Journal of Advanced Transportation

Data Analysis for Self-Driving Vehicles in Intelligent Transportation Systems

Lead Guest Editor: Hyunhee Park

Guest Editors: Kandaraj Piamrat, Kamal D. Singh, and Hsing-Chung Chen

Data Analysis for Self-Driving Vehicles in Intelligent Transportation Systems

Journal of Advanced Transportation

Data Analysis for Self-Driving Vehicles in Intelligent Transportation Systems

Lead Guest Editor: Hyunhee Park
Guest Editors: Kandaraj Piamrat, Kamal D. Singh, and Hsing-Chung Chen

Editorial Board

Contents

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 9386148, 2 pages
https://doi.org/10.1155/2020/9386148

Editorial

Data Analysis for Self-Driving Vehicles in Intelligent Transportation Systems

Hyunhee Park,[1] **Kandaraj Piamrat**,[2] **Kamal Singh**,[3] **and Hsing-Chung Chen**[4,5]

[1] *Korean Bible University, Seoul, Republic of Korea*
[2] *University of Nantes, Nantes, France*
[3] *Laboratory Hubert Curien, University Jean Monnet, St-Etienne, France*
[4] *Asia University, Taichung, Taiwan*
[5] *China Medical University Hospital, China Medical University, Taiwan*

Correspondence should be addressed to Hyunhee Park; parkhyunhee@gmail.com

Received 31 January 2020; Accepted 31 January 2020; Published 19 February 2020

Self-driving vehicles are regarded as the future of transportation. In the near future, self-driving vehicles would ferry passengers from one place to another place, like driverless taxis, and transport packages and raw materials from city to city. However, for all the optimism surrounding self-driving vehicles, there is also an equal amount of scepticism and concern. Many people believe that self-driving vehicles will be "no safer" than human-controlled vehicles. Therefore, the willingness of the public to ride in a fully self-driving vehicle will be very low due to nonzero accident rates.

A lot more data and testing are required to influence the public's beliefs on self-driving vehicles being ready for the road. Collecting more datasets will help to improve self-driving car modelling using data analysis; however, an incremental approach has to be taken for in-depth exploration of data analysis techniques applied to self-driving vehicles. This is due to the lack of sufficient information regarding how rare traffic and weather events should be modelled in transportation systems.

This special issue aims to provide a comprehensive overview of the most recent and promising advancements of data analysis technologies for self-driving vehicles in intelligent transportation systems. Data analysis technologies for self-driving vehicles are expected to cover the current state of the art and highlight remaining challenges and barriers to the development of self-driving vehicles as part of intelligent transportation systems. 11 papers were submitted for this issue; 4 of them were accepted for publication.

Below, we provide an overview of the selected articles for this special issue.

Vehicle Movement Analyses Considering Altitude Based on Modified Digital Elevation Model and Spherical Bilinear Interpolation Model: Evidence from GPS-Equipped Taxi Data in Sanya, Zhengzhou, and Liaoyang. The modified digital elevation (MDE) model and spherical bilinear interpolation (SBI) model were proposed for vehicle movement analyses considering altitude. In addition, the experimental data of 9,990 GPS-enabled taxis in Sanya, Zhengzhou, and Liaoyang were adopted to support comparisons. Measurement results showed that the MDE model having over 99% less disparity with direct solution as compared to the original model and SBI model could further improve the effects. In conclusion, the contributions of this study are as follows: (1) the MDE model was built to calculate vehicle movements by digital elevation data based on mathematical equations and (2) the SBI model was proposed and applied to improve the preciseness of GPS data with altitude of collaborative vehicles.

Malware Detection in Self-Driving Vehicles Using Machine Learning Algorithms. A machine learning-based data analysis method was proposed to accurately detect abnormal behaviours due to malware in large-scale network traffic in real time. First, the authors define a detection architecture, which is required by the intrusion detection module to detect and block malware attempting to affect the vehicle via a smartphone. Then, they propose an efficient algorithm for detecting malicious behaviours in a network environment and conduct experiments to verify algorithm accuracy and cost through comparisons with other algorithms. Here are the main contributions of this article: (1) the machine learning-based detection model was proposed for detecting adware and malware in a self-driving vehicle environment and (2) intrusion detection module architecture was defined to detect malware and prevent it from affecting the self-driving vehicles through a smartphone.

Data Analysis for Emotion Classification Based on Bio-Information in Self-Driving Vehicles. To classify the electroencephalography (EEG) data into four types of biometric emotion information (stability, relaxation, tension, and excitement), a fuzzy control system was designed and the pulse rate and blood pressure data were composed in single packets separately and sent to the database. The authors propose a system for inferring emotion using EEG, pulse, and blood pressure (systolic and diastolic blood pressure) of the user and recommending color and music according to the emotional state of the user for a user service in self-driving vehicle.

An Improved Automatic Traffic Incident Detection Technique Using a Vehicle to Infrastructure Communication. An ITS model was proposed to estimate the traffic incident from a hybrid observer (HO) method and detect the traffic incident by using an improved automatic incident detection (AID) technique based on the lane-changing speed mechanism in the highway traffic environment. Moreover, the authors have further evaluated the performance of the proposed method with well-known techniques such as Naive Bayes, KNN, and SVM using I-880 traffic data. This study has the following contributions: (1) the authors developed the connection between vehicles and roadside units (RSUs) by using a beacon mechanism and (2) they utilized the probabilistic approach to collect the traffic information data, by using a vehicle to infrastructure (V2I) communication.

The guest editors of this special issue believe that the selected articles contribute in moving the self-driving vehicles field forward and open new avenues for further research of smart transportation. Especially, these 4 selected articles provide key ICT solutions to various challenges faced by the developers of technologies used for self-driving vehicle safety and intelligent transportation systems.

Conflicts of Interest

The guest editors declare that there are no conflicts of interest regarding this special issue.

Acknowledgments

The guest editors would like to thank the reviewers for the time and efforts devoted to the analysis of the submitted manuscripts, as well as for the comments and suggestions used to improve the quality of the accepted articles.

Hyunhee Park
Kandaraj Piamrat
Kamal Singh
Hsing-Chung Chen

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 6301703, 21 pages
https://doi.org/10.1155/2020/6301703

Research Article

Vehicle Movement Analyses Considering Altitude Based on Modified Digital Elevation Model and Spherical Bilinear Interpolation Model: Evidence from GPS-Equipped Taxi Data in Sanya, Zhengzhou, and Liaoyang

Jiawei Gui [1,2] and Qunqi Wu [1,2]

[1] *School of Economics and Management, Chang'an University, Xi'an, Shanxi 710064, China*
[2] *Center of Comprehensive Transportation Economic Management, Chang'an University, Xi'an, Shanxi 710064, China*

Correspondence should be addressed to Jiawei Gui; gjw@chd.edu.cn

Received 25 May 2019; Revised 13 October 2019; Accepted 28 October 2019; Published 20 January 2020

Guest Editor: Kamal D. Singh

Aggravating energy shortages and increasing labor costs have become global problems and have garnered special importance in recent years in the transportation sector, especially in taxi markets. Automatic vehicles have a bright future, however, there is an equal amount of skepticism and concern about safety for all the optimism. To unlock the potential of automatic vehicles in intelligent transportation systems, a lot more data and testing are required to promote safety level as far as possible and achieve the organizational aim of reducing accidents to zero tolerance. And it is vital to provide accurate models for vehicle movement analyses. In this study, Modified Digital Elevation (MDE) model and Spherical Bilinear Interpolation (SBI) model were proposed for vehicle movement analyses considering altitude. And the experimental data of 9,990 GPS-enabled taxis in Sanya, Zhengzhou, and Liaoyang were adopted to support comparisons. Measurement results showed that MDE model had over 99% less disparity with direct solution than original model and SBI model could further improve the effects. It indicated that the application of MDE model and SBI model could improve both accuracy and efficiency of vehicle movement analyses and it had a bright future in the field of automatic vehicles. Future directions could be improving models and expanding data.

1. Introduction

Nowadays, there are plenty of urgent social issues in the transportation research field. For example, aggravating energy shortages and increasing labor costs have become global problems and have garnered special importance in recent years in the transportation sector, especially in taxi markets. With the worsening situation of energy shortages and labor costs, taxi markets are facing increasing opportunities as well as challenges. It is essential that appropriate counter measures must be taken to suppress or reverse or at least alleviate the worsening situation.

With the rapid progress of self-driving technologies, there are more and more advanced applications involved in intelligent traffic system. For example, automatic vehicles have a bright future. Dubai Roads and Transportation Authority tested autonomous pods, the world's first automatic taxi, in 2018. According to the evidence lab of union bank of Switzerland, the number of taxis in New York might drop by two-thirds if automatic vehicles were promoted completely, and the global automatic taxi market may exceed $2 trillion by 2030. Tesla Inc published a full self-driving computer called Autopilot 3.0 and an automatic taxi plan called Robo Taxi. Besides, many companies have already tested their automatic taxis or automatic buses along the road, such as Waymo in USA, Yandex in Russia, Gateway in UK, Easymile in France, Bestmile in Switzerland, Vislab in Italia, ZMP in Japan, KT in South Korea, GAC in China, etc.

However, there is an equal amount of skepticism and concern for all the optimism surrounding automatic vehicles. Most crucially, many people doubt the safety of automatic vehicles because they are not controlled by reliable human drivers but computers. On March 19th in 2018, a terrible accident took place in Tempe, Arizona that a 49-year-old woman with a bicycle on the sidewalk was hit to die by an Uber self-driving car at a speed of 65 kilometers per hour. Since then, the willingness of the public to ride in a fully self-driving vehicle fell in a very low level.

To unlock the potential of automatic vehicles in intelligent transportation systems, a lot more data and testing are required to promote safety level as far as possible and achieve the organizational aim of reducing accidents to zero tolerance. And it is vital to provide accurate models for vehicle movement analyses. In this study, modified digital elevation model and spherical bilinear interpolation model were proposed for vehicle movement analyses considering altitude. And GPS-enabled taxi data in Sanya, Zhengzhou, and Liaoyang were adopted to support comparisons.

The rest of this paper were organized as follows: Section 2 reviewed related works; Section 3 introduced formulas, models, data and tools; Section 4 presented a sample and described results in Sanya, Zhengzhou, and Liaoyang; Section 5 made a discussion; Section 6 summarized main conclusions, contributions and proposed future directions.

2. Literature Review

In this section, there are several similar problems and some related works till date, including Internet of Vehicles problem (see Section 2.1), data analysis for automatic vehicles (see Section 2.2) and taxi service improving problem (see Section 2.3).

2.1. Internet of Vehicles Problem. The Internet is a popular network technology all over the world and it has continuous developments in recent years. The evolution of network technology has led to a deployment of various access networks as introduced by Piamrat et al. [1]. Internet of Things (IoT) is a novel paradigm that is rapidly gaining ground in the scenario of modern wireless telecommunications as addressed by Atzori et al. [2]. Miorandi et al. proposed an overview of IoT technologies and services [3]. Xu et al. reviewed classical researches of IoT technologies and major applications in industries [4]. Zanella et al. proposed an urban IoT system and explored the application of the IoT paradigm to smart cities, taking Padova of Italy as an example [5].

Intelligent Transportation Systems (ITS) had significant impact on our life as introduced by Wang [6]. Internet of Vehicle (IoV) is one of the revolutions mobilized by IoT as addressed by Kaiwartya et al. [7]. Lu et al. regarded IoV as the next frontier for automotive revolution and the key to the evolution to ITS [8]. Yang et al. proposed an abstract network model of the IoV and presented its applications [9]. However, IoV also posed new challenges to the communication technology [10]. The information security of IoV was a considerable challenge. Singh et al. presented the potential of transforming vehicle communication in terms of efficiency and safety [11]. Chen proposed a trust-based cooperation authentication bit-map routing protocol against insider security threats in wireless ad-hoc networks [12]. Huang et al. proposed a proactive scheme to secure Edge computing-based IoV against RSU hotspot attack [13]. Furthermore, IoV has rapid developments in recent years. Butt et al. presented a scalable Social IoV (SIoV) architecture based on Restful web technology [14]. Chen et al. proposed Cognitive IoV (CIoV) to enhance transportation safety and network security by communication technologies [15].

Vehicular Ad-hoc Networks (VANET) comprise communications among vehicles and infrastructures by wireless local network technologies as addressed by Hartenstein and Laberteaux [16]. With the wide spread of Global Positioning System (GPS) and Geographic Information System (GIS), the participants in VANET could acquire much more information than before. Benefit from that, GPS-Equipped cars could not only acquire their real-time locations, but also road directions. Thus, VANET technologies helped improving road safety and providing comfort for passengers [17, 18]. In the field of ITS, IoV and VANET technologies created essential conditions for automatic vehicles.

2.2. Data Analysis for Automatic Vehicles. Automatic vehicles are regarded as the future of transportation. It has been a long-lasting dream of robotics researchers and enthusiasts as indicated by Petrovskaya and Thrun [19]. However, safety is the dominant factor in any automatic vehicle control system design as proposed by Shladover et al. [20]. Much more work was required until autonomous vehicles could participate in real-world urban traffic safely and robustly as proposed by Luettel et al. [21]. To ensure the safety of autonomous vehicles, a holistic fleet deployment scheme from interdisciplinary perspectives should be established as proposed by Koopman and Wagner [22]. Wang et al. investigated the acceptance of intelligent driving vehicles in Guangzhou and indicated that consumers focused on not only the developmental prospects but also the technological safety of intelligent driving vehicles [23]. With the rapid progress of IoV and VANET, there is a great deal of technologies involved in data analysis for automatic vehicles, which are described as follows.

(i) Vehicle communication systems. Machine-to-Machine (M2M) communication for the IoT system was considered to be a key technology in future networks as introduced by Chen et al. [24]. Park et al. proposed a transmission strategy for use with a capillary M2M system under wireless personal area networks based on a series of performance analysis for contention adaptation of M2M devices with directional antennas [25, 26]. Zhou et al. proposed a two-stage access control and resource allocation algorithm based on M2M communication and verified it under various simulation scenarios. With the rapid progress of IoV and VANET, M2M communication has been applied to vehicle communication in recent years. The vehicle communication systems include but are not limited to Vehicle-to-Pedestrian (V2P), Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I). Rahimian et al. examined the feasibility that sending traffic warnings to texting pedestrians based on V2P communication when they initiate an unsafe road crossing influences their decisions and actions [27]. Fan et al. proposed a traffic-aware relay selection based on millimeter-wave

V2V communication to overcome the Line-of-sight blockage problem [28]. Lyamin et al. proposed a data-mining-based method for V2V communications by random and On-off models [26]. Han et al. proposed an optimal signal control algorithm for signalized intersections using individual vehicle's trajectory data based on V2I communication [30]. Jia et al. verified that autonomous vehicles have significantly improvements on traffic efficiency via V2V and V2I communications [31].

(ii) Machine learning methods. Machine learning were used to identify objects in images, transcribe speech into text, match news items, posts or products with users' interests, and select relevant results of search as introduced by Lecun et al. [32]. With the rapid progress of IoV and VANET, machine learning methods has been applied to vehicle simulation, emulation and prediction in recent years. Aramrattana et al. proposed a simulation framework for cooperative safety assessment and efficiency evaluation in ITS [33]. Elwekeil et al. proposed a deep learning approach to achieve both reliability and spectrum efficiency of V2V communications [34]. Lv et al. proposed a modified cyber-physical system for automated electric vehicles based on unsupervised machine learning algorithms [35]. Li et al. proposed a proactively load balancing approach for vehicular network traffic control based on V2I communication, convolutional neural networks and deep learning to enable efficient cooperation among mobile edge servers [36].

(iii) Big data technologies. In recent years, a significant change in ITS was that much more data were collected from a variety of sources and can be processed into various forms as proposed by Zhang et al. [37]. And numerous advanced multidisciplinary journals began publishing a special issue of big data, for instance, Nature in 2008 [38] and Science in 2011 [39]. To solve big data, several technologies were proposed, including cloud computing, artificial intelligence and blockchain technology. Gubbi et al. proposed a cloud centric vision for worldwide implementation of IoT [40]. Whaiduzzaman et al. verified that cloud computing was a technologically feasible and economically viable paradigm for vehicles [41]. Al-Fuqaha et al. explored the relationships between IoT and big data analytics [42]. Ning et al. proposed a three-layer model based on fog computing to enable distributed traffic management and verified it by trajectory-based performance analysis [43]. As for applications, more and more researches focus on large scale taxi GPS data [44–46].

2.3. Taxi Service Improving Problem. Taxi is the main constituent of urban transportation. With the rapid progress of wireless sensor network [47], vehicle markets meet an evolution from intelligent vehicle grid to autonomous, Internet-connected vehicles and vehicular cloud as discussed by Gerla et al. [48]. Automatic taxis have gradually become reality in several advanced regions but not fully promoted yet. Just like every coin has two sides, automatic taxis have advantages and disadvantages. On the one hand, automatic taxis overcome some defects of human drivers, including excessive stress [49], inadequate sleepiness [50], and negative behaviors [51]. On the other hand, automatic taxis need mountains of work to achieve functions and ensure security. Taxi automation is one of the most promising technologies for the future and it is a challenge well worth meeting for taxi companies.

Fast and reliable service that can compete with the single occupancy vehicle was one of the demands of transit users as proposed by El-Geneidy et al. [52]. And it has become significant to provide users with a range of security-related and user-oriented vehicular applications as proposed by Ning et al. [53]. Thus, it is necessary for taxi companies improve their service to meet the demand of passengers. Yuan et al. proposed a price equilibrium model for taxi market to improve the service level [54]. Song et al. proposed a planning concept from the perspective of supply and demand economic equilibrium to optimize the transportation markets [55]. Dou et al. proposed a heuristic line piloting method that a taxi deviating from the typical route would raise an alert when malfunction took place or even hijacked by criminals [56]. Tang et al. proposed a customer-search model based on route choice behavior analysis to help taxi drivers find next passengers in urban road networks [57]. Although taxi companies and governments have already made great efforts, there are still plenty of service improving problems to be settled.

To unlock the potential of IoV in ITS, in the previous study [58], Gui and Wu measured taxi efficiency based on 2191 GPS-equipped taxi data in Sanya and indicated that the application of Motorcade-Sharing model could not only alleviate urban traffic congestion but also optimize urban taxi markets. However, altitude was not considered in that study which might reduce the measurement accuracy and result in lower safety level when altitude is nonnegligible. Thus, Original Model need improvements.

To sum up, some helpful works have already been done. However, few researches took altitude into consideration and proposed accurate models for vehicle movement analyses. Besides, autonomous vehicles need assigning continuous directives in time while it is difficult for complex algorithms to response immediately under the background of big data.

How much the altitude will affect the vehicle movement analyses? How to simplify complex algorithms of the Direct Solution for automatic vehicles? And how to handle the situation when some data was missing? In the next section, methods and data are involved to explore these questions.

3. Methods and Materials

In this section, there are several parts of methods and materials. First, an analysis procedure was presented as an overview (see Section 3.1). Second, traditional methods were reviewed, including Direct Solution (see Section 3.2) and Original Model (see Section 3.3). Third, modified models were proposed, including MDE model (see Section 3.4) and SBI model (see Section 3.5).

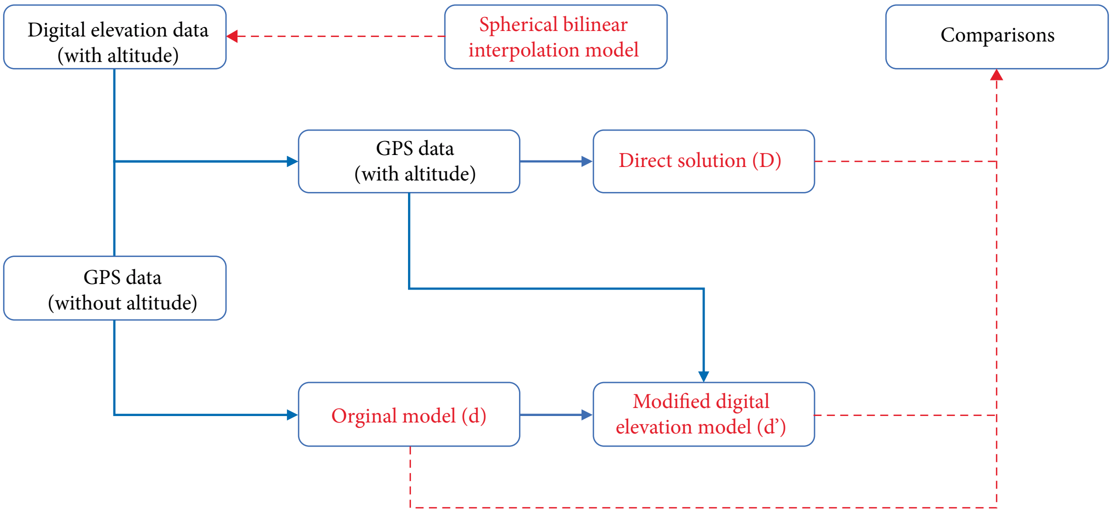

Digital elevation data (with altitude)
Spherical bilinear interpolation model
Comparisons
GPS data (with altitude)
Direct solution (D)
GPS data (without altitude)
Orginal model (d)
Modified digital elevation model (d')

Figure 1: Analysis procedure design.

Table 1: Function comparison of direct solution, original model, and MDE model.

Data source	Data	Direct solution (D)	Original model (d)	MDE model (d')
GPS data (without altitude)	Longitude	√	√	√
	Latitude	√	√	√
Digital elevation data	Altitude	√		√

3.1. Analysis Procedure. To solve this problem, a series of analysis procedure was established (see Figure 1).

In terms of data, GPS data without altitude and digital elevation data were collected at the beginning. GPS data with altitude was calculated on the basis of them afterwards (see Section 3.6). And the preciseness of GPS data with altitude was further improved by Spherical Bilinear Interpolation (SBI) Model (see Section 3.5).

