. 211–218, doi: 10.1109/PST.2016.7906929.

[10] G. Chris, "Car-hackers driving off with top motors: Increasing numbers being stolen after thieves simply bypass security devices," *Daily Mail Crime Correspondent*, 2015. http://www.dailymail.co.uk/news/article-2938793/Carhackers-driving-motors-Increasing-numbers-stolen-thieves-simplybypass-security-devices.html (accessed Jan. 21, 2022).
[11] A. Mallik, "Man-in-the-middle-attack: Understanding in simple words," *Cyberspace: Jurnal Pendidikan Teknologi Informasi*, vol. 2, no. 2, pp. 109-134, Jan. 2019,
[12] I. Pekaric, C. Sauerwein, S. Haselwanter, and M. Felderer, "A taxonomy of attack mechanisms in the automotive domain," *Computer Standards & Interfaces*, vol. 78, p. 103539, Oct. 2021, doi: 10.1016/j.csi.2021.103539.
[13] BMW Group, "BMW ConnectedDrive," *www.bmwusa.com*, 2016. http://www.bmw.de/de/footer/publications-links/technology-guide/bmw-connecteddrive.html%5Cnwww.press.bmwgroup.com/austria/download.html?textId=45660&textAttachmentId=63611 (accessed Jan. 21, 2022).
[14] K. Iehira, H. Inoue and K. Ishida, "Spoofing attack using bus-off attacks against a specific ECU of the CAN Bus," in *2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC), 2018*, pp. 1-4, doi: 10.1109/CCNC.2018.8319180.
[15] A. Albert, "Comparison of event-triggered and time-triggered concepts with regard to distributed control systems," *Embedded World*, pp. 235–252, 2004.
[16] S. Ullah and D.-H. Kim, "Lightweight driver behavior identification model with sparse learning on in-vehicle CAN-Bus sensor data," *Sensors*, vol. 20, no. 18, p. 5030, Sep. 2020, doi: 10.3390/s20185030.
[17] A. Girma, X. Yan, and A. Homaifar, "Driver identification based on vehicle telematics data using LSTM-recurrent neural network," in *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, Nov. 2019, vol. 2019-Novem, pp. 894–902, doi: 10.1109/ICTAI.2019.00127.
[18] T. Wakita *et al*., "Driver identification using driving behavior signals," in *Proceedings. 2005 IEEE Intelligent Transportation Systems*, 2005., 2005, vol. 2005, pp. 907–912, doi: 10.1109/ITSC.2005.1520171.
[19] X. Zhang, X. Zhao, and J. Rong, "A study of individual characteristics of driving behavior based on hidden markov model," in *19th Intelligent Transport Systems World Congress, ITS 2012*, 2012, pp. 194–202.
[20] G. Castignani, T. Derrmann, R. Frank, and T. Engel, "Driver behavior profiling using smartphones: A low-cost platform for driver monitoring," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 91–102, 2015, doi: 10.1109/MITS.2014.2328673.
[21] A. Kashevnik, I. Lashkov, and A. Gurtov, "Methodology and mobile application for driver behavior analysis and accident prevention," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2427–2436, Jun. 2020, doi: 10.1109/TITS.2019.2918328.
[22] M. Van Ly, S. Martin, and M. M. Trivedi, "Driver classification and driving style recognition using inertial sensors," in *IEEE Intelligent Vehicles Symposium, Proceedings*, Jun. 2013, pp. 1040–1045, doi: 10.1109/IVS.2013.6629603.
[23] M. N. Azadani and A. Boukerche, "Driver identification using vehicular sensing data: a deep learning approach," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, Mar. 2021, vol. 2021-March, pp. 1–6, doi: 10.1109/WCNC49053.2021.9417463.
[24] S. Choi, J. Kim, D. Kwak, P. Angkititrakul, and J. H. L. Hansen, "Analysis and classification of driver behavior using in-vehicle CAN-bus information," *Biennial Workshop on DSP for In-Vehicle and Mobile Systems*, no. October 2015, pp. 17–19, 2007.
[25] G. Kedar-Dongarkar and M. Das, "Driver classification for optimization of energy usage in a vehicle," *Procedia Computer Science*, vol. 8, pp. 388–393, 2012, doi: 10.1016/j.procs.2012.01.077.
[26] Y. Xun, J. Liu, N. Kato, Y. Fang, and Y. Zhang, "Automobile driver fingerprinting: a new machine learning based authentication scheme," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1417–1426, Feb. 2020, doi: 10.1109/TII.2019.2946626.
[27] N. Abdennour, T. Ouni, and N. Ben Amor, "Driver identification using only the CAN-Bus vehicle data through an RCN deep learning approach," *Robotics and Autonomous Systems*, vol. 136, p. 103707, Feb. 2021, doi: 10.1016/j.robot.2020.103707.
[28] M. A. A. Khan, Ali, A. A. Khan, and M. Kabir, "Comparison among short range wireless networks: Bluetooth, Zig Bee & Wi-Fi," *Advances in Computer Science and Engineering*, vol. 4, no. 2, pp. 19–28, 2016.
[29] K. W. Al-ani, A. S. Abdalkafor, and A. M. Nassar, "An overview of wireless sensor network and its applications," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 17, no. 3, p. 1480, Mar. 2020, doi: 10.11591/ijeecs.v17.i3.pp1480-1486.
[30] M. S. Farag, M. M. M. E. Din, and H. A. E. Shenbary, "Deep learning versus traditional methods for parking lots occupancy classification," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 19, no. 2, pp. 964–973, 2020, doi: 10.11591/ijeecs.v19i2.pp964-973.
[31] HCRL, "Driving dataset," *ocslab.hksecurity.net*. https://ocslab.hksecurity.net/Datasets/driving-dataset (accessed Jan. 22, 2022).
[32] T. U. Kang, H. M. Song, S. Jeong and H. K. Kim, "Automated Reverse Engineering and Attack for CAN Using OBD-II," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pp. 1-7, Aug. 2018, doi: 10.1109/VTCFall.2018.8690781
[33] C. Miyajima *et al*., "Driver modeling based on driving behavior and its evaluation in driver identification," *Proceedings of the IEEE*, vol. 95, no. 2, pp. 427–437, Feb. 2007, doi: 10.1109/JPROC.2006.888405.
[34] Y. Nishiwaki, K. Ozawa, T. Wakita, C. Miyajima, K. Itou, and K. Takeda, "Driver identification based on spectral analysis of driving behavioral signals," in *Advances for In-Vehicle and Mobile Systems: Challenges for International Standards*, Boston, MA: Springer US, 2007, pp. 25–34.
[35] A. Wahab, C. Quek, C. K. Tan, and K. Takeda, "Driving profile modeling and recognition based on soft computing approach," *IEEE Transactions on Neural Networks*, vol. 20, no. 4, pp. 563–582, Apr. 2009, doi: 10.1109/TNN.2008.2007906.
[36] A. Hajraoui and M. Sabri, "Generic and robust method for head pose estimation," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 4, no. 2, pp. 439–446, Nov. 2016, doi: 10.11591/ijeecs.v4.i2.pp439-446.
[37] S. Kalmegh, "Analysis of WEKA data mining algorithm REPTree, simple cart and randomtree for classification of Indian News," *International Journal of Innovative Science, Engineering & Technology*, vol. 2, no. 2, pp. 438–446, 2015.
[38] M. T. Habib, A. Majumder, A. Z. M. Jakaria, M. Akter, M. S. Uddin, and F. Ahmed, "Machine vision based papaya disease recognition," *Journal of King Saud University-Computer and Information Sciences*, vol. 32, no. 3, pp. 300–309, Mar. 2020, doi: 10.1016/j.jksuci.2018.06.006.
[39] J. Brownlee, "A gentle introduction to k-fold Cross-Validation," *Machine Learning Mastery*, 2018. https://machinelearningmastery.com/k-fold-cross-validation (accessed Feb. 11, 2022).
[40] J. Zhang *et al*., "a deep learning framework for driving behavior identification on in-vehicle CAN-Bus sensor data," *Sensors*, vol. 19, no. 6, p. 1356, Mar. 2019, doi: 10.3390/s19061356.