In terms of model, three models were adopted for analyzing vehicle movements. They are direct solution (see Section 3.2), original model (see Section 3.3) and Modified Digital Elevation (MDE) model (see Section 3.4) and they have different functions (see Table 1).

(i) Direct solution. It is the traditional approach with large computation requirements because of its square algorithm, which is one of its drawbacks.

(ii) Original model. It simplifies the square algorithm of Direct Solution by cosine theorems (see Distance Formula in the previous study [58]). However, it ignores altitude which affects the measurement accuracy of results when altitude is nonnegligible.

(iii) MDE model. It adopts original model and takes altitude into consideration. In theory, MDE model has more accurate results than original model especially when altitude is nonnegligible. In addition, MDE model adopts SBI model to improve the preciseness of GPS data with altitude.

In this study, vehicle movements were analyzed by 3 models above mentioned at the same time and then a series of comparisons were raised on the basis of results. The notations for variables were in the Appendix (see Appendix A).

3.2. Direct Solution. Direct solution is the traditional approach and it calculates distances in a direct way (see Figure 2).

There are 4 steps in this section.

First, in order to describe the location of vehicles in a mathematical way, the Spherical Coordinate System was established in Formula (1).

$$P(\rho, \theta, \varphi), \tag{1}$$

where $\rho \geq 0$, $0 \leq \theta \leq \pi$, and $0 \leq \varphi < 2\pi$. The variable ρ represents the radial distance. The variable θ represents the polar angle. The variable φ represents the azimuthal angle. The radial distance can be divided into two parts by Formula (2).

$$\rho = R + h, \tag{2}$$

where $R = 6378137.00$ (m), which represents the radius of the earth. The variable h represents the altitude. Based on Formulas (1), (2), the coordinate of vector $\overrightarrow{OP}$ in Spherical Coordinate System can be represented by Formula (3).

$$\overrightarrow{OP} = (R + h, \theta, \varphi). \tag{3}$$

Second, in order to calculate the distance between two locations by Distance Formula, the Rectangular Coordinate System was established in Formula (4).

$$\overrightarrow{OP} = (x, y, z). \tag{4}$$

Based on Distance Formula, the distance between points P_1 and P_2 in Rectangular Coordinate System can be calculated by Formula (5).

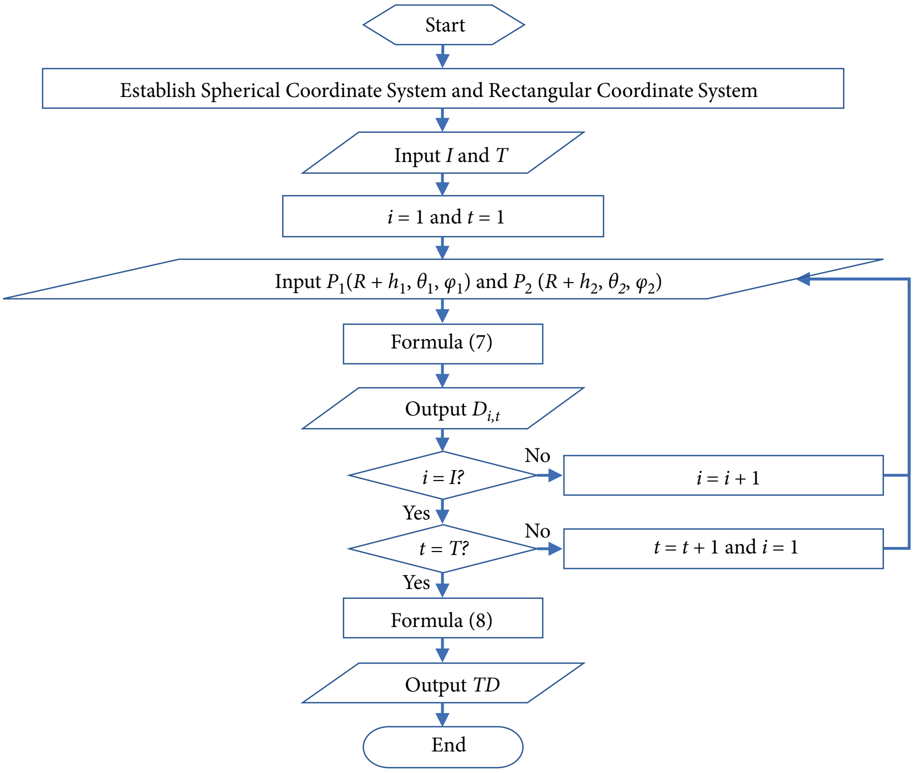


FIGURE 2: Flow chart of direct solution.

$$|P_1P_2| = \sqrt{(\Delta x)^2 + (\Delta y)^2 + (\Delta z)^2}. \tag{5}$$

Third, spherical coordinates were converted into rectangular coordinates. Based on Formulas (1), (4), the relationship of them can be represented by Formula (6).

$$\begin{aligned} x &= \rho \sin\theta \cos\varphi, \\ y &= \rho \sin\theta \sin\varphi, \\ z &= \rho \cos\theta. \end{aligned} \tag{6}$$

Based on Formula (6), the spherical coordinate of point $P_1(R+h_1, \theta_1, \varphi_1)$ was converted into $P_1((R+h_1)\sin\theta_1\cos\varphi_1$, $(R+h_1)\sin\theta_1\sin\varphi_1$, $(R+h_1)\cos\theta_1)$ in Rectangular Coordinate System, and the spherical coordinate of point $P_2(R+h_2, \theta_2, \varphi_2)$ was converted into $P_2((R+h_2)\sin\theta_2\cos\varphi_2$, $(R+h_2)\sin\theta_2\sin\varphi_2$, $(R+h_2)\cos\theta_2)$ in Rectangular Coordinate System.

Based on Formula (5), the distance between points P_1 and P_2 under direct solution can be calculated by Formula (7), which was sophisticated to some extent.

$$D = \sqrt{[(R+h_1)\sin\theta_1\cos\varphi_1 - (R+h_2)\sin\theta_2\cos\varphi_2]^2 + [(R+h_1)\sin\theta_1\sin\varphi_1 - (R+h_2)\sin\theta_2\sin\varphi_2]^2 + [(R+h_1)\cos\theta_1 - (R+h_2)\cos\theta_2]^2}. \tag{7}$$

Fourth, all the distances were summed up. Supposing that the variable TD represents total movements of vehicles under Direct Solution and it can be calculated by Formula (8).

$$TD = \sum_{i=1}^{I}\sum_{t=1}^{T} D_{i,t}. \tag{8}$$

3.3. Original Model. Original Model simplifies the square algorithm of Direct Solution by cosine theorems and it calculates distances in a simple way (see Figure 3).

There are 5 steps in this section.

First, the Spherical Coordinate System was established in Formula (1).

Second, the Rectangular Coordinate System was established in Formula (4).

Third, in order to calculate the distance between two locations, the cosine value of the angle between vectors $\overrightarrow{OP_1}$ and $\overrightarrow{OP_2}$ was calculated by Formula (9).

$$\cos\left\langle \overrightarrow{OP_1}, \overrightarrow{OP_2} \right\rangle = \frac{\overrightarrow{OP_1} \cdot \overrightarrow{OP_2}}{\left|\overrightarrow{OP_1}\right|\left|\overrightarrow{OP_2}\right|}. \tag{9}$$

Formula (9) can be converted into Formula (10) and the derivation process is in the Appendix part (see Appendix B in the previous study [58]).

$$\cos\left\langle \overrightarrow{OP_1}, \overrightarrow{OP_2} \right\rangle = \sin\theta_1\sin\theta_2\cos(\varphi_2 - \varphi_1) + \cos\theta_1\cos\theta_2. \tag{10}$$

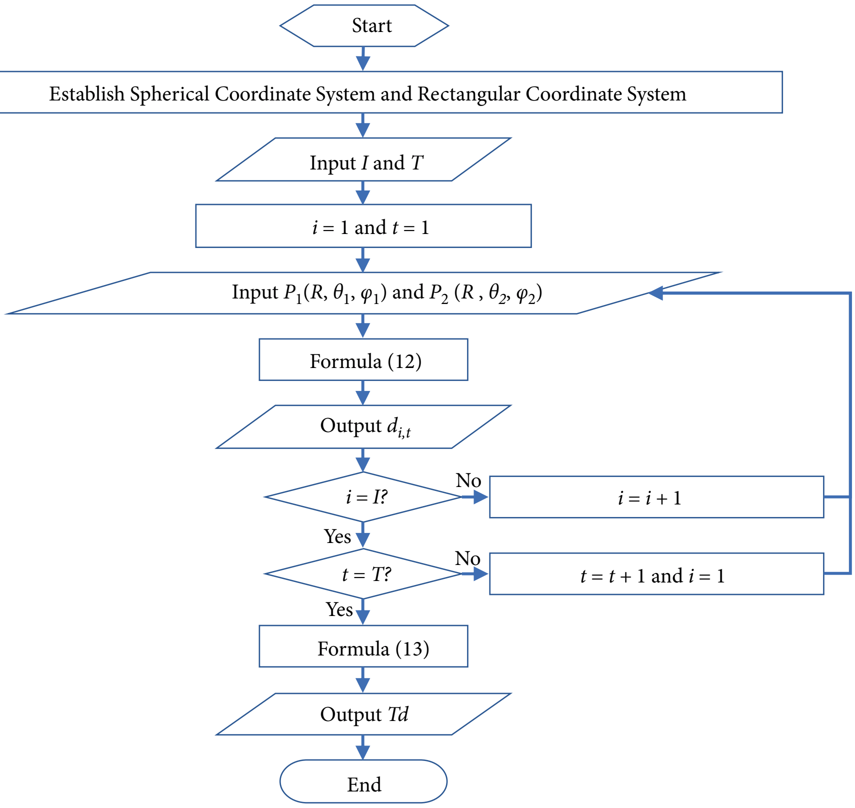


Figure 3: Flow chart of original model.

Fourth, considering the equation of arc length calculation, the distance between vectors $\overrightarrow{OP_1}$ and $\overrightarrow{OP_2}$ was calculated by Formula (11).

$$d = R\left\langle \overrightarrow{OP_1}, \overrightarrow{OP_2} \right\rangle. \tag{11}$$

Based on Formula (9), Formula (11) can be converted into Formula (12).

$$d = R \arccos\left(\sin\theta_1 \sin\theta_2 \cos\left(\varphi_2 - \varphi_1\right) + \cos\theta_1 \cos\theta_2\right). \tag{12}$$

In other words, the distance between $P_1(R, \theta_1, \varphi_1)$ and $P_2(R, \theta_2, \varphi_2)$ can be calculated approximately by the longitudes and latitudes of them based on GPS data, and the radius of the earth under Original Model.

Fifth, all the distances were summed up. Supposing that the variable Td represents total movements of vehicles under Original Model and it can be calculated by Formula (13).

$$Td = \sum_{i=1}^{I} \sum_{t=1}^{T} d_{i,t}. \tag{13}$$

3.4. Modified Digital Elevation Model. MDE adopts Original Model and SBI Model to improve the preciseness of results and it calculates distances in a comprehensive way (see Figure 4).

There are 5 steps in this section.

First, the Spherical Coordinate System was established in Formula (1).

Second, the Rectangular Coordinate System was established in Formula (4).

Third, all the unknown altitude values were calculated approximately by Spherical Bilinear Interpolation Model.

Fourth, the distance between P_1 and P_2 was calculated approximately by Modified Digital Elevation Model. Based on Formula (12), the distance between points $P_1(R + h_1, \theta_1, \varphi_1)$ and $P_2(R + h_2, \theta_2, \varphi_2)$ can be calculated by Formula (14) when $\rho = R + h_1$.

$$\begin{aligned} d|_{\rho = R + h_1} &= \left(R + h_1\right) \arccos\left(\sin\theta_1 \sin\theta_2 \cos\left(\varphi_2 - \varphi_1\right)\right. \\ &\quad \left. + \cos\theta_1 \cos\theta_2\right). \end{aligned} \tag{14}$$

Based on Formula (12), the distance between points $P_1(R + h_1, \theta_1, \varphi_1)$ and $P_2(R + h_2, \theta_2, \varphi_2)$ can be calculated by Formula (15) when $\rho = R + h_2$.

$$d|_{\rho = R + h_2} = \left(R + h_2\right) \arccos\left(\sin\theta_1 \sin\theta_2 \cos\left(\varphi_2 - \varphi_1\right) + \cos\theta_1 \cos\theta_2\right). \tag{15}$$

Based on Distance Formula, the distance can be calculated by Formula (16).

$$d' = \sqrt{\left(\frac{d|_{\rho = R + h_1} + d|_{\rho = R + h_2}}{2}\right)^2 + (\Delta h)^2}. \tag{16}$$

Based on Formulas (14), (15), Formula (16) can be converted into Formula (17).

$$d' = \sqrt{\left[\left(R + \frac{h_1 + h_2}{2}\right) \arccos\left(\sin\theta_1 \sin\theta_2 \cos\left(\varphi_2 - \varphi_1\right) + \cos\theta_1 \cos\theta_2\right)\right]^2 + \left(h_2 - h_1\right)^2}. \tag{17}$$

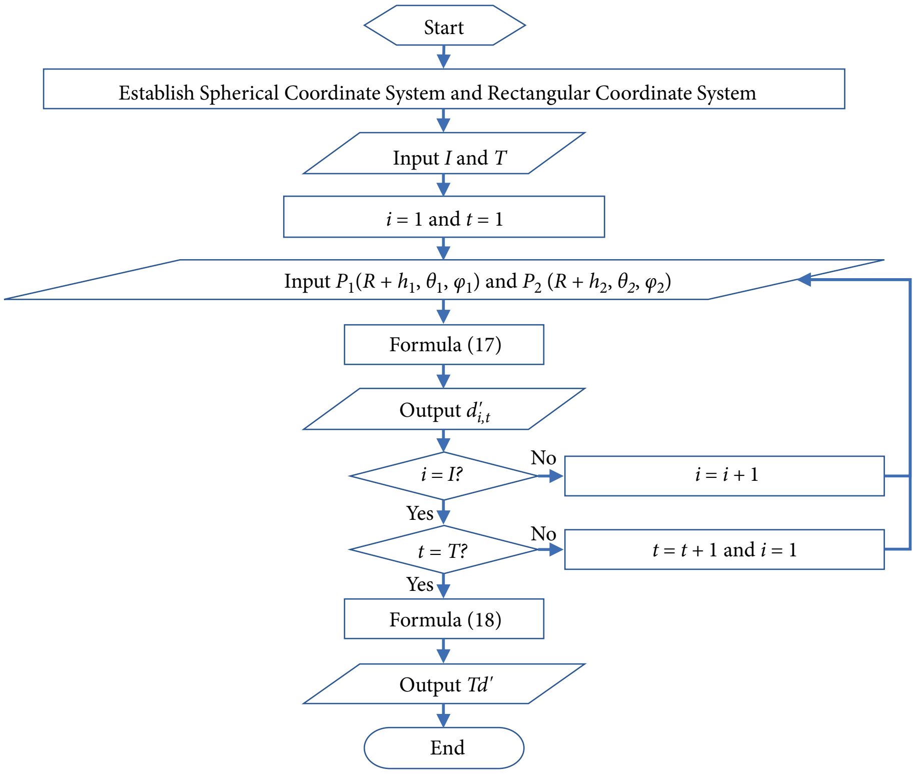


FIGURE 4: Flow chart of Modified Digital Elevation Model.

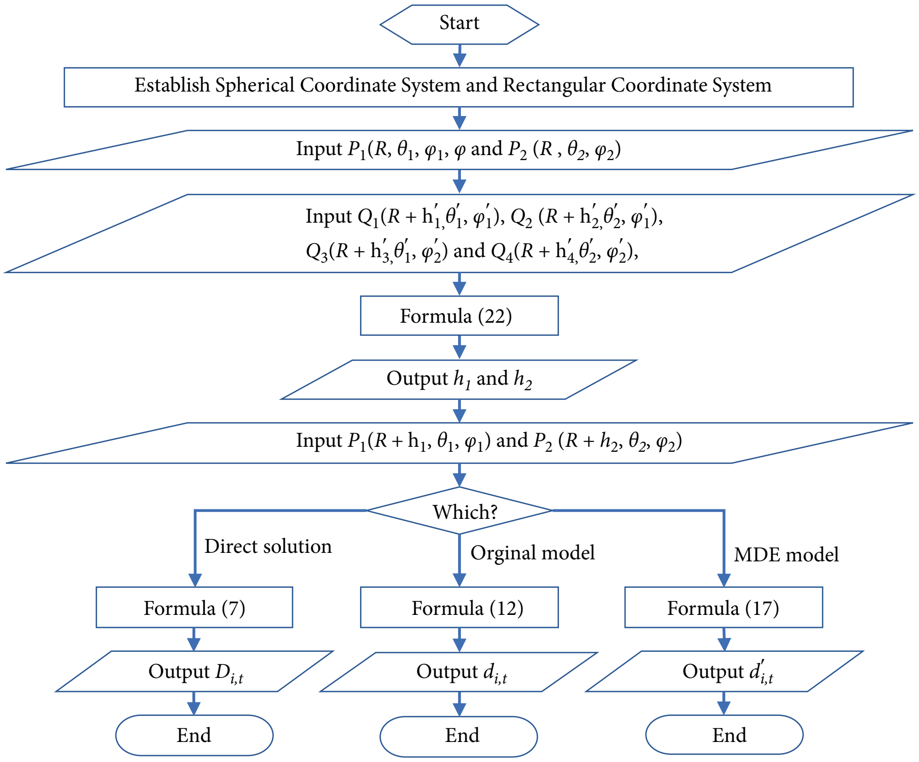

Start
Establish Spherical Coordinate System and Rectangular Coordinate System
Input $P_1(R, \theta_1, \varphi_1, \varphi$ and P_2 $(R , \theta_2, \varphi_2)$
Input $Q_1(R + \mathrm{h}'_{1,}\theta'_1, \varphi'_1)$, Q_2 $(R + \mathrm{h}'_{2,}\theta'_2, \varphi'_1)$,
$Q_3(R + \mathrm{h}'_{3,}\theta'_1, \varphi'_2)$ and $Q_4(R + \mathrm{h}'_{4,}\theta'_2, \varphi'_2)$,
Formula (22)
Output h_1 and h_2
Input $P_1(R + \mathrm{h}_1, \theta_1, \varphi_1)$ and P_2 $(R + h_2, \theta_2, \varphi_2)$
Which?
Direct solution
Orginal model
MDE model
Formula (7)
Formula (12)
Formula (17)
Output $D_{i,t}$
Output $d_{i,t}$
Output $d'_{i,t}$
End
End
End

FIGURE 5: Flow chart of Spherical Bilinear Interpolation Model and its applications.

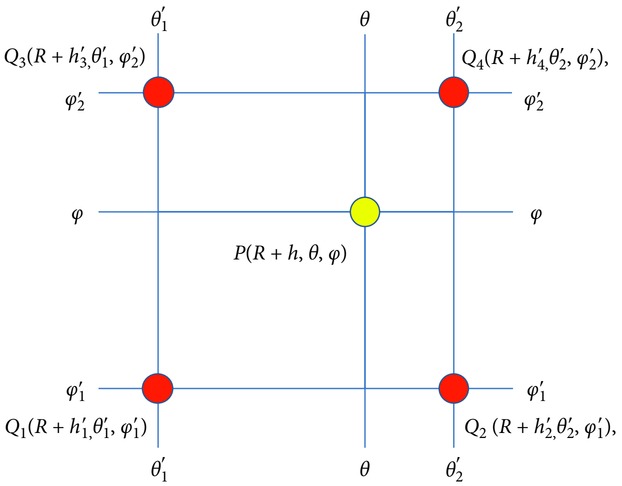


Figure 6: Schematic diagram of vertexes.

Fifth, all the distances were summed up. Supposing that the variable Td' represents total movements of vehicles under Modified Digital Elevation Model and it can be calculated by Formula (18).

$$Td' = \sum_{i=1}^{I}\sum_{t=1}^{T} d_{i,t}'. \tag{18}$$

3.5. Spherical Bilinear Interpolation Model. Spherical Bilinear Interpolation (SBI) Model helps improving the preciseness of GPS data with altitude especially when the data of altitudes is not precise enough or missing (see Figure 5).

There are 3 steps in this section.

First, find out 4 vertexes by positional notation based on the spherical coordinate of point P. For any point $P(R+h,\theta,\varphi)$ where $\theta_1' \le \theta \le \theta_2'$ and $\varphi_1' \le \varphi \le \varphi_2'$, there are 4 points $Q_1(R+h_1',\theta_1',\varphi_1')$, $Q_2(R+h_2',\theta_2',\varphi_1')$, $Q_3(R+h_3',\theta_1',\varphi_2')$, $Q_4(R+h_4',\theta_2',\varphi_2')$ around it (see Figure 6).

Second, figure out the relationship among the spherical coordinates of 5 points P, Q_1, Q_2, Q_3, Q_4. Supposing that there are 4 factors $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ satisfying Formula (19).

$$\overrightarrow{OP} = \alpha_1\overrightarrow{OQ_1} + \alpha_2\overrightarrow{OQ_2} + \alpha_3\overrightarrow{OQ_3} + \alpha_4\overrightarrow{OQ_4}. \tag{19}$$

As a result, the solution of Formula (19) was settled by Formula (20). The derivation process of Formula (20) is in the Appendix (see Appendix B).

$$\begin{aligned}
\alpha_1 &= \frac{(\theta_2'-\theta)\times(\varphi_2'-\varphi)}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')},\\
\alpha_2 &= \frac{(\theta-\theta_1')\times(\varphi_2'-\varphi)}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')},\\
\alpha_3 &= \frac{(\theta_2'-\theta)\times(\varphi-\varphi_1')}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')},\\
\alpha_4 &= \frac{(\theta-\theta_1')\times(\varphi-\varphi_1')}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')}.
\end{aligned} \tag{20}$$

Based on Formula (20), Formula (19) can be converted into Formula (21).

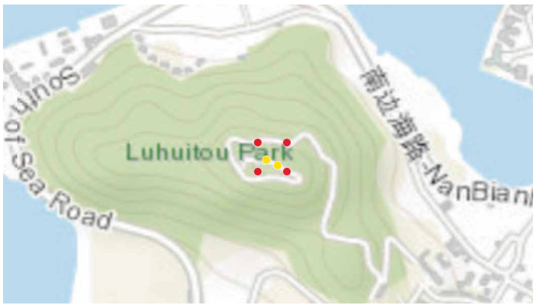


Figure 7: Actual position of the sample.

$$\begin{aligned}
\overrightarrow{OP} &= \frac{(\theta_2'-\theta)\times(\varphi_2'-\varphi)}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')}\overrightarrow{OQ_1}\\
&\quad + \frac{(\theta-\theta_1')\times(\varphi_2'-\varphi)}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')}\overrightarrow{OQ_2}\\
&\quad + \frac{(\theta_2'-\theta)\times(\varphi-\varphi_1')}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')}\overrightarrow{OQ_3}\\
&\quad + \frac{(\theta-\theta_1')\times(\varphi-\varphi_1')}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')}\overrightarrow{OQ_4}.
\end{aligned} \tag{21}$$

The checking process of Formula (21) is in the Appendix (see Appendix C). Based on Formula (C.24) in the Appendix C, the altitude value of point $P(R+h,\theta,\varphi)$ can be calculated by Formula (22).

$$\begin{aligned}
h &= \frac{(\theta_2'-\theta)\times(\varphi_2'-\varphi)}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')}\times h_1' + \frac{(\theta-\theta_1')\times(\varphi_2'-\varphi)}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')}\\
&\quad \times h_2' + \frac{(\theta_2'-\theta)\times(\varphi-\varphi_1')}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')}\times h_3' + \frac{(\theta-\theta_1')\times(\varphi-\varphi_1')}{(\theta_2'-\theta_1')\times(\varphi_2'-\varphi_1')}\times h_4'.
\end{aligned} \tag{22}$$

Third, substitute h_1', h_2', h_3', h_4' by the altitude value of points Q_1, Q_2, Q_3, Q_4. Finally, the altitude value of point P can be calculated approximately by Spherical Bilinear Interpolation Model.

3.6. Data and Tools

(i) Data Source. The GPS data of taxis in this study was collected from the big data platform, Travel Cloud, which was developed by Ministry of Transport of the People's Republic of China (see Data Availability). It was provided in part by the transportation department of Liaoning Province, in part by Henan Province department of transportation, in part by Sanya Traffic and Transportation Bureau of Hainan Province. Its major data items include anonymous vehicle ID, longitude, latitude and recording time. And the digital elevation data in this study was collected from Google Map.

(ii) Data Processing. In terms of GPS data processing, there are 3 steps. First, all the taxi locations were averaged every 60 seconds so as to improve the accuracy of data, that is to say, average locations of those taxis were recorded every 60 seconds entirely. Second, the flaw data was removed so as to ensure the integrity of

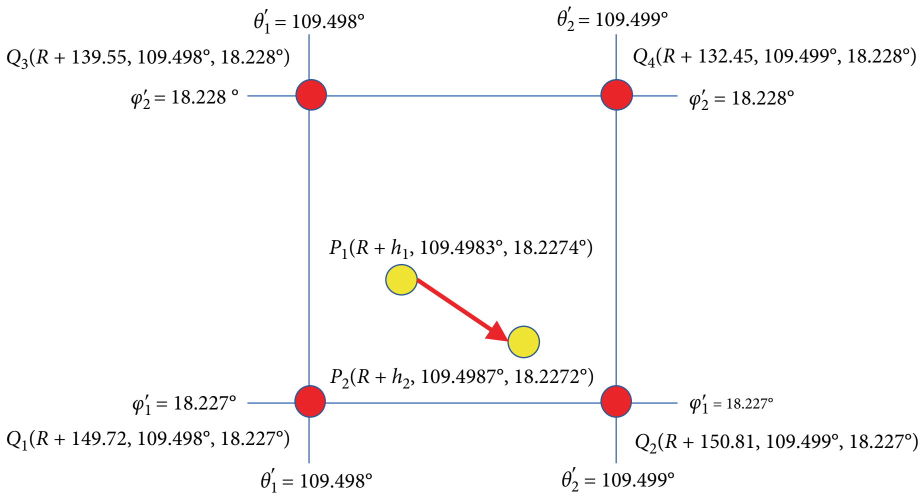

$\theta_1' = 109.498°$
$\theta_2' = 109.499°$
$Q_3(R + 139.55, 109.498°, 18.228°)$
$Q_4(R + 132.45, 109.499°, 18.228°)$
$\varphi_2' = 18.228°$
$\varphi_2' = 18.228°$
$P_1(R + h_1, 109.4983°, 18.2274°)$
$P_2(R + h_2, 109.4987°, 18.2272°)$
$\varphi_1' = 18.227°$
$\varphi_1' = 18.227°$
$Q_1(R + 149.72, 109.498°, 18.227°)$
$Q_2(R + 150.81, 109.499°, 18.227°)$
$\theta_1' = 109.498°$
$\theta_2' = 109.499°$

Figure 8: Schematic diagram of the sample.

data. Third, the obtained data was checked to avoid mistakes. As a result, GPS data without altitude was prepared. In terms of digital elevation data processing, there are 4 steps. First, the original digital elevation data was imported by Arcgis. Second, GPS data without altitude in a Microsoft Excel file was converted into a XY table of Arcgis geodatabase by the conversion tool called Excel to Table. Third, the XY table was adopted to create a point feature class on the basis of its coordinate system, World Geodetic System 1984 by the data management tool called Add XY Data. Fourth, the altitude data of the point feature class was extracted by the spatial analyst tool called Extract Values by Points. As a result, GPS data with altitude was prepared (see the file named "Data.xlsx" in the Supplementary Material).

(iii) Implement Tool. Several kinds of software were adopted in this study as follows. Rivermap X3.1 was applied for extracting the altitude from digital elevation data. Microsoft Excel 2019 was applied for original data processing and result storage. Stata 14.1 was applied for statistic analyzing. Matlab R2018a was applied for programming. Global Mapper 14.1 was applied for map drawing. Arcgis Pro 2.2.0 (trial version) was applied for map drawing and spatial analyzing.

4. Results

In this section, a sample (see Section 4.1) was presented for better description and vehicle movements were analyzed by Direct Solution, Original Model, MDE model at the same time, including Sanya (see Section 4.2), Zhengzhou (see Section 4.3), Liaoyang (see Section 4.4). And the results of them were compared with each other (see Section 4.5).

4.1. Sample. In the sample, there was a taxi around Luhuitou Park in Sanya of Hainan Province moving from the left yellow point to the right yellow point (see Figure 7). Luhuitou Park was on a hill and the taxi was on the road uphill. It indicated that the altitude of the taxi in the sample would rise from the left yellow point to the right yellow point.

In Figure 7, Luhuitou Park was surrounded by South of Sea Road, also known as Nanbianhai Road in Chinese. Furthermore, four red points surrounding yellow points were supplementary points drawn for SBI Model. Their longitudes and latitudes kept only 3 decimal digits. Actual position in Figure 3 can be converted into Schematic diagram (see Figure 8).

In Figure 8, there are 2 yellow points $P_1(R + h_1, 109.4983^\circ, 18.2274^\circ)$, $P_2(R + h_2, 109.4987^\circ, 18.2272^\circ)$ and 4 red points $Q_1(R + 152, 109.498^\circ, 18.227^\circ)$, $Q_2(R + 150.81, 109.499^\circ, 18.227^\circ)$, $Q_3(R + 139.55, 109.498^\circ, 18.228^\circ)$, $Q_4(R + 132.45, 109.499^\circ, 18.228^\circ)$.

According to the original digital elevation data, the altitude value of points P_1 and P_2 were 145 (m) and 147 (m) respectively. The taxi altitude in the sample changed approximately 2 (m) from P_1 to P_2. Based on SBI Model, the altitude value of points P_1 and P_2 can be further calculated by Formula (17). And the altitude value of points P_1 and P_2 adjusted by SBI Model were 145.00 (m) and 147.30 (m) respectively. Based on Formulas (7), (12), (17), the sample results and comparisons of three methods were figured out (see Table 2). The derivation process of Table 2 is in the Appendix (see Appendix D).

In Table 2, Value1 adopted original digital elevation data while Value2 adopted the digital elevation data adjusted by SBI Model. Under Value1, the vehicle movement $\overrightarrow{P_1P_2}$ under Direct Solution was 49.267562 (m). The deviation between Original Model and Direct Solution was 0.041677 (m) while the deviation between MDE Model and Direct Solution was only 0.000061 (m). Under Value2, the vehicle movement $\overrightarrow{P_1P_2}$ under Direct Solution was 49.280653 (m). The deviation between Original Model and Direct Solution was 0.054768 (m) while the deviation between MDE Model and Direct Solution was only 0.000062 (m). It can be seen that MDE Model had much less disparity with Direct Solution than Original Model in the sample results. In addition, even if the original digital elevation data, the altitude value of points P_1 and P_2 was

Table 2: Sample results and comparisons of three methods. Unit: m.

Method	Variable	Value1	Value2	$\lvert Value1 - Value2\rvert$
Direct Solution	D	49.267562	49.280653	0.013091
Original Model	d	49.225885	49.225885	0
MDE Model	d'	49.267623	49.280715	0.013092
(Comparisons)	$\lvert D - d\rvert$	0.041677	0.054768	0.013091
	$\lvert D - d'\rvert$	0.000061	0.000062	0.000001

Table 3: Sanya results and comparisons of three methods. Unit: m.

Method	Variable	Value1	Value2	$\lvert Value1 - Value2\rvert$
Direct Solution	TD	34,065,354.87	34,064,690.71	664.16
Original Model	Td	34,058,703.99	34,058,703.99	0
MDE Model	Td'	34,065,343.29	34,064,679.28	664.01
(Comparisons)	$\lvert TD - Td\rvert$	6,650.88	5,986.72	664.16
	$\lvert TD - Td'\rvert$	11.58	11.43	0.15

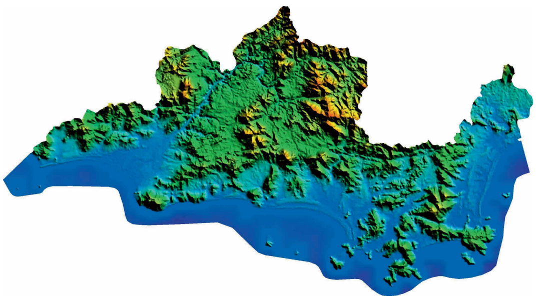


Figure 9: Elevation map of Sanya.

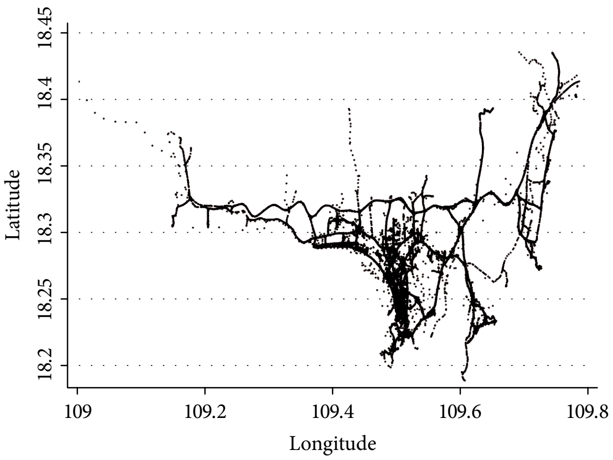


Figure 10: Taxi positions of Sanya.

missing, it could be further calculated by SBI Model. By contrast, the deviation between Original Model and Direct Solution was 0.013091 (m) while the deviation between MDE Model and Direct Solution was only 0.000001 (m). It further verified the advantage of MDE Model even though some data was missing in the sample results and SBI Model was adopted to improve the preciseness of GPS data with altitude.

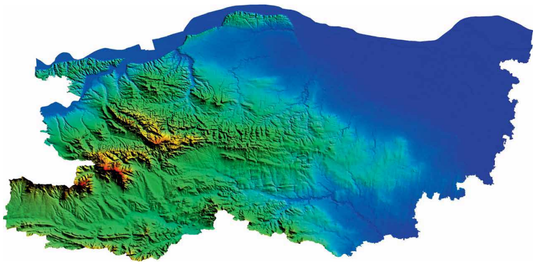


Figure 11: Elevation map of Zhengzhou.

The deviation between Original Model and MDE Model may be more obvious when sample size increases.

4.2. Sanya. Sanya is a city in Hainan Province where the longitude was in the range of [108.928°, 109.807°] while the latitude was in the range of [18.144°, 18.625°]. It located in the southern China and its altitudes had differences. (see Figure 9).

In Figure 9, colors reflected the elevation. The altitudes of blue area were low. The altitudes of green area were medium. The altitudes of yellow area were high.

In the origin data of Sanya, the locations of 2,506 taxis were recorded every 15 seconds from 9:00 a.m. to 9:59 a.m. on Nov. 15th in 2016, adding up to 766,042 records (see Figure 10).

In Figure 10, taxi positions of Sanya located minutely at a fixed monitor where the longitude was in the range of [109.0°, 109.8°] while the latitude was in the range of [18.1°, 18.5°]. After data processing, there were 2,191 taxis and 131,460 records left in the experimental data. And 107,427 movements of 2,096 taxis were extracted as a result. Based on Formulas (7), (12), (17), the Sanya results and comparisons of three methods were figured out (see Table 3).

In Table 3, Value1 adopted original digital elevation data while Value2 adopted the digital elevation data adjusted by SBI Model. Under Value1, total vehicle movements under Direct Solution was $34,065,354.87$ (m). The deviation between Original Model and Direct Solution was $6,650.88$ (m) while the deviation between MDE Model and Direct Solution was only

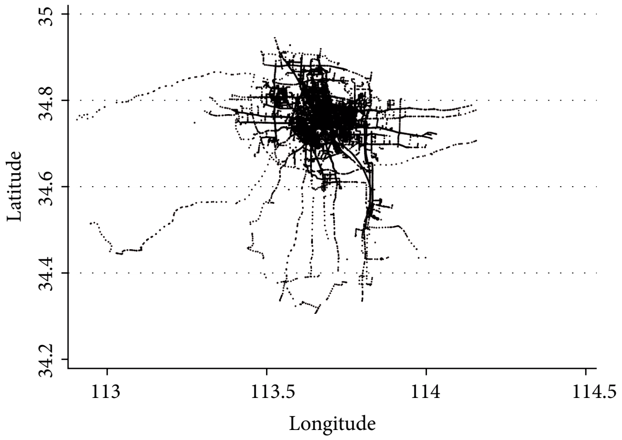


Figure 12: Taxi positions of Zhengzhou.

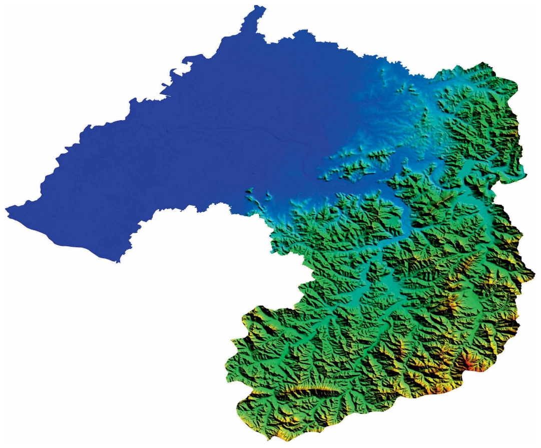


Figure 13: Elevation map of Liaoyang.

Table 4: Zhengzhou results and comparisons of three methods. Unit: m.

Method	Variable	Value1	Value2	|Value1 − Value2|
Direct Solution	TD	94,767,484.51	94,767,008.47	476.04
Original Model	Td	94,764,096.35	94,764,096.35	0
MDE Model	Td'	94,767,481.20	94,767,005.17	476.03
(Comparisons)	$\|TD - Td\|$	3,388.16	2,912.12	476.04
	$\|TD - Td'\|$	3.31	3.30	0.01

11.58 (m). Under Value2, total vehicle movements under Direct Solution was 34, 064, 690.71 (m). The deviation between Original Model and Direct Solution was 5, 986.72 (m) while the deviation between MDE Model and Direct Solution was only 11.43 (m). It can be seen that the deviation between Original Model and MDE Model in Sanya was much more obvious than sample. The results in Sanya verified that MDE Model had much less disparity with Direct Solution than Original Model. In addition, even if some original digital elevation data of actual positions in Sanya were missing, it could be further calculated by SBI Model. By contrast, the deviation between Original Model and Direct Solution was 664.16 (m) while the deviation between MDE Model and Direct Solution was only 0.15 (m). It further verified the advantage of MDE Model even though some data was missing in Sanya and SBI Model was adopted to improve the preciseness of GPS data with altitude.

4.3. Zhengzhou. Zhengzhou is a city in Henan Province where the longitude was in the range of [112.714°, 114.206°] while the latitude was in the range of [34.262°, 34.985°]. It located in the midland China and its altitudes had differences. (see Figure 11).

In Figure 11, colors reflected the elevation. The altitudes of blue area were low. The altitudes of green area were medium. The altitudes of yellow area were high.

In the origin data of Zhengzhou, the locations of 9,703 taxis were recorded every 15 seconds from 14:50 p.m. to 15:38 p.m. on Nov. 15th in 2016, adding up to 1,048,575 records (see Figure 12).

In Figure 12, taxi positions of Zhengzhou located minutely at a fixed monitor where the longitude was in the range of [112.8°, 114.5°] while the latitude was in the range of [34.2°, 35.0°]. After data processing, there were 5,774 taxis and 282,926 records left in the experimental data. And 229,370 movements of 5,597 taxis were extracted as a result. Based on Formulas (7), (12), (17), the Zhengzhou results and comparisons of three methods were figured out (see Table 4).

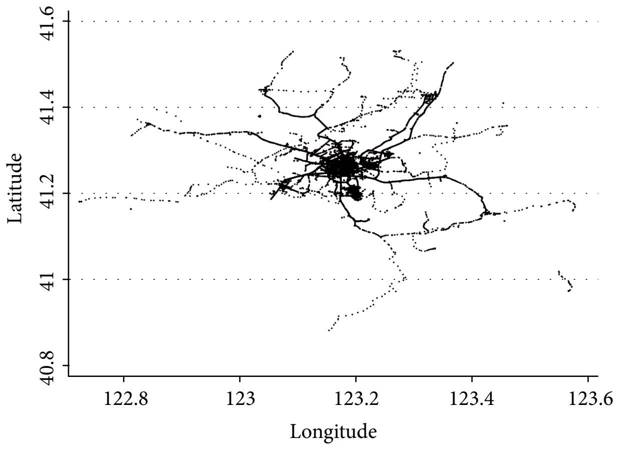


Figure 14: Taxi positions of Liaoyang.

In Table 4, Value1 adopted original digital elevation data while Value2 adopted the digital elevation data adjusted by SBI Model. Under Value1, total vehicle movements under Direct Solution was 94,767,484.51 (m). The deviation between Original Model and Direct Solution was 3,388.16 (m) while the deviation between MDE Model and Direct

Table 5: Liaoyang results and comparisons of three methods. Unit: m.

Method	Variable	Value1	Value2	|Value1 − Value2|
Direct Solution	TD	29,796,201.87	29,796,059.67	142.20
Original Model	Td	29,795,356.19	29,795,356.19	0
MDE Model	Td'	29,796,201.11	29,796,058.91	142.20
(Comparisons)	$\|TD - Td\|$	845.68	703.48	142.20
	$\left\|TD - Td'\right\|$	0.76	0.76	0.00

Table 6: Comprehensive results and comparisons of three methods. Unit: m.

Variable	Data	Sanya	Zhengzhou	Liaoyang	Total
$\|TD - Td\|$	Value1	6,650.88	3,388.16	845.68	10,884.72
	Value2	5,986.72	2,912.12	703.48	9,602.32
$\left\|TD - Td'\right\|$	Value1	11.58	3.31	0.76	15.65
	Value2	11.43	3.30	0.76	15.49
$\frac{\|TD-Td'\|}{\|TD-Td\|}$	Value1	0.17%	0.10%	0.09%	0.14%
	Value2	0.19%	0.11%	0.11%	0.16%

Solution was only 3.31 (m). Under Value2, total vehicle movements under Direct Solution was 94,767,008.47 (m). The deviation between Original Model and Direct Solution was 2,912.12 (m) while the deviation between MDE Model and Direct Solution was only 3.30 (m). It can be seen that the deviation between Original Model and MDE Model in Zhengzhou was much more obvious than sample. The results in Zhengzhou verified that MDE Model had much less disparity with Direct Solution than Original Model. In addition, even if some original digital elevation data of actual positions in Zhengzhou were missing, it could be further calculated by SBI Model. By contrast, the deviation between Original Model and Direct Solution was 476.04 (m) while the deviation between MDE Model and Direct Solution was only 0.01 (m). It further verified the advantage of MDE Model even though some data was missing in Zhengzhou and SBI Model was adopted to improve the preciseness of GPS data with altitude.