BIOGRAPHIES OF AUTHORS

Md. Abbas Ali Khan is currently pursuing his Ph.D. at Jahangirnagar University, Bangladesh and received his M.Sc. in Computer Networks from School of Technology & Health, KTH Royal Institute of Technology, Stockholm, Sweden and B.Sc. in Computer Science from Daffodil International University, Dhaka, Bangladesh. Mr. Khan has been teaching as an Assistant Professor at Daffodil International University since 2017. He has published research papers both in National and International refereed journals. He is an Associate Member of Bangladesh Computer Society. He can be contacted at email: abbas.cse@diu.edu.bd.

Mohammad Hanif Ali he was born in 1956 and since 2003 Mr. Ali is a professor in the department of Computer Science and Engineering at Jahangirnagar University in Dhaka, Bangladesh. In 1999 he joined as an assistant professor to the university. Moreover, at Atomic Energy Commission he hold a position naming principal scientific officer in 1995. In 1987 Mr. Ali promoted to senior scientific officer and in 1982 he joined as a scientific officer at Atomic Energy Commission. He can be contacted at email: hanif_ju03@juniv.edu.

Professor Dr. Fazlul Haque received his Ph.D. in Computer Science and Engineering from Jahangirnagar University, Bangladesh in 2011. He received his M.Sc. Engineering in Electrical, System Design and Technology, Department of Telecommunication, University of Applied Sciences, Darmstadt, Germany in 2003. Currently he is holding the post of Associate Dean, Faculty of Engineering and Director, Institutional Quality Assurance Cell (IQAC) at Daffodil International University. Professor Haque has authored around 100 (Hundred) referred journal/conference papers in National and International arena. He has supervised more than 300 B.Sc. / M.Sc. thesis at Daffodil International University during the past years. He is also supervising Four Ph.D. students. His research interest's center on Telemedicine, Signal Processing, Computer Networking, and Data Communication. In the networking arena, he has almost 14 years' experience as Cisco Networking CCNA instructor. Professor Haque has been the member of Quality Assurance Unit, Association University of Asia and Pacific and also the member of International Society for Development and Sustainabilit. He can be contacted at email: akmfhaque@daffodilvarsity.edu.bd.

Md. Tarek Habib received his Ph.D. degree in the Department of Computer Science and Engineering at Jahangirnagar University. He obtained his M.S. degree in Computer Science and Engineering (Major in Intelligent Systems Engineering) and B.Sc. degree in Computer Science from North South University in 2009 and BRAC University in 2006, respectively. He is currently an Assistant Professor in the Department of Computer Science and Engineering at Daffodil International University. His research interest is in Artificial Intelligence, Computer Networks and E-Commerce. Ha has published a good number of articles in international journals and conference proceedings. He can be contacted at email: tarek.cse@diu.edu.bd.