4.4. Liaoyang. Liaoyang is a city in Liaoning Province where the longitude was in the range of [112.588°, 123.684°] while the latitude was in the range of [40.710°, 41.615°]. It located in the northern China and its altitudes had differences. (see Figure 13).

In Figure 13, colors reflected the elevation. The altitudes of blue area were low. The altitudes of green area were medium. The altitudes of yellow area were high.

In the origin data of Liaoyang, the locations of 2,237 taxis were recorded every 30 seconds from 9:59 a.m. to 10:58 a.m. on Aug. 8th in 2016, adding up to 268,440 records (see Figure 14).

In Figure 14, taxi positions of Liaoyang located minutely at a fixed monitor where the longitude was in the range of [122.7°, 123.6°] while the latitude was in the range of [40.8°, 41.6°]. After data processing, there were 2,025 taxis and 121,440 records left in the experimental data. And 87,879 movements of 1,810 taxis were extracted as a result. Based on Formulas (7), (12), (17), the Liaoyang results and comparisons of three methods were figured out (see Table 5).

Table 7: Notations for variables.

Notation	Explanation
O	The center of coordinate systems/the center of the earth
P	The location
ρ	The radial distance
θ	The polar angle
φ	The azimuthal angle
R	The radius of the earth
h	The altitude
P_i	The location of vehicles
Q_i	The location near the location of vehicles
θ_i	The longitude of the location of vehicles
φ_i	The latitude of the location of vehicles
h_i	The altitude of the location of vehicles
i	The vehicle ID number
t	The time
I	The maximum of vehicle ID number
T	The maximum of time
D	Movements between two locations of vehicles under Direct Solution
d	Movements between two locations of vehicles under Original Model
d'	Movements between two locations of vehicles under Modified Digital Elevation Model
TD	Total movements under Direct Solution
Td	Total movements under Original Model
Td'	Total movements under Modified Digital Elevation Model

In Table 5, Value1 adopted original digital elevation data while Value2 adopted the digital elevation data adjusted by SBI Model. Under Value1, total vehicle movements under Direct Solution was 29,796,201.87 (m). The deviation between Original Model and Direct Solution was 845.68 (m) while the deviation between MDE Model and Direct Solution was only 0.76 (m). Under Value2, total vehicle movements under Direct Solution was 29,796,059.67 (m). The deviation between Original Model and Direct Solution was 703.48 (m) while the deviation between MDE Model and Direct Solution was only 0.76 (m). It can be seen that the deviation between Original Model and MDE Model in Liaoyang was much more obvious than sample. The results in Liaoyang verified that MDE Model had much less

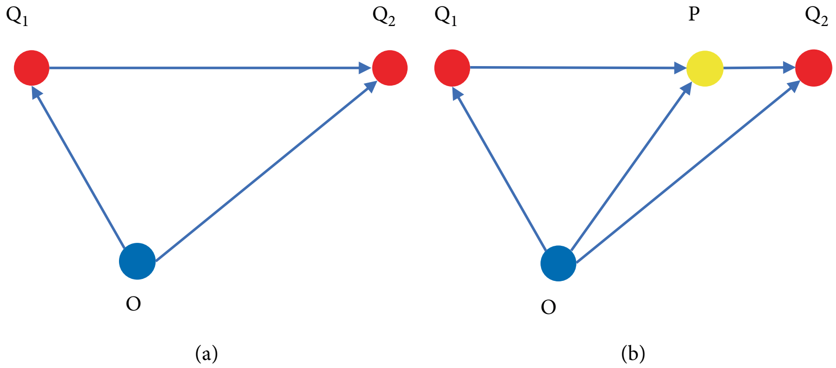


Figure 15: Schematic diagram for the derivation process of Formula (B.6). (a) Original condition. (b) Add the point P on the line Q_1Q_2.

disparity with Direct Solution than Original Model. In addition, even if some original digital elevation data of actual positions in Liaoyang were missing, it could be further calculated by SBI Model. By contrast, the deviation between Original Model and Direct Solution was 142.20 (m) while the deviation between MDE Model and Direct Solution was less than 0.01 (m). It further verified the advantage of MDE Model even though some data was missing in Liaoyang and SBI Model was adopted to improve the preciseness of GPS data with altitude.

4.5. Comparisons. Based on Tables 3–5, the Comprehensive results and comparisons of three methods were figured out (see Table 6).

In Table 6, Value1 adopted original digital elevation data while Value2 adopted the digital elevation data adjusted by SBI Model. Under Value1, total deviation between Original Model and Direct Solution was 10,884.72 (m) while total deviation between MDE Model and Direct Solution was only 15.65 (m). The latter is 0.14% that of the former. Under Value2, total deviation between Original Model and Direct Solution was 9,602.32 (m) while total deviation between MDE Model and Direct Solution was only 15.49 (m). The latter is 0.16% that of the former. Comprehensive results verified that MDE Model had over 99% less disparity with Direct Solution than Original Model because MDE Model took altitude into consideration but Original Model did not. In other words, MDE Model was much more accurate than Original Model.

5. Discussion

In this section, several operational details over the course of this research were presented for fellow colleagues and follow-up studies (see Section 5.1). In addition, the authors discussed the application prospects in the field of automatic vehicles (see Section 5.2).

5.1. Operational Details. Over the course of this research, there were several operational details that had not been contemplated within, including software use and data processing, which are described as follows.

(i) During processing original data by Microsoft Excel 2019, we found that Excel had a processing limit that its maximum capacity was 1,048,576 rows. If original data exceeds this limit, we had to divide them into several pieces before processing. Otherwise, there were other software, Snapde for instance, suiting for the big data more than 1,048,576 rows. In this research, the biggest original data was 1,048,575 records in Zhengzhou (see Section 4.3). Therefore, we adopted Excel as a result.

(ii) During spatial analyzing by Arcgis Pro (trial version), we found that the function of the spatial analyst tool called Extract Values by Points was available in v2.2.0 but unavailable in v2.3.0. Some errors may exist in v2.3.0. Therefore, we adopted v2.2.0 as a result.

(iii) In SBI Model, the first step was to find out 4 vertexes by positional notation based on the spherical coordinate of point P. We tried both 3 decimal digits and 4 decimal digits and found that 3 decimal digits of longitudes and latitudes were sufficient for vertexes.

(iv) In this research, we defined that $R = 6378137.00$ (m), which represents the radius of the earth (see Section 3.2). However, the radius of the earth has differences in different places.

5.2. Application Prospects. In the field of automatic vehicles, vehicle security is always under the spotlight. And elevation changes have a significant effect on the security of automatic vehicles because it is difficult for them to respond as human beings when traffic changes. Most crucially, the specific calculations used to vehicle movement analyses must be not only accurate but also fast. Otherwise, those automatic vehicles will be in danger, especially when they are running at high speed. Thus, appropriate counter measures could be taken to suppress or reverse or at least alleviate its limitations. The application prospects had not been contemplated within former sections, including MDE Model and SBI Model, which are described as follows.

(i) MDE Model. In last section, results verified that altitude was a key element of vehicle movement analyses, and it indicated that MDE Model was much more accurate than Original Model in terms of vehicle movement analysis (see Section 4.2–4.5). It is technically feasible to send alerts to the automatic vehicles

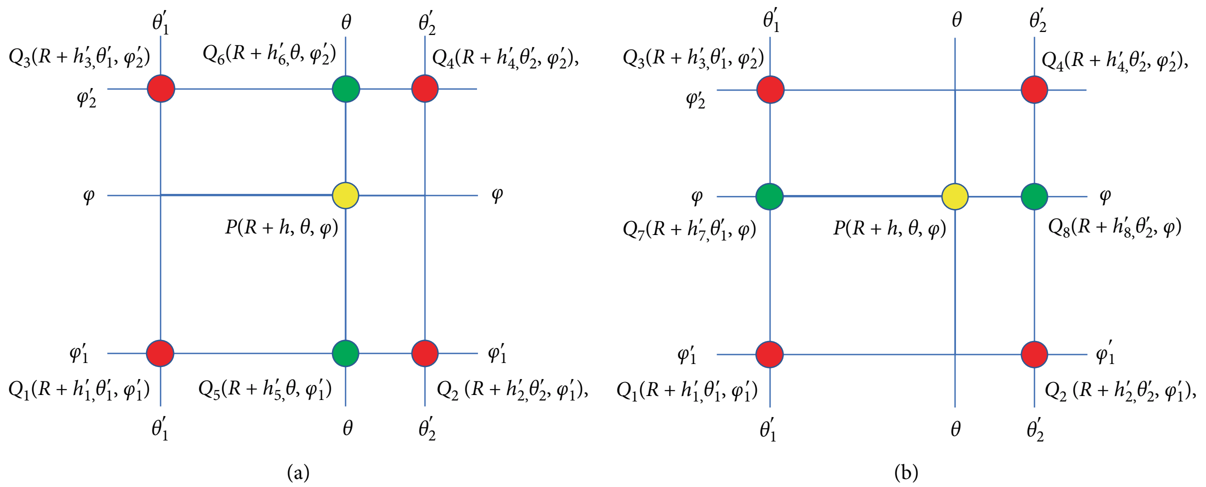

θ_1'
θ
θ_2'
$Q_3(R+h_3',\theta_1',\varphi_2')$
$Q_6(R+h_6',\theta,\varphi_2')$
$Q_4(R+h_4',\theta_2',\varphi_2')$,
φ_2'
φ
φ
$P(R+h,\theta,\varphi)$
φ_1'
φ_1'
$Q_1(R+h_1',\theta_1',\varphi_1')$
$Q_5(R+h_5',\theta,\varphi_1')$
$Q_2\ (R+h_2',\theta_2',\varphi_1')$,
θ_1'
θ
θ_2'
(a)
θ_1'
θ
θ_2'
$Q_3(R+h_3',\theta_1',\varphi_2')$
$Q_4(R+h_4',\theta_2',\varphi_2')$,
φ_2'
φ
φ
$Q_7(R+h_7',\theta_1',\varphi)$
$P(R+h,\theta,\varphi)$
$Q_8(R+h_8',\theta_2',\varphi)$
φ_1'
φ_1'
$Q_1(R+h_1',\theta_1',\varphi_1')$
$Q_2\ (R+h_2',\theta_2',\varphi_1')$,
θ_1'
θ
θ_2'
(b)

FIGURE 16: Schematic diagram for the derivation process of Formula (19). (a) Add two green points Q_5 and Q_6 where their polar angles are equivalent to θ. (b) Add two green points Q_7 and Q_8 where their azimuthal angles are equivalent to φ.

on the road uphill or downhill in advance so that they can adjust their brake systems in time. Compared with Direct Solution, MDE Model reduced the computational costs by avoiding square algorithms. Compared with Original Model, MDE Model raised the computational accuracy by taking altitude into consideration. Thus, analyzing vehicle movements by MDE Model could be faster and more accurate and it has a bright future in the field of automatic vehicles.

(ii) SBI Model. In last section, results indicated that SBI Model could improve the preciseness of GPS data with altitude especially when some data was missing in terms of vehicle movement Analysis (see Section 4.2–4.4). It is technically feasible for automatic vehicles to establish an offline GIS database and evaluate real-time GIS information when complete data is difficult to acquire, in the boondocks for instance, so that they can make vehicle movement analyses continuously. SBI Model can be viewed as a supplementary for MDE Model. Thus, SBI Model is necessary in the field of vehicle movement analyses and it also has a bright future in the field of automatic vehicles.

To sum up, the authors discussed that MDE Model and SBI Model have great practical prospects and they are suitable and useful for autonomous vehicles, especially for automatic taxis.

6. Conclusions

In this study, the authors proposed Modified Digital Elevation (MDE) Model, Spherical Bilinear Interpolation (SBI) Model and adopted the experimental data of 9,990 GPS-enabled taxis in Sanya, Zhengzhou and Liaoyang to make movement comparisons by three methods, including Direct Solution, Original Model and MDE Model. MDE Model was for improving the accuracy of vehicle movement analyses. SBI Model was for further improving the preciseness of GPS data with altitude. Given the arguments above mentioned, the major findings in this article include several contents, which are described as follows.

(i) The application of MDE Model could improve the accuracy of vehicle movement analyses and it resulted in over 99% less disparity with Direct Solution than Original Model because it took altitude into consideration (see Section 4.2–4.5).

(ii) The application of SBI Model could improve the preciseness of GPS data with altitude especially when some data was missing (see Section 4.2–4.4).

(iii) The application of MDE Model and SBI Model could improve effects according to statistic results (see Section 5.2) and it has a bright future in the field of automatic vehicles.

The contributions of this study could be as follows:

(i) MDE Model was built to calculate vehicle movements by digital elevation data based on mathematical equations (see Section 3.4).

(ii) SBI Model was proposed and applied to improve the preciseness of GPS data with altitude of collaborative vehicles (see Section 3.5).

(iii) A sample was presented for better description (see Section 4.1).

(iv) It was verified that MDE Model had higher accuracy than the original (see Section 4.2–4.5).

(v) The Derivation Process and Checking Process of SBI Model was proposed (see Appendix C, D).

Future directions for research can be in two ways. The one is improving MDE Model and SBI Model. The other is expanding the data size.

Appendix

A. Notations for Variables

In Appendix A, notations for variables are presented (see Table 7).

B. Derivation Process of Formula (20)

In Appendix B, a series of formulas are built to find out the solution of Formula (19) and work out Formula (20).

Supposing that the point O is the center of the coordinate system and there are two points Q_1 and Q_2 (see Figure 15(a)). Supposing that the point P is on the line Q_1Q_2, and points Q_1, P and Q_2 are collinear (see Figure 15(b)).

Based on Vector Addition Formula, the relations between $\overrightarrow{OQ_1}$, $\overrightarrow{OQ_2}$, and $\overrightarrow{Q_1Q_2}$ can be expressed by Formula (B.1). In the same way, the relations between $\overrightarrow{OP}$, $\overrightarrow{OQ_1}$, and $\overrightarrow{Q_1P}$ can be expressed by Formula (B.2).

$$\overrightarrow{OQ_1} + \overrightarrow{Q_1Q_2} = \overrightarrow{OQ_2}, \tag{B.1}$$

$$\overrightarrow{OP} = \overrightarrow{OQ_1} + \overrightarrow{Q_1P}. \tag{B.2}$$

Supposing that λ is the proportion of P in the vector $\overrightarrow{Q_1Q_2}$ as Formula (B.3).

$$\lambda = \frac{|Q_1P|}{|Q_1Q_2|}. \tag{B.3}$$

Based on Vector Proportion Formula, the relations between $\overrightarrow{Q_1P}$ and $\overrightarrow{Q_1Q_2}$ can be expressed by Formula (B.4).

$$\overrightarrow{Q_1P} = \lambda\,\overrightarrow{Q_1Q_2}, \tag{B.4}$$

where $0 \le \lambda \le 1$. Based on Formulas (B.2, B.4), $\overrightarrow{OP}$ can be calculated by Formula (B.5).

$$\overrightarrow{OP} = \overrightarrow{OQ_1} + \lambda\left(\overrightarrow{OQ_2} - \overrightarrow{OQ_1}\right). \tag{B.5}$$

Formula (B.5) can be simplified into Formula (B.6).

$$\overrightarrow{OP} = (1-\lambda)\,\overrightarrow{OQ_1} + \lambda\,\overrightarrow{OQ_2}. \tag{B.6}$$

In order to find out the solution of Formula (18) based on Formula (B.6), there are two plans for the derivation process of Formula (19), called Plan A and Plan B.

Plan A adds two green points $Q_5(R+h'_5, \theta, \varphi'_1)$ and $Q_6(R+h'_6, \theta, \varphi'_2)$ on the basis of Figure 1 where their azimuthal angles are equivalent to φ (see Figure 16(a)). In this way, points Q_1, Q_5, Q_2 are collinear, points Q_3, Q_6, Q_4 are collinear, besides, points Q_5, P, Q_6 are collinear. Based on Formula (B.6), $\overrightarrow{OQ_5}$, $\overrightarrow{OQ_6}$, and $\overrightarrow{OP}$ can be calculated respectively by Formulas (B.7), (B.8), (B.9).

$$\overrightarrow{OQ_5} = \frac{|Q_5Q_2|}{|Q_1Q_2|}\overrightarrow{OQ_1} + \frac{|Q_1Q_5|}{|Q_1Q_2|}\overrightarrow{OQ_2}, \tag{B.7}$$

$$\overrightarrow{OQ_6} = \frac{|Q_6Q_4|}{|Q_3Q_4|}\overrightarrow{OQ_3} + \frac{|Q_3Q_6|}{|Q_3Q_4|}\overrightarrow{OQ_4}, \tag{B.8}$$

$$\overrightarrow{OP} = \frac{|PQ_6|}{|Q_5Q_6|}\overrightarrow{OQ_5} + \frac{|Q_5P|}{|Q_5Q_6|}\overrightarrow{OQ_6}. \tag{B.9}$$

Based on Formulas (B.7), (B.8), Formula (B.9) can be converted into Formula (B.10).

$$\begin{aligned}\overrightarrow{OP} &= \frac{|PQ_6|}{|Q_5Q_6|} \times \left(\frac{|Q_5Q_2|}{|Q_1Q_2|}\overrightarrow{OQ_1} + \frac{|Q_1Q_5|}{|Q_1Q_2|}\overrightarrow{OQ_2}\right) \\ &\quad + \frac{|Q_5P|}{|Q_5Q_6|} \times \left(\frac{|Q_6Q_4|}{|Q_3Q_4|}\overrightarrow{OQ_3} + \frac{|Q_3Q_6|}{|Q_3Q_4|}\overrightarrow{OQ_4}\right).\end{aligned} \tag{B.10}$$

Formula (B.10) can be converted into Formula (B.11).

$$\begin{aligned}\overrightarrow{OP} &= \frac{|PQ_6| \times |Q_5Q_2|}{|Q_5Q_6| \times |Q_1Q_2|}\overrightarrow{OQ_1} + \frac{|PQ_6| \times |Q_1Q_5|}{|Q_5Q_6| \times |Q_1Q_2|}\overrightarrow{OQ_2} \\ &\quad + \frac{|Q_5P| \times |Q_6Q_4|}{|Q_5Q_6| \times |Q_3Q_4|}\overrightarrow{OQ_3} + \frac{|Q_5P| \times |Q_3Q_6|}{|Q_5Q_6| \times |Q_3Q_4|}\overrightarrow{OQ_4}.\end{aligned} \tag{B.11}$$

Considering $|Q_1Q_2| = |Q_3Q_4| = \theta'_2 - \theta'_1$, $|Q_1Q_5| = |Q_3Q_6| = \theta - \theta'_1$ $|Q_5Q_2| = |Q_6Q_4| = \theta'_2 - \theta$, $|Q_5P| = \varphi - \varphi'_1$, and $|PQ_6| = \varphi'_2 - \varphi$, Formula (B.11) can be converted into Formula (B.12).

$$\begin{aligned}\overrightarrow{OP} &= \frac{(\varphi'_2 - \varphi) \times (\theta'_2 - \theta)}{(\varphi'_2 - \varphi'_1) \times (\theta'_2 - \theta'_1)}\overrightarrow{OQ_1} + \frac{(\varphi'_2 - \varphi) \times (\theta - \theta'_1)}{(\varphi'_2 - \varphi'_1) \times (\theta'_2 - \theta'_1)}\overrightarrow{OQ_2} \\ &\quad + \frac{(\varphi - \varphi'_1) \times (\theta'_2 - \theta)}{(\varphi'_2 - \varphi'_1) \times (\theta'_2 - \theta'_1)}\overrightarrow{OQ_3} + \frac{(\varphi - \varphi'_1) \times (\theta - \theta'_1)}{(\varphi'_2 - \varphi'_1) \times (\theta'_2 - \theta'_1)}\overrightarrow{OQ_4}.\end{aligned} \tag{B.12}$$

Plan B adds two green points $Q_7(R+h'_7, \theta'_1, \varphi)$ and $Q_8(R+h'_8, \theta'_2, \varphi)$ on the basis of Figure 1 where their azimuthal angles are equivalent to φ (see Figure 16(b)). In this way, points Q_1, Q_7, Q_3 are collinear, points Q_2, Q_8, Q_4 are collinear, besides, points Q_7, P, Q_8 are collinear. Based on Formula (B.6), $\overrightarrow{OQ_7}$, $\overrightarrow{OQ_8}$, and $\overrightarrow{OP}$ can be calculated respectively by Formulas (B.13), (B.14), (B.15).

$$\overrightarrow{OQ_7} = \frac{|Q_7Q_3|}{|Q_1Q_3|}\overrightarrow{OQ_1} + \frac{|Q_1Q_7|}{|Q_1Q_3|}\overrightarrow{OQ_3}, \tag{B.13}$$

$$\overrightarrow{OQ_8} = \frac{|Q_8Q_4|}{|Q_2Q_4|}\overrightarrow{OQ_2} + \frac{|Q_2Q_8|}{|Q_2Q_4|}\overrightarrow{OQ_4}, \tag{B.14}$$

$$\overrightarrow{OP} = \frac{|PQ_8|}{|Q_7Q_8|}\overrightarrow{OQ_7} + \frac{|Q_7P|}{|Q_7Q_8|}\overrightarrow{OQ_8}. \tag{B.15}$$

Based on Formulas (B.13), (B.14), Formula (B.15) can be converted into Formula (B.16).

$$\begin{aligned}\overrightarrow{OP} &= \frac{|PQ_8|}{|Q_7Q_8|} \times \left(\frac{|Q_7Q_3|}{|Q_1Q_3|}\overrightarrow{OQ_1} + \frac{|Q_1Q_7|}{|Q_1Q_3|}\overrightarrow{OQ_3}\right) \\ &\quad + \frac{|Q_7P|}{|Q_7Q_8|} \times \left(\frac{|Q_8Q_4|}{|Q_2Q_4|}\overrightarrow{OQ_2} + \frac{|Q_2Q_8|}{|Q_2Q_4|}\overrightarrow{OQ_4}\right).\end{aligned} \tag{B.16}$$

Formula (B.16) can be converted into Formula (B.17).

$$\begin{aligned}\overrightarrow{OP} &= \frac{|PQ_8| \times |Q_7Q_3|}{|Q_7Q_8| \times |Q_1Q_3|}\overrightarrow{OQ_1} + \frac{|Q_7P| \times |Q_8Q_4|}{|Q_7Q_8| \times |Q_2Q_4|}\overrightarrow{OQ_2} \\ &\quad + \frac{|PQ_8| \times |Q_1Q_7|}{|Q_7Q_8| \times |Q_1Q_3|}\overrightarrow{OQ_3} + \frac{|Q_7P| \times |Q_2Q_8|}{|Q_7Q_8| \times |Q_2Q_4|}\overrightarrow{OQ_4}.\end{aligned} \tag{B.17}$$

Considering $|Q_1Q_3| = |Q_2Q_4| = \varphi_2' - \varphi_1'$, $|Q_1Q_7| = |Q_2Q_8| = \varphi - \varphi_1'$ $|Q_7Q_3| = |Q_8Q_4| = \varphi_2' - \varphi, |Q_7P| = \theta - \theta_1'$, and $|PQ_8| = \theta_2' - \theta$, Formula (B.17) can be converted into Formula (B.18).

$$\begin{aligned}\overrightarrow{OP} &= \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}\overrightarrow{OQ_1} + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}\overrightarrow{OQ_2} \\ &\quad + \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}\overrightarrow{OQ_3} + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}\overrightarrow{OQ_4}.\end{aligned} \tag{B.18}$$

Compare Formula (B.12) with Formula (B.18), it can be concluded that plan A and plan B have the same result. And then α_1, α_2, α_3, and α_4 can be calculated respectively by Formulas (B.19), (B.20), (B.21), (B.22).

$$\alpha_1 = \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{B.19}$$

$$\alpha_2 = \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{B.20}$$

$$\alpha_3 = \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{B.21}$$

$$\alpha_4 = \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}. \tag{B.22}$$

As a result, the solution of Formula (19) was settled by Formula (20) on the basis of Formulas (B.19), (B.20), (B.21), (B.22).

C. Checking Process of Formula (21)

In Appendix C, a series of formulas are built to check Formula (21).

Based on Figure 2, Formula (21) can be converted into Formula (C.1).

$$\begin{aligned}\overrightarrow{OP} &= \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}(R + h_1', \theta_1', \varphi_1') \\ &\quad + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}(R + h_2', \theta_2', \varphi_1') \\ &\quad + \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}(R + h_3', \theta_1', \varphi_2') \\ &\quad + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}(R + h_4', \theta_2', \varphi_2'),\end{aligned} \tag{C.1}$$

where $\theta_2' - \theta_1' > 0$ and $\varphi_2' - \varphi_1' > 0$.

Based on Formula (C.1), the radial distance P_ρ can be calculated by Formula (C.2).

$$\begin{aligned}P_\rho &= \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}(R + h_1') \\ &\quad + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}(R + h_2') \\ &\quad + \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}(R + h_3') \\ &\quad + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}(R + h_4').\end{aligned} \tag{C.2}$$

Formula (C.2) can be simplified into Formulas (C.3), (C.4), (C.5), (C.6), (C.7) in a step-by-step process and converted into Formula (C.8) as a result.

$$\begin{aligned}P_\rho &= R \times \left[\frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}\right. \\ &\quad \left. + \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}\right] \\ &\quad + \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_1' + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \\ &\quad \times h_2' + \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_3' + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_4',\end{aligned} \tag{C.3}$$

$$\begin{aligned}P_\rho &= R \times \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi) + (\theta - \theta_1') \times (\varphi_2' - \varphi) + (\theta_2' - \theta) \times (\varphi - \varphi_1') + (\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \\ &\quad + \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_1' + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_2' + \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_3' \\ &\quad + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_4',\end{aligned} \tag{C.4}$$

$$\begin{aligned}P_\rho &= R \times \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi + \varphi - \varphi_1') + (\theta - \theta_1') \times (\varphi_2' - \varphi + \varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} + \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_1' \\ &\quad + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_2' + \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_3' + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_4',\end{aligned} \tag{C.5}$$

$$\begin{aligned}P_\rho = R &\times \frac{(\varphi_2' - \varphi_1') \times (\theta_2' - \theta + \theta - \theta_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} + \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \\ &\times h_1' + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \\ &\times h_2' + \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \\ &\times h_3' + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_4',\end{aligned} \tag{C.6}$$

$$\begin{aligned}P_\rho = R &\times \frac{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} + \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_1' \\ &+ \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_2' + \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \\ &\times h_3' + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_4',\end{aligned} \tag{C.7}$$

$$\begin{aligned}P_\rho = R &+ \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_1' + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_2' \\ &+ \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_3' + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times h_4'.\end{aligned} \tag{C.8}$$

Based on Formula (C.1), the polar angle P_θ can be calculated by Formula (C.9).

$$\begin{aligned}P_\theta = &\frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times \theta_1' + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times \theta_2' \\ &+ \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times \theta_1' + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times \theta_2'.\end{aligned} \tag{C.9}$$

Formula (C.9) can be simplified into Formulas (C.10), (C.11), (C.12), (C.13), (C.14) in a step-by-step process and converted into Formula (C.15) as a result.

$$P_\theta = \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi) \times \theta_1' + (\theta - \theta_1') \times (\varphi_2' - \varphi) \times \theta_2' + (\theta_2' - \theta) \times (\varphi - \varphi_1') \times \theta_1' + (\theta - \theta_1') \times (\varphi - \varphi_1') \times \theta_2'}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{C.10}$$

$$P_\theta = \frac{\theta_1' \times (\theta_2' - \theta) \times (\varphi_2' - \varphi + \varphi - \varphi_1') + \theta_2' \times (\theta - \theta_1') \times (\varphi_2' - \varphi + \varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{C.11}$$

$$P_\theta = \frac{(\varphi_2' - \varphi_1') \times [\theta_1' \times (\theta_2' - \theta) + \theta_2' \times (\theta - \theta_1')]}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{C.12}$$

$$P_\theta = \frac{(\varphi_2' - \varphi_1') \times (\theta_1' \times \theta_2' - \theta_1' \times \theta + \theta_2' \times \theta - \theta_2' \times \theta_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{C.13}$$

$$P_\theta = \frac{(\varphi_2' - \varphi_1') \times (\theta_2' - \theta_1') \times \theta}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{C.14}$$

$$P_\theta = \theta. \tag{C.15}$$

Based on Formula (C.1), the azimuthal angle P_φ can be calculated by Formula (C.16).

$$\begin{aligned}P_\varphi = &\frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times \varphi_1' + \frac{(\theta - \theta_1') \times (\varphi_2' - \varphi)}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times \varphi_1' \\ &+ \frac{(\theta_2' - \theta) \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times \varphi_2' + \frac{(\theta - \theta_1') \times (\varphi - \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')} \times \varphi_2'.\end{aligned} \tag{C.16}$$

Formula (C.16) can be simplified into Formulas (C.17), (C.18), (C.19), (C.20), (C.21) in a step-by-step process and converted into Formula (C.22) as a result.

$$P_\varphi = \frac{(\theta_2' - \theta) \times (\varphi_2' - \varphi) \times \varphi_1' + (\theta - \theta_1') \times (\varphi_2' - \varphi) \times \varphi_1' + (\theta_2' - \theta) \times (\varphi - \varphi_1') \times \varphi_2' + (\theta - \theta_1') \times (\varphi - \varphi_1') \times \varphi_2'}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{C.17}$$

$$P_\varphi = \frac{\varphi_1' \times (\varphi_2' - \varphi) \times (\theta_2' - \theta + \theta - \theta_1') + \varphi_2' \times (\varphi - \varphi_1') \times (\theta_2' - \theta + \theta - \theta_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{C.18}$$

$$P_\varphi = \frac{(\theta_2' - \theta_1') \times [\varphi_1' \times (\varphi_2' - \varphi) + \varphi_2' \times (\varphi - \varphi_1')]}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{C.19}$$

$$P_\varphi = \frac{(\theta_2' - \theta_1') \times (\varphi_1' \times \varphi_2' - \varphi_1' \times \varphi + \varphi_2' \times \varphi - \varphi_2' \times \varphi_1')}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{C.20}$$

$$P_\varphi = \frac{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1') \times \varphi}{(\theta_2' - \theta_1') \times (\varphi_2' - \varphi_1')}, \tag{C.21}$$

$$P_\varphi = \varphi. \tag{C.22}$$

Based on Formulas (C.8), (C.15), (C.22), Formula (C.1) can be converted into Formula (C.23).

$$\begin{aligned}\overrightarrow{OP} = \Bigg(R &+ \frac{(\theta_2' - \theta)\times(\varphi_2' - \varphi)}{(\theta_2' - \theta_1')\times(\varphi_2' - \varphi_1')}\times h_1' \\ &+ \frac{(\theta - \theta_1')\times(\varphi_2' - \varphi)}{(\theta_2' - \theta_1')\times(\varphi_2' - \varphi_1')}\times h_2' \\ &+ \frac{(\theta_2' - \theta)\times(\varphi - \varphi_1')}{(\theta_2' - \theta_1')\times(\varphi_2' - \varphi_1')}\times h_3' \\ &+ \frac{(\theta - \theta_1')\times(\varphi - \varphi_1')}{(\theta_2' - \theta_1')\times(\varphi_2' - \varphi_1')}\times h_4', \theta, \varphi \Bigg).\end{aligned} \tag{C.23}$$

Compare Formula (C.23) with the origin coordinate $P(R+h,\theta,\varphi)$, the equational relations can be expressed by Formula (C.24).

$$\begin{aligned} R &+ \frac{(\theta_2' - \theta)\times(\varphi_2' - \varphi)}{(\theta_2' - \theta_1')\times\left(\varphi_2' - \varphi_1'\right)} \times h_1' \\ &+ \frac{(\theta - \theta_1')\times(\varphi_2' - \varphi)}{(\theta_2' - \theta_1')\times(\varphi_2' - \varphi_1')} \times h_2' \\ &+ \frac{(\theta_2' - \theta)\times(\varphi - \varphi_1')}{(\theta_2' - \theta_1')\times(\varphi_2' - \varphi_1')} \times h_3' \\ &+ \frac{(\theta - \theta_1')\times(\varphi - \varphi_1')}{(\theta_2' - \theta_1')\times(\varphi_2' - \varphi_1')} \times h_4' = R + h, \\ &\theta = \theta, \\ &\varphi = \varphi. \end{aligned} \tag{C.24}$$

As a result, Formula (20) is available after checking and h can be calculated by Formula (21) on the basis of Formula (C.24).

D. Derivation Process of Table 2

In Appendix D, a series of formulas are built to work out Table 2. It is worth noting that all the θ and φ in degree measure should be converted into radian measure by multiplying $\pi/180^{\circ}$ before calculating.

Now, we calculate the movements.

Based on Formulas (7), (12), (21), the vehicle movement $\overrightarrow{P_1P_2}$ under Direct Solution, Original Model, MDE Model can be calculated respectively by Formulas (29), (30), (31).

$$\begin{aligned} D^2 = &\Bigg[(6378137.00 + 145) \times \sin\left(109.4983^{\circ} \times \frac{\pi}{180^{\circ}}\right) \\ &\times \cos\left(18.2274^{\circ} \times \frac{\pi}{180^{\circ}}\right) - (6378137.00 + 147) \\ &\times \sin\left(109.4987^{\circ} \times \frac{\pi}{180^{\circ}}\right) \times \cos\left(18.2272^{\circ} \times \frac{\pi}{180^{\circ}}\right)\Bigg]^2 \\ &+ \Bigg[(6378137.00 + 145) \times \sin\left(109.4983^{\circ} \times \frac{\pi}{180^{\circ}}\right) \\ &\times \sin\left(18.2274^{\circ} \times \frac{\pi}{180^{\circ}}\right) - (6378137.00 + 147) \\ &\times \sin\left(109.4987^{\circ} \times \frac{\pi}{180^{\circ}}\right) \times \sin\left(18.2272^{\circ} \times \frac{\pi}{180^{\circ}}\right)\Bigg]^2 \\ &+ \Bigg[(6378137.00 + 145) \times \cos\left(109.4983^{\circ} \times \frac{\pi}{180^{\circ}}\right) \\ &-(6378137.00 + 147) \times \cos\left(109.4987^{\circ} \times \frac{\pi}{180^{\circ}}\right)\Bigg]^2, \end{aligned} \tag{D.1}$$

$$\begin{aligned} d = 6378137.00 &\times \arccos\Bigg(\sin\left(109.4983^{\circ} \times \frac{\pi}{180^{\circ}}\right) \\ &\times \sin\left(109.4987^{\circ} \times \frac{\pi}{180^{\circ}}\right) \times \cos\left(\left(18.2272^{\circ} \times \frac{\pi}{180^{\circ}}\right)\right. \\ &\left.-\left(18.2274^{\circ} \times \frac{\pi}{180^{\circ}}\right)\right) + \cos\left(109.4983^{\circ} \times \frac{\pi}{180^{\circ}}\right) \\ &\times \cos\left(109.4987^{\circ} \times \frac{\pi}{180^{\circ}}\right)\Bigg), \end{aligned} \tag{D.2}$$

$$\begin{aligned} d'^2 = &\Bigg[\left(6378137.00 + \frac{145 + 147}{2}\right) \times \arccos \\ &\cdot\Bigg(\sin\left(109.4983^{\circ} \times \frac{\pi}{180^{\circ}}\right) \times \sin\left(109.4987^{\circ} \times \frac{\pi}{180^{\circ}}\right) \\ &\times \cos\left(\left(18.2272^{\circ} \times \frac{\pi}{180^{\circ}}\right) - \left(18.2274^{\circ} \times \frac{\pi}{180^{\circ}}\right)\right) \\ &+ \cos\left(109.4983^{\circ} \times \frac{\pi}{180^{\circ}}\right) \\ &\times \cos\left(109.4987^{\circ} \times \frac{\pi}{180^{\circ}}\right)\Bigg)\Bigg]^2 + (145 - 147)^2. \end{aligned} \tag{D.3}$$

Based on Formulas (D.1), (D.2), (D.3), the values of D, d, d' are certain as shown by Formula (D.4).

$$\begin{aligned} D &= 49.267562, \\ d &= 49.225885, \\ d' &= 49.267623. \end{aligned} \tag{D.4}$$

Now, Spherical Bilinear Interpolation Model is adopted. Based on Formula (17), h_1 can be calculated by Formulas (D.5).

$$\begin{aligned} h_1 = &\frac{(109.499^{\circ} - 109.4983^{\circ})\times(18.228^{\circ} - 18.2274^{\circ})}{(109.499^{\circ} - 109.498^{\circ})\times(18.228^{\circ} - 18.227^{\circ})} \times 149.72 \\ &+ \frac{(109.4983^{\circ} - 109.498^{\circ})\times(18.228^{\circ} - 18.2274^{\circ})}{(109.499^{\circ} - 109.498^{\circ})\times(18.228^{\circ} - 18.227^{\circ})} \times 150.81 \\ &+ \frac{(109.499^{\circ} - 109.4983^{\circ})\times(18.2274^{\circ} - 18.227^{\circ})}{(109.499^{\circ} - 109.498^{\circ})\times(18.228^{\circ} - 18.227^{\circ})} \times 139.55 \\ &+ \frac{(109.4983^{\circ} - 109.498^{\circ})\times(18.2274^{\circ} - 18.227^{\circ})}{(109.499^{\circ} - 109.498^{\circ})\times(18.228^{\circ} - 18.227^{\circ})} \times 132.45. \end{aligned} \tag{D.5}$$

Formula (D.5) can be simplified into Formulas (D.6) and converted into Formula (D.7) as a result.

$$\begin{aligned} h_1 &= 0.7 \times 0.6 \times 149.72 + 0.3 \times 0.6 \times 150.81 \\ &+ 0.7 \times 0.4 \times 139.55 + 0.3 \times 0.4 \times 132.45, \end{aligned} \tag{D.6}$$

$$h_1 = 144.9962 \approx 145.00\,(m). \tag{D.7}$$

Based on Formula (17), h_2 can be calculated by Formula (D.8).

$$\begin{aligned} h_2 = &\frac{(109.499^{\circ} - 109.4987^{\circ})\times(18.228^{\circ} - 18.2272^{\circ})}{(109.499^{\circ} - 109.498^{\circ})\times(18.228^{\circ} - 18.227^{\circ})} \times 149.72 \\ &+ \frac{(109.4987^{\circ} - 109.498^{\circ})\times(18.228^{\circ} - 18.2272^{\circ})}{(109.499^{\circ} - 109.498^{\circ})\times(18.228^{\circ} - 18.227^{\circ})} \times 150.81 \\ &+ \frac{(109.499^{\circ} - 109.4987^{\circ})\times(18.2272^{\circ} - 18.227^{\circ})}{(109.499^{\circ} - 109.498^{\circ})\times(18.228^{\circ} - 18.227^{\circ})} \times 139.55 \\ &+ \frac{(109.4987^{\circ} - 109.498^{\circ})\times(18.2272^{\circ} - 18.227^{\circ})}{(109.499^{\circ} - 109.498^{\circ})\times(18.228^{\circ} - 18.227^{\circ})} \times 132.45. \end{aligned} \tag{D.8}$$

Formula (D.8) can be simplified into Formulas (D.9) and converted into Formula (D.10) as a result.

$$\begin{aligned} h_2 &= 0.3 \times 0.8 \times 149.72 + 0.7 \times 0.8 \times 150.81 \\ &\quad + 0.3 \times 0.2 \times 139.55 + 0.7 \times 0.2 \times 132.45, \end{aligned} \tag{D.9}$$

$$h_2 = 147.3024 \approx 147.30\,(m). \tag{D.10}$$

Based on Formula (D.7), (D.10), the coordinates of points $P_1(R + 145.00, 109.4983^\circ, 18.2274^\circ)$, $P_2(R + 147.30, 109.4987^\circ, 18.2272^\circ)$ are certain.

Now, we calculate the movements.

Based on Formula (7), (12), (21), the vehicle movement $\overrightarrow{P_1P_2}$ under Direct Solution, Original Model, MDE Model can be calculated respectively by Formula (D.11), (D.12), (D.13).

$$\begin{aligned} D^2 = {} & \Big[(6378137.00 + 145.00) \times \sin\left(109.4983^\circ \times \frac{\pi}{180^\circ}\right) \\ & \times \cos\left(18.2274^\circ \times \frac{\pi}{180^\circ}\right) - (6378137.00 + 147.30) \\ & \times \sin\left(109.4987^\circ \times \frac{\pi}{180^\circ}\right) \times \cos\left(18.2272^\circ \times \frac{\pi}{180^\circ}\right)\Big]^2 \\ & + \Big[(6378137.00 + 145.00) \times \sin\left(109.4983^\circ \times \frac{\pi}{180^\circ}\right) \\ & \times \sin\left(18.2274^\circ \times \frac{\pi}{180^\circ}\right) - (6378137.00 + 147.30) \\ & \times \sin\left(109.4987^\circ \times \frac{\pi}{180^\circ}\right) \times \sin\left(18.2272^\circ \times \frac{\pi}{180^\circ}\right)\Big]^2 \\ & + \Big[(6378137.00 + 145.00) \times \cos\left(109.4983^\circ \times \frac{\pi}{180^\circ}\right) \\ & -(6378137.00 + 147.30) \times \cos\left(109.4987^\circ \times \frac{\pi}{180^\circ}\right)\Big]^2, \end{aligned} \tag{D.11}$$

$$\begin{aligned} d = {} & 6378137.00 \times \arccos\Big(\sin\left(109.4983^\circ \times \frac{\pi}{180^\circ}\right) \\ & \times \sin\left(109.4987^\circ \times \frac{\pi}{180^\circ}\right) \times \cos\Big(\left(18.2272^\circ \times \frac{\pi}{180^\circ}\right) \\ & -\left(18.2274^\circ \times \frac{\pi}{180^\circ}\right)\Big) + \cos\left(109.4983^\circ \times \frac{\pi}{180^\circ}\right) \\ & \times \cos\left(109.4987^\circ \times \frac{\pi}{180^\circ}\right)\Big), \end{aligned} \tag{D.12}$$

$$\begin{aligned} d'^2 = {} & \Big[\left(6378137.00 + \frac{145.00 + 147.30}{2}\right) \times \arccos \\ & \cdot\Big(\sin\left(109.4983^\circ \times \frac{\pi}{180^\circ}\right) \times \sin\left(109.4987^\circ \times \frac{\pi}{180^\circ}\right) \\ & \times \cos\Big(\left(18.2272^\circ \times \frac{\pi}{180^\circ}\right) - \left(18.2274^\circ \times \frac{\pi}{180^\circ}\right)\Big) \\ & + \cos\left(109.4983^\circ \times \frac{\pi}{180^\circ}\right) \times \cos\left(109.4987^\circ \times \frac{\pi}{180^\circ}\right)\Big)\Big]^2 \\ & + (145.00 - 147.30)^2, \end{aligned} \tag{D.13}$$

Based on Formulas (D.11), (D.12), (D.13), the values of D, d, d' are certain as shown by Formula (D.14).

$$\begin{aligned} D &= 49.280653, \\ d &= 49.225885, \\ d' &= 49.280715. \end{aligned} \tag{D.14}$$

Based on Formulas (D.4), (D.14), Table 2 was worked out as a result.

Data Availability

The data collected during the study were freely available at the website. https://transportdata.cn/traffictravel/open. In detail, the data of Liaoyang was freely available at the website https://transportdata.cn/traffictravel/open/detail?id=180, the data of Zhengzhou was freely available at the website https://transportdata.cn/traffictravel/open/detail?id=205, the data of Sanya was freely available at the website https://transportdata.cn/traffictravel/open/detail?id=188.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Authors' Contributions

Jiawei Gui contributed to conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, manuscript preparation for all versions, and funding acquisition in part. Qunqi Wu contributed to funding acquisition in part and project administration. The authors reviewed the results and approved the final version of the manuscript.

Acknowledgments

The authors are indebted to the Kamal D. Singh and anonymous referees for their thoughtful comments that have helped substantially improve this work. The authors are also indebted to Ramya Kabali, Pani Vignesh, and anonymous editors for providing excellent editing services. Jiawei Gui was rewarded by the National Scholarship of China for doctoral students and appreciated that. The work described in this article was supported in part by the Strategic Planning Research Project of Ministry of Transport of China [2018-7-9] and [2018-16-9], in part by the National Social Science Fund of China [17BJY139], in part by the Chang'an University Excellent Doctoral Dissertation Project of Chinese Universities Scientific Fund of China [300102239718].

Supplementary Materials

In this section, a file named "Data.xlsx" is presented. In this file, it recorded the major data used in this study, including Sample, Direct Solution, Original Model, MDE Model and the comparison results between them. Due to space limitation, tables in this file (819 KB in size) records only partial contents of full data (over 500 MB in size). (*Supplementary Materials*)

References

[1] K. Piamrat, A. Ksentini, J.-M. Bonnin, and C. Viho, "Radio resource management in emerging heterogeneous wireless networks," *Computer Communications*, vol. 34, no. 9, pp. 1066–1076, 2011.

[2] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[3] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Internet of things: vision, applications and research challenges," *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497–1516, 2012.

[4] L. D. Xu, W. He, and S. C. Li, "Internet of things in industries: a survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.

[5] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.

[6] F. Y. Wang, "Parallel control and management for intelligent transportation systems: concepts, architectures, and applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 630–638, 2010.

[7] O. Kaiwartya, A. H. Abdullah, Y. Cao, M. Prasad, C.-T. Lin, and X. Liu, "Internet of vehicles : motivation, layered architecture, network model, challenges, and future aspects," *IEEE Access*, vol. 4, pp. 5356–5373, 2016.

[8] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, "Connected vehicles: solutions and challenges," *IEEE Internet of Things Journal*, vol. 1, no. 4, pp. 289–299, 2014.

[9] F. Yang, S. Wang, J. Li, Z. Liu, and Q. Sun, "An overview of internet of vehicles," *China Communications*, vol. 11, no. 10, pp. 1–15, 2014.

[10] P. Rawat, K. D. Singh, and J. M. Bonnin, "Cognitive radio for M2M and internet of things: a Survey," *Computer Communications*, vol. 94, pp. 1–29, 2016.

[11] K. D. Singh, P. Rawat, and J. M. Bonnin, "Cognitive radio for vehicular ad Hoc networks (CR-VANETs): approaches and challenges," *EURASIP Journal on Wireless Communications and Networking*, vol. 49, no. 1, pp. 1–22, 2014.

[12] H. C. Chen, "TCABRP: a trust-based cooperation authentication bit-map routing protocol against insider security threats in wireless ad hoc networks," *IEEE Systems Journal*, vol. 11, no. 2, pp. 449–459, 2017.

[13] X. M. Huang, R. Yu, M. Pan, and L. Shu, "Secure roadside unit hotspot against eavesdropping based traffic analysis in edge computing based internet of vehicles," *IEEE Access*, vol. 6, pp. 62371–62383, 2018.

[14] T. A. Butt, R. Iqbal, S. C. Shah, and T. Umar, "Social internet of vehicles: architecture and enabling technologies," *Computers & Electrical Engineering*, vol. 69, pp. 68–84, 2018.

[15] M. Chen, Y. Tian, G. Fortino, J. Zhang, and I. Humar, "Cognitive internet of vehicles," *Computer Communications*, vol. 120, pp. 58–70, 2018.

[16] H. Hartenstein and K. P. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 164–171, 2008.

[17] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular ad hoc network," *Journal of Network and Computer Applications*, vol. 37, pp. 380–392, 2014.

[18] M. Chaqfeh, A. Lakas, and I. Jawhar, "A survey on data dissemination in vehicular ad hoc networks," *Vehicular Communications*, vol. 1, no. 4, pp. 214–225, 2014.

[19] A. Petrovskaya and S. Thrun, "Model based vehicle detection and tracking for autonomous urban driving," *Autonomous Robots*, vol. 26, no. 2–3, pp. 123–139, 2009.

[20] S. E. Shladover, C. A. Desoer, J. K. Hedrick et al., "Automatic vehicle control developments in the PATH program," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 114–130, 1991.

[21] T. Luettel, M. Himmelsbach, and H. J. Wuensche, "Autonomous ground vehicles-concepts and a path to the future," *Proceedings of the IEEE*, vol. 100, pp. 1831–1839, 2012.

[22] P. Koopman and M. Wagner, "Autonomous vehicle safety: an interdisciplinary challenge," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 1, pp. 90–96, 2017.

[23] H. W. Wang, F. You, X. N. Chu, X. Li, and X. Sun, "Research on customer marketing acceptance for future automatic driving-a case study in china city," *IEEE Access*, vol. 7, pp. 20938–20949, 2019.

[24] H.-C. Chen, I. You, C.-E. Weng, C.-H. Cheng, and Y.-F. Huang, "A security gateway application for end-to-end M2M communications," *Computer Standards & Interfaces*, vol. 44, pp. 85–93, 2016.

[25] H. Park, C. Lee, Y. S. Lee, and E.-J. Kim, "Performance analysis for contention adaptation of M2M devices with directional antennas," *Journal of Supercomputing*, vol. 72, no. 9, pp. 3387–3408, 2016.

[26] H. Park and E. J. Kim, "Location-oriented multiplexing transmission for capillary machine-to-machine systems," *Multimedia Tools and Applications*, vol. 75, no. 22, pp. 14707–14719, 2016.

[27] S. U. Rehman, M. A. Khan, M. Imran, T. A. Zia, and M. Iftikhar, "Enhancing quality-of-service conditions using a cross-layer paradigm for ad-hoc vehicular communication," *IEEE Access*, vol. 5, pp. 12404–12416, 2017.

[28] B. Fan, H. Tian, S. Zhu, Y. Chen, and X. Zhu, "Traffic-aware relay vehicle selection in millimeter-wave vehicle-to-vehicle communication," *IEEE Wireless Communications Letters*, vol. 8, no. 2, pp. 400–403, 2019.

[29] N. Lyamin, D. Kleyko, Q. Delooz, and A. Vinel, "Real-time jamming DoS detection in safety-critical V2V C-ITS using data mining," *IEEE Communications Letters*, vol. 23, no. 3, pp. 442–445, 2019.

[30] E. Han, H. P. Lee, S. Park, J. So, and I. Yun, "Optimal signal control algorithm for signalized intersections under a V2I communication environment," *Journal of Advanced Transportation*, vol. 2019, Article ID 6039741, 9 pages, 2019.

[31] D. Jia, D. Ngoduy, and H. L. Vu, "A multiclass microscopic model for heterogeneous platoon with vehicle-to-vehicle communication," *Transportmetrica B-Transport Dynamics*, vol. 7, no. 1, pp. 448–472, 2019.

[32] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[33] M. Aramrattana, T. Larsson, J. Jansson, and A. Nåbo, "A simulation framework for cooperative intelligent transport

systems testing and evaluation," *Transportation Research Part F-Traffic Psychology and Behaviour*, vol. 61, pp. 268–280, 2019.

[34] M. Elwekeil, T. Wang, and S. Zhang, "Deep learning for joint adaptations of transmission rate and payload length in vehicular networks," *Sensors*, vol. 19, no. 5, p. 1113, 2019.

[35] C. Lv, X. Hu, A. Sangiovanni-Vincentelli, Y. Li, C. M. Martinez, and D. Cao, "Driving-style-based codesign optimization of an automated electric vehicle: a cyber-physical system approach," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 4, pp. 2965–2975, 2019.

[36] J. Li, G. Luo, N. Cheng et al., "An end-to-end load balancer based on deep learning for vehicular network traffic control," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 953–966, 2019.

[37] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1624–1639, 2011.

[38] F. Frankel and R. Reid, "Big data: distilling meaning from data," *Nature*, vol. 455, no. 7209, p. 30, 2008.

[39] W. Los and J. Wood, "Dealing with data: upgrading infrastructure," *Science*, vol. 331, no. 6024, pp. 1515–1516, 2011.

[40] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): a vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[41] M. Whaiduzzaman, M. Sookhak, A. Gani, and R. Buyya, "A survey on vehicular cloud computing," *Journal of Network and Computer Applications*, vol. 40, pp. 325–344, 2014.

[42] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: a survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys and Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[43] Z. L. Ning, J. Huang, and X. J. Wang, "Vehicular fog computing: enabling real-time traffic management for smart cities," *IEEE Wireless Communications*, vol. 26, no. 1, pp. 87–93, 2019.

[44] J. Tang, S. Zhang, X. Chen, F. Liu, and Y. Zou, "Taxi trips distribution modeling based on entropy-maximizing theory: a case study in Harbin City-China," *Physica A-Statistical Mechanics and Its Applications*, vol. 493, pp. 430–443, 2018.

[45] J. Tang, J. Liang, S. Zhang, H. Huang, and F. Liu, "Inferring driving trajectories based on probabilistic model from large scale taxi GPS data," *Physica A-Statistical Mechanics and Its Applications*, vol. 506, pp. 566–577, 2018.

[46] J. Bao, P. Liu, X. Qin, and H. Zhou, "Understanding the effects of trip patterns on spatially aggregated crashes with large-scale taxi GPS data," *Accident Analysis and Prevention*, vol. 120, pp. 281–294, 2018.

[47] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, "Wireless sensor networks: a survey on recent developments and potential synergies," *Journal of Supercomputing*, vol. 68, no. 1, pp. 1–48, 2014.

[48] M. Gerla, E.-K. Lee, G. Pau, and U. Lee, "Internet of vehicles: from intelligent grid to autonomous cars and vehicular clouds," in *2014 IEEE World Forum on Internet of Things*, pp. 241–246, Seoul, South Korea, 2014.

[49] M. N. Rastgoo, B. Nakisa, A. Rakotonirainy, V. Chandran, and D. Tjondronegoro, "A critical review of proactive detection of driver stress levels based on multimodal measurements," *ACM Computing Surveys*, vol. 51, no. 5, pp. 35–35, 2019.

[50] S. Barua, M. U. Ahmed, C. Ahlstrom, and S. Begum, "Automatic driver sleepiness detection using EEG, EOG and contextual information," *Expert Systems with Applications*, vol. 115, pp. 121–135, 2019.

[51] M.-S. Chen, C.-P. Hwang, T.-Y. Ho et al., "Driving behaviors analysis based on feature selection and statistical approach: a preliminary study," *Journal of Supercomputing*, vol. 75, no. 4, pp. 2007–2026, 2019.

[52] A. M. El-Geneidy, J. Horning, and K. J. Krizek, "Analyzing transit service reliability using detailed data from automatic vehicular locator systems," *Journal of Advanced Transportation*, vol. 45, no. 1, pp. 66–79, 2011.

[53] Z. Ning, X. Hu, Z. Chen, and M. Zhou, B. Hu, J. Cheng, and M. S. Obaidat, "A cooperative quality-aware service access system for social internet of vehicles," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2506–2517, 2018.

[54] C. Yuan, D. Wu, D. Wei, and H. Liu, "Modeling and analyzing taxi congestion premium in congested cities," *Journal of Advanced Transportation*, vol. 2017, Article ID 2619810, 12 pages, 2017.

[55] J. Song, F. Chen, Q. Wu, W. Liu, F. Xue, and K. Du, "Optimization of passenger transportation corridor mode supply structure in regional comprehensive transport considering economic equilibrium," *Sustainability*, vol. 11, no. 4, pp. 1–18, 2019.

[56] W. Dou, W. Tang, S. Li, S. Yu, and K.-K. Raymond Choo, "A heuristic line piloting method to disclose malicious taxicab driver's privacy over GPS big data," *Information Sciences*, vol. 483, pp. 247–261, 2019.

[57] J. Tang, Y. Wang, W. Hao, F. Liu, H. Huang, and Y. Wang, "A mixed path size logit-based taxi customer-search model considering spatio-temporal factors in route choice," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–12, 2019.

[58] J. Gui and Q. Wu, "Taxi efficiency measurements based on motorcade-sharing model: evidence from GPS-equipped taxi data in Sanya," *Journal of Advanced Transportation*, vol. 2018, Article ID 4360516, 10 pages, 2018.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 3035741, 9 pages
https://doi.org/10.1155/2020/3035741

Research Article

Malware Detection in Self-Driving Vehicles Using Machine Learning Algorithms

Seunghyun Park and Jin-Young Choi

Graduate School of Information Security, Korea University, Seoul 02841, Republic of Korea

Correspondence should be addressed to Jin-Young Choi; choi@formal.korea.ac.kr

Received 26 July 2019; Revised 19 October 2019; Accepted 19 November 2019; Published 17 January 2020

Guest Editor: Hsing-Chung Chen

The recent trend for vehicles to be connected to unspecified devices, vehicles, and infrastructure increases the potential for external threats to vehicle cybersecurity. Thus, intrusion detection is a key network security function in vehicles with open connectivity, such as self-driving and connected cars. Specifically, when a vehicle is connected to an external device through a smartphone inside the vehicle or when a vehicle communicates with external infrastructure, security technology is required to protect the software network inside the vehicle. Existing technology with this function includes vehicle gateways and intrusion detection systems. However, it is difficult to block malicious code based on application behaviors. In this study, we propose a machine learning-based data analysis method to accurately detect abnormal behaviors due to malware in large-scale network traffic in real time. First, we define a detection architecture, which is required by the intrusion detection module to detect and block malware attempting to affect the vehicle via a smartphone. Then, we propose an efficient algorithm for detecting malicious behaviors in a network environment and conduct experiments to verify algorithm accuracy and cost through comparisons with other algorithms.

1. Introduction

As automobiles become more intelligent, so do transportation systems [1]. New business requirements in the automotive market and advances in automotive communication technology are increasing the connectivity of automobiles. This greater connectivity portends the increased likelihood of future automobile cyberattacks [2]. Therefore, it is necessary to prepare countermeasures for various attack vectors to combat threats to vehicle cybersecurity.

For example, in 2015, Miller and Valasek [3] remotely hacked a traveling Jeep Cherokee to control the audio, windshield wipers, steering and braking, revealing that an unprepared cybersecurity system can threaten driver safety. Furthermore, in 2016 and 2017, Keen Security Lab [4] hacked a Tesla vehicle to demonstrate security threats and potential attacks related to connected vehicles. Typically, connected vehicles are a closed environment that only accepts remote control commands in an authorized communication path, such as a server built by the manufacturer or dedicated applications published by the manufacturer. In a closed environment, unauthorized commands are blocked. However, recent self-driving vehicles share their control signals and internal data with not only the controllers inside the vehicle, but also various unspecified vehicles, infrastructures, and smart devices outside the vehicle in real time. Thus, vehicle network protection should be prioritized in open environments.

The security of a self-driving vehicle is directly related to passenger safety; therefore, it is necessary to comprehensively consider the various attack vectors against vehicles based on the integrity, availability, and confidentiality of their cybersecurity [5]. When a connected vehicle's software is updated, it is essential to verify the integrity of the software. Attackers may use malicious applications to illegally steal privileges or gain access, repackage the software installed in the vehicle by injecting malicious code, and induce the installation of maliciously modified applications. This malicious software looks the same as the authorized software, but malicious code contained in the modified applications can collect the user's input to steal account information, activate abnormal service ports, or retain authorization for the attacker to access later. Such malicious software can be even used as a medium for additional remote attacks through communication with the command and control server. Thus, it is important to protect vehicle software when either the vehicle is connected to an external device such as a smartphone via an interface inside the vehicle, or a communication channel is opened between the vehicle and surrounding infrastructure. Previous research has installed vehicle gateways which allow only authorized communication to the vehicles and introduced vehicle Intrusion Detection Systems (IDSs) to detect abnormal behaviors in the Controller Area Network (CAN) [6]. However, it is difficult for a gateway or IDS to block these actions in advance, as most malware and adware are behavior-based. In order to detect unknown threats, it is vital to introduce a technology that can detect abnormal behaviors and analyze anomalous indicators using data analysis technology.

In this study, we review the various security threats to self-driving vehicles imposed by malware in Android operating system (OS) and discuss a method for detecting such malware. In an embedded environment such as a vehicle, both response time and detection accuracy are key factors because resources are limited, and real-time responses are required. Therefore, we propose a machine learning-based detection model that can reduce analysis time and improve detection accuracy. The specific contributions of this research are as follows:

(i) We present a method for detecting adware and malware in a self-driving vehicle environment.

(ii) We define the intrusion detection module architecture required to detect malware and prevent it from affecting the vehicle through a smartphone.

(iii) We experimentally compare the detection accuracy and cost of different algorithms and present the most efficient algorithm.

First, we describe the security technology protecting the internal and external communication networks of self-driving vehicles. We then propose an architecture for an intrusion detection module that detects malicious behavior in the vehicle network based on machine learning. Then, we present an effective intrusion detection method and compare it with existing algorithms in experiments. Finally, we present the conclusions and future work.

2. Preliminaries

2.1. Vehicle-to-Device Communication. In the paradigm of vehicle-to-everything communication, communicating with a specific device is termed vehicle-to-device (V2D) communication [7]. Android-based smartphones are typical devices that communicate with a vehicle. Services that identify vehicle operational information or diagnose vehicle abnormalities via a smartphone are classified as performing V2D communication. Initially, to carry out these functions, vehicles were directly connected to an external device outside the vehicle through a universal serial bus connector or Bluetooth, and the data on the device were used. Because a direct wired connection from the vehicle to the device occurred only if the target vehicle was physically occupied, a hacker could not directly control multiple vehicles remotely, even if the vehicles were successfully stolen. Since then, vehicle manufacturers have installed telematics control units (TCUs) or connectivity control units (CCUs) in vehicles and implemented interfaces for remote control of vehicles that include communication functions. In addition, this service is not limited to the original equipment manufacturer. Global telecommunication companies or Internet of Things device manufacturers can also install Long-Term Evolution communication modules on the on-board diagnostics II terminal to collect and manage various data inside the vehicle. When the vehicle is connected to a server or smartphone through such a communication module, information from the vehicle can be transmitted externally. Similarly, it is also possible to control the vehicle by injecting commands to the vehicle from the outside. A connection to a smartphone or external communication device is used not only for convenience services such as music playback and navigation, but also for important functions for updating the vehicle software. If a connection is unauthorized or infected by malicious codes, it can be a serious security threat to the vehicle network. Therefore, security technology to protect the vehicle software and network is essential in V2D communication.

2.2. Android-Based Hacking Attacks. Malicious code is a widely used attack method at the application level that comes in various forms [8]. Various security threats such as leakage of private information, elevation of application privileges, and a denial-of-service (DoS) attack have been reported. The most common attack in the Android OS is the use of an application containing malicious code imported when a specific web page or email is loaded. Most malicious code is injected into the device without the user's awareness during the attack. When an application containing malicious code is executed on an Android OS, the code collects device and user information and sends it to a remote server. It also configures a backdoor by activating the service port to allow the attacker to reenter the device and elevate the privileges of available accounts. Subsequently, the malicious code can gain entire access to the infected device by rooting it. In particular, when an infected Android OS is connected to the inside of a self-driving vehicle, malicious code can be infiltrated directly into the vehicle to take control of the embedded OS or application software environment. For this reason, we need to detect malicious code from a self-driving vehicle.

2.3. Dataset. Recently, machine learning algorithms have been used to detect malicious code. This study proposes a machine learning-based intrusion detection module using the Android Adware and General Malware (AW&GM) dataset [9], which was developed by the Canadian Institute for Cybersecurity (CIC) in 2017. This publicly available dataset comprises Android sandboxes, Android adware, malware, and normal application traffic. It consists of traffic from 1,900 applications downloaded from Google Play (Android official application

Table 1: Three categories of the android adware and general malware dataset.

Class	Malware family	Number of apps
Adware	Airpush, dowgin, kemoge, mobidash, shuanet	250
General malware	AVpass, fakeAV, fakeflash/fakeplayer, GGtracker, penetho	150
Benign	Google play market (top free popular and top free new, 2015-2016)	1,500

market) and is used to classify normal and malicious code based on network traffic. This dataset is categorized with the following three classes (see Table 1).

2.4. Related Work for Protecting Vehicle Communication Networks. Kwon et al. [10] proposed a method for reconfiguring the electronic control units (ECUs) in a vehicle and deactivating attack packets to defend against network intrusion. In the proposed architecture, an IDS is introduced to detect cyberattacks in the network inside the vehicle, and a control module, called a mitigation manager, is applied to mitigate the damage from detected attacks. They then proposed an architecture to deliver commands to reconfigure ECUs, deactivate packets, reconfigure head units, delete packets in gateways at each domain, or switch domains into a secure mode. However, the framework and algorithms for the methodology were only proposed and not developed, and performance evaluations of the specific shape or architecture were insufficient. Therefore, a testbed and simulation environment should be prepared in order to verify the architecture appropriateness based on practical data such as detection accuracy, detection time, and resource utilization.

Han et al. [11] suggested an anomaly intrusion detection method for vehicular networks based on survival analysis. The method is based on an anomaly detection algorithm that detects a suspicious pattern within the usual pattern information. The method aims to detect three typical attack scenarios—flooding attacks, fuzzy attacks, and malfunction attacks—that attempt to manipulate and control using malicious packets. The authors noted that the proposed method can detect unknown attacks; however, they did not describe how to detect scenarios other than the three mentioned.

Zhang et al. [12] presented a cloud-assisted vehicle malware defense framework to defend vehicles against malware attacks. Such a service can help defend resource-constrained vehicle systems against malware by detecting new malware and updating onboard malware defense capabilities. Although the method is a cloud-based malware detection service, in-vehicle devices are also required to perform onboard threat defense functions. The premise of this service is that a single gateway should be able to control all external communication interfaces in the vehicle. If the vehicle cannot access the security cloud, it must find another way to inspect malware, however, no alternatives were explicitly suggested by the authors.

Table 2: User interaction scenarios analyzed in this study.

Category	User interaction scenario
Confidentiality	Information leakage (trip/location records, camera video/images, contact list, call/SMS history)
Integrity	Intentionally manipulated application installation (injection of malicious code disguised as a modified application)
Availability	Continuous resource consumption (large-scale traffic transmission) and system termination (intentionally resulting in various exception cases)

3. Machine Learning-Based Intrusion Detection Module

3.1. Malware Detection in Vehicle Networks. Study Group SG17 of the Telecommunication Standardization Sector, one of the International Telecommunication Unions that develops telecommunications standards, established the Intelligent Transport System (ITS) security investigation branch in order to standardize the ITS [13]. Specifically, X.itssec-4, which covers methodologies for IDSs for in-vehicle systems, defines the system structure and methods. Existing mechanisms for detecting unauthorized access into a CAN, injection of a malicious control message, and DoS attack include vehicle gateways and vehicle IDSs [14]. Attacks using adware and malware have various user interaction scenarios that can intrude into a vehicle through a smartphone (see Table 2).

Connected or self-driving vehicles are connected to external or public networks outside the vehicle via various interfaces. TCUs or CCUs are equipped with a modem and external communication interfaces to enable receipt of Global Positioning System signals and access to mobile networks. In-vehicle infotainment systems, which provide entertainment and information content, enable various applications by applying an embedded OS, such as QNX OS or Android OS. If security design is not considered in wired or wireless networks, these interfaces can be abused as a path for malware or malicious commands to enter the vehicle network (see Figure 1). In particular, the embedded OS environment can be controlled from the malware or malicious commands when these malicious processes bypass OS-level security logic or acquire root authority from self-privilege elevation. Therefore, in order to prevent malicious commands from gaining control of the embedded OS, this paper proposes a CAN gateway architecture that includes an intrusion detection module and detects malicious behaviors when Android OS-based devices are connected to the vehicle.

In this study, a machine learning-based intrusion detection module is installed in the vehicle IDS, which can detect intrusion into the CAN or any abnormalities, so that a head unit or ECU can be protected from malicious code. Such detection methods are implemented in the form of software-based computing modules to monitor malware injection or malicious code behaviors in the vehicle. The software can be installed as a component of the vehicle intrusion detection module or as an anti-virus agent in a head unit.

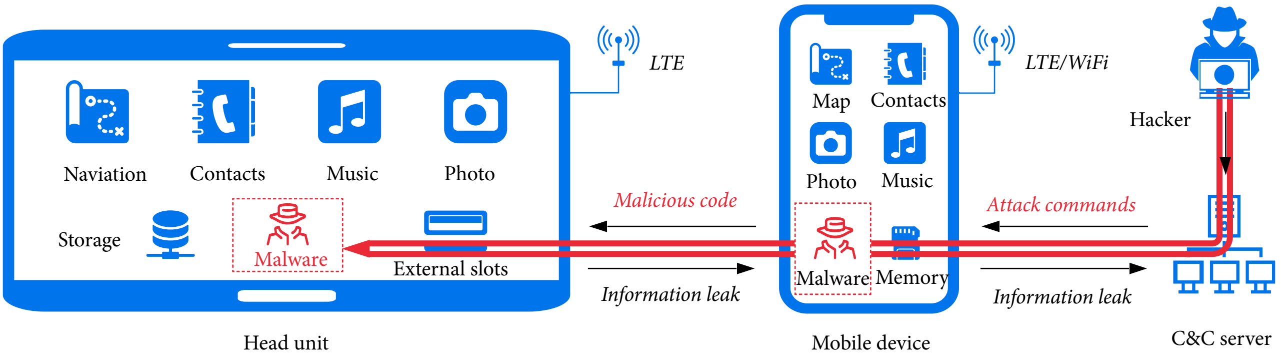


FIGURE 1: Schematic showing the head unit connected to an Android mobile device.

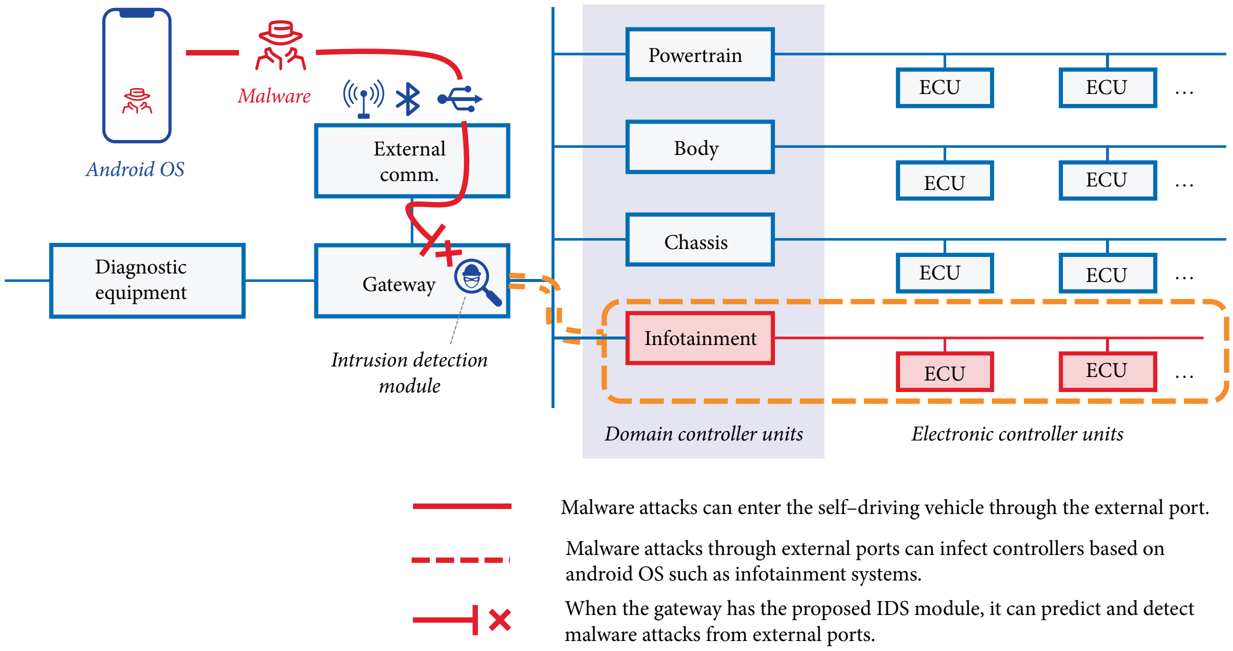

Malware
Android OS
External comm.
Diagnostic equipment
Gateway
Intrusion detection module
Powertrain
Body
Chassis
Infotainment
ECU
ECU
ECU
ECU
ECU
ECU
ECU
ECU
Domain controller units
Electronic controller units
Malware attacks can enter the self–driving vehicle through the external port.
Malware attacks through external ports can infect controllers based on android OS such as infotainment systems.
When the gateway has the proposed IDS module, it can predict and detect malware attacks from external ports.

FIGURE 2: CAN topology with the vehicle gateway and intrusion detection module.

The proposed detection software consists of input, analysis, evaluation, and notification modules. The traffic injected through the CAN is processed through the input module and entered into the analysis module, which is equipped with a machine learning algorithm (see Figure 2). The analysis module evaluates intrusion or abnormal behaviors based on a learned model and provides intrusion behavior information to a user or control center in real time. This machine learning-based intrusion detection module can improve the model's accuracy by repeatedly learning, verifying, and evaluating message patterns. Furthermore, detection rules for malicious behaviors can be updated to the vehicle gateway and each controller to accurately detect malicious code.

3.2. Data Preprocessing for Malicious Code Analysis. The characteristics of 79 features included in the CIC AW&GM dataset are analyzed using the Waikato Environment for Knowledge Analysis [15]. Feature selection is needed to reduce the dimensionality of the data. First, ten-fold cross-validation is applied by employing correlation-based feature selection (CFS) and an entropy-based information gain (IG) method. Constructing a validated dataset for an efficient experimental environment is important in machine learning. In this paper, we propose the improved feature selection (IFS) method, which combines the higher values derived from correlation and IG methods.

The proposed learning algorithm uses the selected network traffic features to detect malware. Unlike existing feature selection methods, IFS finds both greedy features and the highest correlation. There are two broad categories that can be used to measure the correlation between two random variables, one based on classical linear correlation (i.e., CFS) and the other based on information theory (i.e., the IG method). First, a pair of variables is defined for the CFS method and the linear correlation coefficient is derived [16]. In addition, the IG method decides how important a given attribute of the feature vectors is [17]. These two vectors are combined in order to determine the final features from the dataset that are highly

Table 3: Feature selection results.

Category	Selected Features[1] from IFS	F1 score
CFS (5)	*min_flowpktl*, *max_flowpktl*, *max_idle*, *bVarianceDataBytes*, *Init_Win_bytes_forward*	0.796
IG (5)	*avgPacketSize*, *max_fpktl*, *max_flowiat*, *fPktsPerSecond*, *Init_Win_bytes_forward*	0.806

[1] The feature of *min_flowpktl* means minimum length of a flow; *max_flowpktl* means maximum length of a flow; *max_idle* means maximum time a flow was idle before becoming active; *bVarianceDataBytes* means variance of total bytes used in backward direction; *avgPacketSize* means average size of packet; *max_fpktl* means maximum size of packet in forward direction; *max_flowiat* means maximum inter-arrival time of packet; *fPktsPerSecond* means number of forward packets per second; *Init_Win_bytes_forward* means the total number of bytes sent in initial window in the forward direction, respectively. Especially the last item is included in both CFS and IG results.

```
Input: $U$ is a universal set with all features.
Output: $\Omega^*$ is a subset with selected feature by IFS method.
Initialize $x_i, y_i, z_i \in U, (1 \le i \le N)$.
Get all $r(x_i, y_i)$ by linear correlation coefficient.
Sort $|r_{x_i,y_i}|$ values for $(1 \le i \le N)$.
Choose $j$ sets for top $x_i$ with high value of $|r|$, for relevant variable $j$ and $(1 \le j \le N)$.
Get combination $x_i, y_i \in C_j$, where $C_j \subset U$ and $n(C_j) = j$.
Determine $C_j^*$, where the maximum of F1 score with $x_i$.
Get all $H(i)$ by information gain.
Get $k$ is related to highly ranking variable.
Choose $k$ sets for top $z_i$ with high value of $H(i)$, relevant variable $k$ and $(1 \le k \le N)$.
Get elements $z_i \in C_k$, where $C_k \subset U$ and $n(C_k) = k$.
Determine $C_k^*$, where maximum of F1 score with $z_i$.
Merge $\Omega^* = \{C_j^*\} \cup \{C_k^*\}$.
```

Algorithm 1: Improved feature selection

correlated and have a strong impact between classes (see Algorithm 1).

In the CFS stage, we derive the linear correlation coefficient, $r(x_i, y_i)$. In this paper, we determine the verified features x_i with high value of $|r|$. The number of elements in the set C_j with elements x_i is j. These elements are selected with a relevant variable from CFS. The final C_j^* consists of a set with elements x_i calculating the highest F1 score (see Table 3 for CFS features). In the IG stage, we derive the IG ranking $H(i)$. The IG method finds the top 20% of z_i features, according to the χ^2 statistical distribution, from the 79 features. It finds that the statistic result is saturated at around k. The final C_k^* consists of a set with elements z_i calculating the highest F1 score (see Table 3 for IG features). The final feature selection is made by finding the union of the CFS and IG method feature sets. In this paper, each method selected five features; in total, nine features are used as input features (one feature was included in both feature sets). A total of 631,955 elements with these features were used for our model.

In-vehicle applications can be infected by Android malware via wireless or wired communication channels, as illustrated in Figure 1. Several studies suggesting IDS architectures have reported that malware can be detected in the network traffic of devices [18, 19]. This paper selected nine features using the IFS method and shows that malware can be detected from the network traffic using a machine learning-based IDS module.

Table 4: Feature scaling results.

Category	F1 score
MinMaxScaler	0.813
StandardScaler	0.810

Table 5: Types of applications and their ratio.

Category	Count	Ratio (%)
Benign	471,597	74.6
Adware	155,613	24.6
General malware	4,745	0.8

As the original data has unique characteristics and distributions, learning from these data may be slow or result in modeling errors. In the case of network traffic, it is essential to perform scaling because each feature has a uniquely defined data range and unit. Scaling is a data preprocessing task that helps prevent underflow and overflow when learning from experimental data. It is performed based on the nine selected features. The F1 score results after applying the MinMaxScaler and StandardScaler are described in Table 4. The MinMaxScaler scales all features to be exactly between zero and one. The StandardScaler, in contrast, does not limit the minimum and maximum values, but ensures that all features have an average of zero and a variance of one. Thus, all features have the same size. A comparison of their F1 score results of the two scaling methods indicates that MinMaxScaler is more advantageous due to the nature of the network traffic, which comprises a wide range of data. Therefore, in this study, the MinMaxScaler technique is applied to each algorithm.

Next, we analyzed algorithms that detect adware and malware typical in Android OS. In this study, these attack detection techniques are compared by applying six machine learning algorithms to the dataset. Furthermore, we analyzed the results of using a general machine learning algorithm, assuming that the computing power employed in the vehicle-embedded software can analyze traffic data using a general specification rather than a high-performance system. The dataset used in this study consists of three classes: benign, adware, and general malware. There is a strong imbalance between these classes (see Table 5).

When the data modeling results are evaluated with general accuracy, the evaluation result may suggest that its

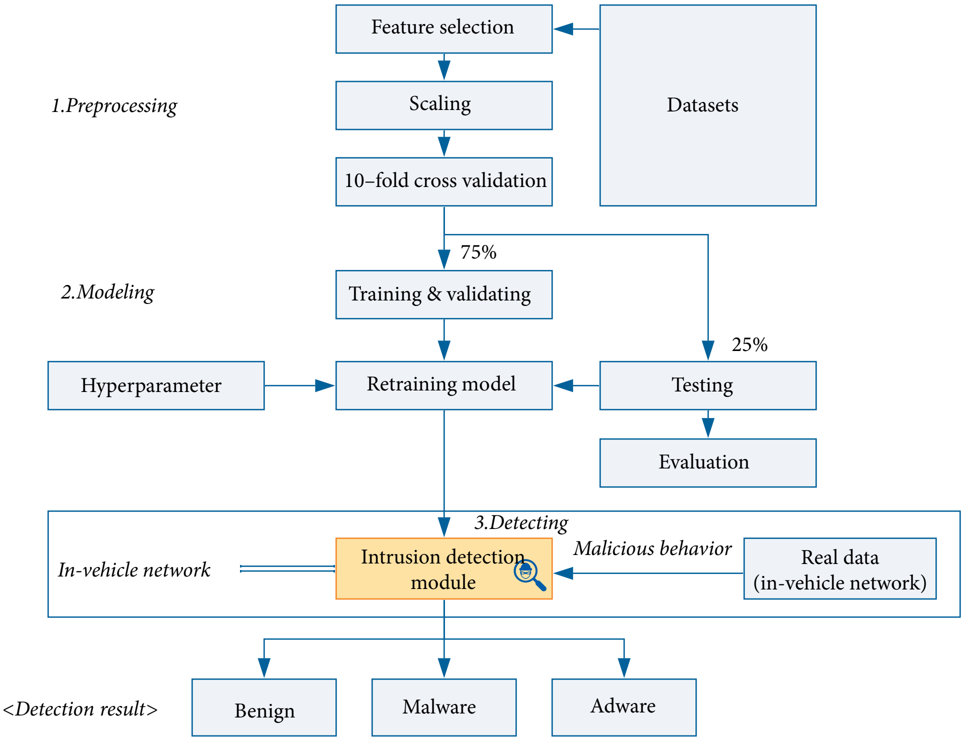


Figure 3: Intrusion detection module in a vehicle network.

performance is good even when it is not. For example, the overall accuracy can be high if the benign category, which has high importance in the dataset, is accurately predicted, even if general malware, which has low importance, is not accurately predicted. Therefore, the F1 score, which uses the harmonic mean based on recall and precision, is used to evaluate prediction accuracy.

In summary, the proposed machine learning-based intrusion detection module detects Android malware for a self-driving vehicle and labels its type (i.e., adware or general malware). The procedure, which is based on the detection of the network traffic deviation on Android OS, is divided into three phases, as shown in Figure 3. The first phase focuses on data preprocessing. Feature selection is performed to select the most relevant features from all measuring features in the dataset. The second phase consists of modeling. Using ten-fold cross-validation, this phase trains the machine learning model using 75% of the dataset and suggests the most suitable hyperparameters for the retraining model. In addition, this phase uses 25% of the dataset for testing and evaluating the proposed intrusion detection module. Therefore, a machine learning model tuned by hyperparameters is created using the training dataset, and a testing dataset is applied to evaluate the model. In the third phase, the intrusion detection module can detect malicious behaviors in real time when real data flows into the self-driving vehicle. Specifically, the proposed intrusion detection module should be included in the vehicle gateway shown in Figure 2.

4. Simulation Results

Six machine learning algorithms—the random forest (RF), decision tree (DT), *k*-nearest neighbors classifier (KC), gradient boosting classifier (GB), extra tree classifier (ET), and bagging classifier (BC) algorithms—are used to analyze the data and their results are compared to those of the proposed algorithm. In addition, we also present hyperparameters for each algorithm for comparative verification of the machine learning used to implement the malware detection module in a self-driving vehicle gateway. It is important to tune hyperparameters for result, performance, and cost optimization when analyzing data using a machine learning algorithm. Indeed, significant differences in the performance and accuracy of analysis results can occur depending on the configuration of the hyperparameters. We present the hyperparameters used in each experiment with the F1 score and elapsed time for each algorithm. These hyperparameters were derived by changing various experimental conditions repeatedly for each algorithm.

The nine input features are defined through the feature selection process and the output is defined using two classification scenarios to analyze the experimental results. In the first scenario (see Figure 4(a)), benign code, adware, and general malware are accurately detected, whereas the second scenario (see Figure 4(b)) is a binary classification scenario where only benign code and adware are detected because general malware accounts for only 0.8% of the dataset. It is meaningful to compare the results of the binary classification because its impact can be predicted through the first scenario.

In this paper, malware detection using machine learning is included to develop the IDS module included in self-driving vehicles. The F1 score used in machine learning calculates the accuracy, recall, and precision values for all cases to evaluate the model's performance. This general method, which took about 3.570 s to verify the dataset on average, is not suitable for real-time detection. We applied a faster F1 score evaluation

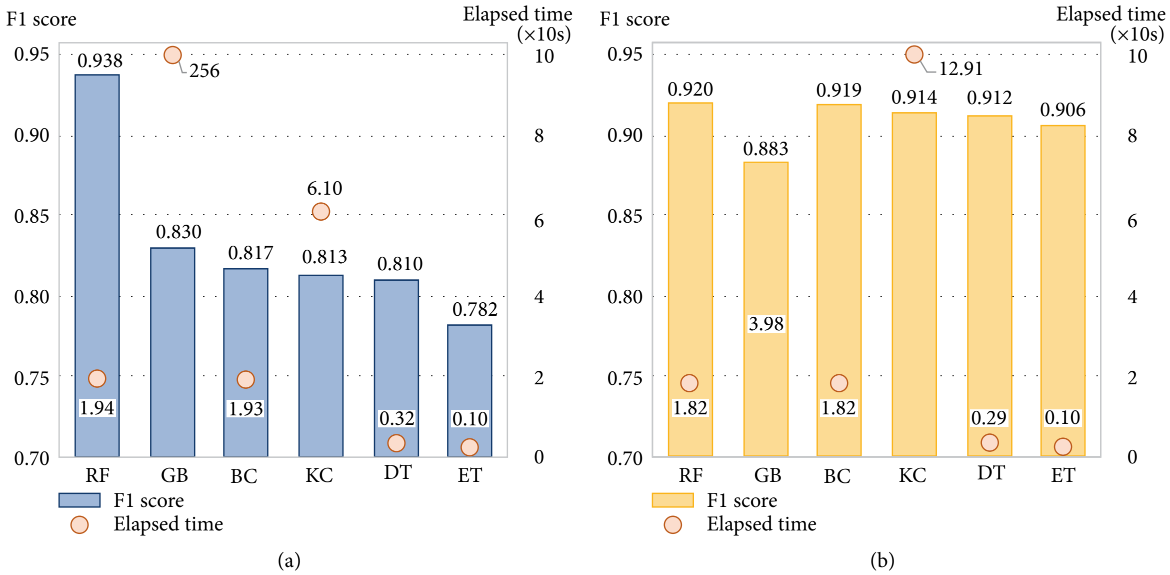


Figure 4: Simulation results. (a) Multiclass classification. (b) Binary classification.

Table 6: Comparison of the proposed score–function model with a basic scikit-learn method.

Evaluation factors	Basic model	Score-function model
Elapsed time (second)	3.570	0.049
Accuracy	0.929	0.929
Precision	0.849	0.849
Recall	0.761	0.761
F1 score	0.796	0.803

method because malware should be detected in real time on autonomous vehicles. In order to generate a class that calculates and returns a confusion matrix quickly, we proposed a new score function. Through this function, we obtained the F1 score directly when the model was training. The function stored the value of the computed confusion matrix and was reusable when the F1 score was called for performance evaluation. In this case, the elapsed average time was 0.049 s, which is acceptable for real-time detection. Abnormal behavior prediction can therefore determine within 0.049 s when new traffic occurred in the self-driving vehicle (see Table 6).

For the RF algorithm, the highest F1 score is obtained when the random state is 42, the number of estimators is 85, the maximum depth is 24, and the maximum number of features is 4. Although the RF's prediction accuracy is typically higher when using binary classification, in this case, it is higher under scenario 1. Overall, the RF algorithm had the highest prediction accuracy of the machine learning algorithms tested. For the DT algorithm, the highest F1 score was observed when the random state is 42 and the minimum leaf sample is 2. For KC, the highest F1 score was observed under the following conditions: uniform weights and 7 estimators. For both DT and KC, accuracy may decrease in datasets with large data imbalances. Moreover, although the KC algorithm exhibits higher prediction accuracy in binary classification scenarios, its learning time is more than twice that in scenario 1. For GB, the highest F1 score is observed when the random state is 42, the number of estimators is 50, the maximum depth is 15, and the maximum number of features is 5. Its prediction accuracy is generally high, but its learning time is the longest of all algorithms, at 2,556,517 ms; for comparison, the learning time of the second longest algorithm, KC, is 60,967 ms and that of the shortest, ET, is merely 976 ms. Therefore, although GB is suitable for binary classification, the learning time costs are too large for general classification. For ET, the highest F1 score was observed when the random state is 42, the splitter is random, and the number of estimators is 100. This algorithm shows the shortest learning time under both scenarios. Although the binary classification has high prediction accuracy and the shortest learning time (95 ms), making it very efficient, its prediction accuracy is significantly reduced in scenario 1. For BC, the highest F1 score is observed when the random state is 42 and the number of estimators is 10. Similar to ET, BC is efficient because of its high prediction accuracy in the binary classification scenario, but it has significantly reduced prediction accuracy under scenario 1.

In summary, the algorithm's overall prediction accuracy was 90% or greater with binary classification for all algorithms except GB. Therefore, in this case, an algorithm with a short learning time can be selected. In order to detect malware or adware in an embedded software environment such as a vehicle, high accuracy and a fast response time are very important. Therefore, the ET algorithm, with its learning time of 95 ms and prediction accuracy of 90.6% in binary classification scenarios, would be suitable. However, considering that the attack detection method in the Android OS is classification scenario 1, the RF algorithm, which has the highest prediction accuracy and a learning time of 19,401 ms, would be the most suitable.

We use the receiver operating characteristic (ROC) curve to evaluate the experimental results of each algorithm. The ROC curve, a widely used tool for binary classification, plots the method's sensitivity against its specificity. The area under

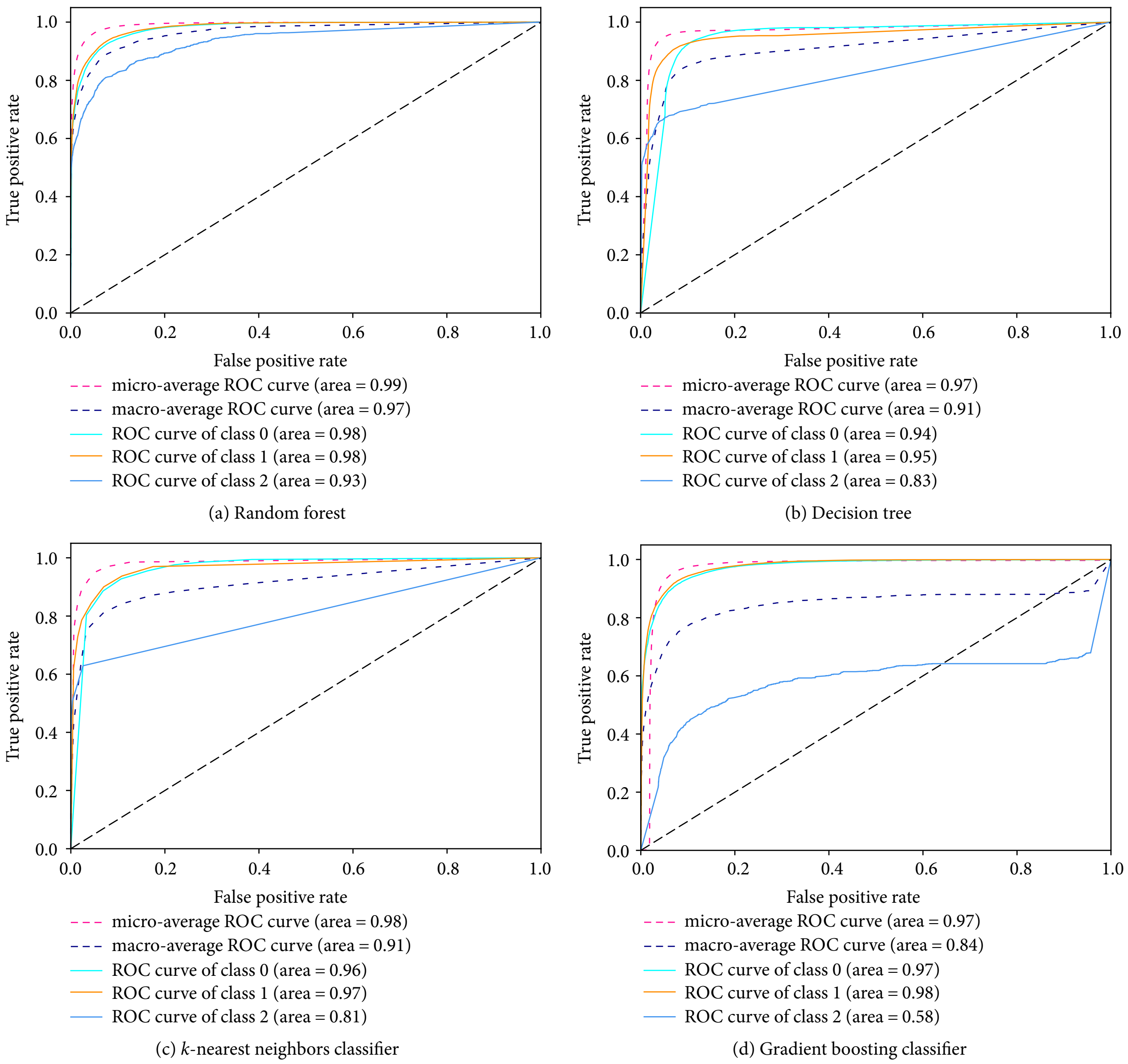


Figure 5: Simulation results with ROC curve and AUC, where "Area" denotes the AUC and classes 0, 1, and 2 correspond to benign, adware, and general malware, respectively. (a) RF. (b) DT. (c) KC. (d) GB.

the ROC curve (AUC), which represents the surface integral under the curve, is an indicator of the detection performance of each classifier. When the curve approaches the graph of $y = x$, the classifier is purely random and the AUC is near 0.5; likewise, detection performance is better when the curve at the top left area is far from the random classifier line. The AUC of the perfect classifier is 1.

We compared the performance of four algorithms: RF, DT, KC, and GB (see Figure 5). In the RF algorithm shown in Figure 5(a), the AUC of the macro-average obtained by calculating the measurement of each class is 0.97 and the micro-average for integrated classes is 0.99. For the imbalanced (class 2) malware, the AUC was 0.93, which is relatively good compared to other classifiers. In Figures 5(b) and 5(c), the DC and KC show similar detection results. However, when class 2 malware is detected, the DC is slightly better than the KC. Moreover, in Figure 5(d), the micro-average and macro-average detection results for GB are 0.97 and 0.84, respectively. However, the detection result of class 2 is low (0.58) due to class 2's scarcity in the dataset. That is, GB can perform well for binary classification, but is not suitable for multiclass classification. In conclusion, the ROC curve and AUC of each classifier show that the RF algorithm better detects malware than the other algorithms.

5. Conclusion

The increasing connectivity of vehicles has also increased security threats. Malicious code can flow into a vehicle's internal network when a device infected with malicious code is connected to the vehicle through an external communication channel. High accuracy and speed are key for detecting malicious behaviors in the embedded environment of vehicles, where responses must be processed in real time. This study, therefore, analyzed security threats from adware and malware in the Android OS within a self-driving vehicle. Network traffic was analyzed to detect malicious behaviors at the network in the module. In addition, a machine learning-based intrusion detection module for malware detection was proposed. Finally, we proposed a machine learning algorithm that can detect Android malware for vehicles with high accuracy and in a short time. We compared the algorithm's detection accuracy and speed with proposed optimal hyperparameters to six machine learning algorithms. In addition, we also found that we can significantly reduce the elapsed time by using the novel score-function model for real-time detection. Our simulation we demonstrated that our algorithm is highly accurate (92.9%) and fast (0.049 s), making it suitable for real-time malware detection in a self-driving vehicle environment.

Data Availability

The CIC AW&GM dataset can be found in the official webpage of the institute https://www.unb.ca/cic/datasets/android-adware.html.

Conflicts of Interest

Seunghyun Park and Jin-Young Choi declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) [No. 2018-0-00532, Development of High-Assurance (≥EAL6) Secure Microkernel].

References

[1] J. Takahashi, "An overview of cyber security for connected vehicles," *IEICE Transactions on Information and Systems*, vol. E101-D, no. 11, pp. 2561–2575, 2018.

[2] J. Cui, L. S. Liew, G. Sabaliauskaite, and F. Zhou, "A review on safety failures, security attacks, and available countermeasures for autonomous vehicles," *Ad Hoc Networks*, vol. 90, pp. 1–13, 2019.

[3] C. Miller and C. Valasek, "Remote exploitation of an unaltered passenger vehicle," *Black Hat USA*, p. 91, 2015.

[4] S. Nie, L. Liu, and Y. Du, "Free-fall: hacking tesla from wireless to CAN bus," *Black Hat USA*, pp. 1–16, 2017.

[5] C. Riggs, C.-E. Rigaud, R. Beard, T. Douglas, and K. Elish, "A survey on connected vehicles vulnerabilities and countermeasures," *Journal of Traffic and Logistics Engineering*, vol. 6, no. 1, pp. 11–16, 2018.

[6] T. Hoppe, S. Kiltz, and J. Dittmann, "Applying intrusion detection to automotive it—early insights and remaining challenges," *Journal of Information Assurance and Security*, vol. 4, no. 3, pp. 226–235, 2009.

[7] K. Bian, G. Zhang, and L. Song, "Toward secure crowd sensing in vehicle-to-everything networks," *IEEE Network*, vol. 32, no. 2, pp. 126–131, 2018.

[8] N. Milosevic, A. Dehghantanha, and K.-K. R. Choo, "Machine learning aided android malware classification," *Computers & Electrical Engineering*, vol. 61, pp. 266–274, 2017.

[9] A. H. Lashkari, A. F. A. Kadir, H. Gonzalez, K. F. Mbah, and A. A. Ghorbani, "Towards a network-based framework for android malware detection and characterization," in *Proceedings of 2017 15th Annual Conference on Privacy, Security and Trust*, pp. 233–234, IEEE, Canada, 2017.

[10] H. Kwon, S. Lee, J. Choi, and B. Chung, "Mitigation mechanism against in-vehicle network intrusion by reconfiguring ECU and disabling attack packet," in *Proceedings of 2018 International Conference on Information Technology (InCIT)*, pp. 55–59, IEEE, Thailand, 2018.

[11] M. L. Han, B. I. Kwak, and H. K. Kim, "Anomaly intrusion detection method for vehicular networks based on survival analysis," *Vehicular Communications*, vol. 14, pp. 52–63, 2018.

[12] T. Zhang, H. Antunes, and S. Aggarwal, "Defending connected vehicles against malware: challenges and a solution framework," *IEEE Internet Things Journal*, vol. 1, no. 1, pp. 10–21, 2014.

[13] E. Yagdereli, C. Gemci, and A. Z. Aktas, "A study on cybersecurity on autonomous and unmanned vehicles," *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, vol. 12, no. 4, pp. 369–381, 2015.

[14] B. Groza and P.-S. Murvay, "Security solution for the controller area network: bringing authentication to in-vehicle networks," *IEEE Vehicular Technology Magazine*, vol. 13, no. 1, pp. 40–47, 2018.

[15] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[16] M. Doshi and S. Chaturvedi, "Correlation based feature selection (CFS) technique to predict student performance," *International Journal of Computer Networks & Communications (IJCNC)*, vol. 6, no. 3, pp. 197–206, 2014.

[17] S. Lei, "A feature selection method based on information gain and genetic algorithm," in *Proceedings of 2012 International Conference on Computer Science and Electronics Engineering (ICCSEE)*, pp. 355–358, IEEE, China, 2012.

[18] A. Arora, S. Garg, and S. K. Peddoju, "Malware detection using network traffic analysis in android based mobile devices," in *Proceedings of 2014 Eighth International Conference on Next Generation Mobile Applications, Services and Technologies (NGMAST 2014)*, pp. 66–71, IEEE, United Kingdom, 2014.

[19] S. Iqbal, A. Haque, and M. Zulkernine, "Towards a security architecture for protecting connected vehicles from malware," in *Proceedings of 2019 IEEE 89th Vehicular Technology Conference (VTC2019-Srping)*, pp. 1–5, IEEE, Malaysia, 2019.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8167295, 11 pages
https://doi.org/10.1155/2020/8167295

Research Article

Data Analysis for Emotion Classification Based on Bio-Information in Self-Driving Vehicles

Tae-Yeun Kim,[1] Hoon Ko ,[2] and Sung-Hwan Kim [1]

[1]*SW Convergence Education Institute, Chosun University, Gwangju, Republic of Korea*
[2]*IT Research Institute, Chosun University, Gwangju, Republic of Korea*

Correspondence should be addressed to Sung-Hwan Kim; shkimtop@chosun.ac.kr

Received 13 May 2019; Accepted 28 October 2019; Published 16 January 2020

Guest Editor: Hyunhee Park

All persons in self-driving vehicle would like to receive each service. To do it, the system has to know the person's state from emotion or stress, and to know the person's state, it has to catch by analyzing the person's bio-information. In this paper, we propose a system for inferring emotion using EEG, pulse, blood pressure (systolic and diastolic blood pressure) of user, and recommending color and music according to emotional state of user for a user service in self-driving vehicle. The proposed system is designed to classify the four emotional information (stability, relaxation, tension, and excitement) by using EEG data to infer and classify emotional state according to user's stress. SVM algorithm was used to classify bio information according to stress index using brain wave data of the fuzzy control system, pulse, and blood pressure data. When 80% of data were learned according to the ratio of training data by using the SVM algorithm to classify the EEG, blood pressure, and pulse rate databased on the biometric emotion information, the highest performance of 86.1% was shown. The bio-information classification system based on the stress index proposed in this paper will help to study the interaction between human and computer (HCI) in the 4th Industrial Revolution by classifying emotional color and emotional sound according to the emotion of the user it is expected.

1. Introduction

In recent years, a new future technology called emotional artificial intelligence (AI) has emerged owing to the advancement of the fourth industrial revolution era. Particularly, AI-based emotional computing technology, which can interpret and analyze human emotions, is advancing rapidly owing to a convergence of information and communication technology (ICT) and cognitive science areas [1, 2]. Accordingly, the human–computer interface (HCI) technology is becoming increasingly important, and along with the progress made in the studies on HCI, studies are being increasingly conducted on computer reactions based on emotion inference or user intention rather than computer reactions induced by direct inputs of the user [3]. Particularly, the brain–computer interface (BCI) technology analyzes, commands, or controls electroencephalography (EEG) signals in frequency domain, which were measured at a human scalp. Various studies are conducted to provide interaction between humans and computers by converging the user's emotion information and environment information based on the BCI technology [4, 5]. Emotion is a cognitive ability of humans and a reaction for external sensory stimulation. Humans react emotionally according to various social and cultural factors and feel various emotions accordingly. Emotion recognition technology is a means of making intelligent decisions that can enable appropriate behavior by extracting information such as facial expressions or body gestures of a user based on emotion data [6]. As such, it is extremely important for computers to have emotion recognition capability to process human emotions through learning and adaptation in order to process the interaction between humans and computers more efficiently. In modern society, mental stresses such as various work-related stresses, conflicts in interpersonal relationships, and financial problems have emerged as social problems, and efforts are actively being made to improve the quality of physical and psychological life [7]. As such, it is important to measure the emotional stress level of users by using EEG and biosignals in objective numerical values in terms of psychological state, and efforts to relieve such stresses by recognizing the subsequent physical changes are necessary. As colors and music are formed in a short time and last for a long time in memory, they can play vital roles in understanding and analyzing human emotions. Therefore, this study aims to design a system that infers emotion by using biometrics of user, such as EEG, pulse rate, and blood pressure (systolic and diastolic blood pressure), and that recommends colors and music based on the emotional state of the user, i.e., the stress index. The system proposed in this paper recognizes the emotion of the user by learning and patterning the reactions appearing according to the user's emotional state and classifies the biometric emotional information according to the stress index. A fuzzy system was designed by using EEG data to classify the biometrics into biometric emotion information according to the stress index. Furthermore, it was designed to produce the pulse rate and blood pressure (systolic and diastolic blood pressure) data in single packets separately and to send them to the database. As such, after acquiring the biometrics, the biometric emotion information is classified according to the stress index through a support vector machine (SVM) algorithm. In general, various learning algorithms are applied to decision-support systems. However, as the biometric data used in this study consist of EEG, pulse rate, and blood pressure (systolic blood pressure and diastolic blood pressure) data, which have nonlinear data structures, the SVM algorithm was used to solve the nonlinear discrimination problem in a multilayer perceptron structure. Depending on the classified stress emotion, the data corresponding to the color and music values are classified. This paper is organized as follows. Section 2 discusses related research, and Section 3 describes the system configuration and design. Subsequently, Section 4 presents the performance evaluation and experimental results. Finally, Section 5 discusses the study's conclusions and directions for future research.

2. Related Research

2.1. Cognitive Science. Cognitive science examines the cognitive processes of humans and animals, such as perception, language, learning, and emotions. In this field, techniques are studied for applying these cognitive processes to the development of robots, electronic products, buildings, and so on [8, 9]. Recently, as cognitive science has begun to attract attention, studies are being conducted on the detection, processing, and analyzing of various types of signals that are generated by bodily activities. In particular, studies on the structure and functions of the brain are being actively conducted. These studies utilize various biometrics for not only medical diagnoses, but also the examination of people's cognitive and emotional states through analysis via certain algorithms.

2.2. Emotional Engineering. Emotions are high-level psychological experiences that occur in people via their senses and cognition in response to an external physical stimuli. These are often-changing psychological effects that occur in people owing to complex feelings, such as comfort, pleasure, unease, and discomfort. These are the moods and feelings that are connected to expressive behavior during emotional reactions and physiological changes. Further, they are a dynamic aspect of thought. Emotional engineering is a field that studies emotions and deals with them in practical terms. This combines areas related to human psychological reactions and their applications, such as human factor engineering, cognitive science, behavioral science, pragmatic aesthetics, and environmental psychology [10]. Changes in people's psychological states are expressed externally in the form of feelings or emotions. In the past, people's psychological states and emotions were mostly analyzed and used as clinical data for medical diagnoses and treatments. However, they are currently used in a variety of fields, such as buildings that incorporate psychotherapeutic art and automatic lighting that reflects emotions, cars that prevent sleepiness, and robots that understand and respond to human feelings. Researchers are studying emotionally intelligent computer technology that can understand psychological states, feelings, and emotions. Emotionally intelligent computing allows computers to recognize people's feelings and emotions and to perform actions that are suitable to the circumstances. This entails autonomous systems that are able to perform suitable actions based on previous knowledge or the current psychological state. Recently, as wearable computing technology has evolved, it is able to measure biosignals such as electroencephalography (EEG), electrography (EMG), electrocardiogram (ECG), and galvanic skin response (GSR) more accurately, and produces improved recognition results regarding subjects' mental states, emotions, and physiological states [11].

2.3. Biobased Emotional Cognitive Technology Trend. In general, the emotions felt by people can be distinguished by the central nervous system or autonomic nervous system reactions, i.e., information such as ECG, EEG, skin temperature (SKT), and GSR. The biosignal emotion recognition technology, that has been studied so far, can be divided into studies using statistical methods and those using machine learning methods. Studies on rule-based techniques analyze many biosignals, and extract emotional features that express emotions. Then, threshold values that correspond to rules are set for each emotion, and emotions are classified according to these thresholds. Emotion recognition methods that use machine learning are based on techniques such as neural network (NN), support vector machine (SVM), k-nearest neighbor (kNN), multi-layer perceptron (MLP), Gaussian mixture model (GMM), decision tree (DT), and Baysian network (BN) [12–14]. For biosignal learning, it is necessary to collect a large amount of feature data and perform the training required for emotion recognition. Therefore, the amount of collected feature data and its reliability have a significant effect on the subsequent performance of the recognition system. However, most emotion recognition systems that have been studied so far use data with staged and exaggerated emotions. Because it is easy to artificially acquire learning data that includes emotions, this is a major factor that degrades the performance of emotion recognition systems in real situations. Therefore, the collection of reliable biometric data is considered necessary for biosignal-based emotion recognition. However, in biosignal collection, a user's emotional states can easily change according to the environmental conditions or the user's psychological state, and it is very difficult to perfectly recognize a user's emotions based on a single biosignal. Therefore, rather than emotion recognition using only one biosignal, researchers are investigating methods that use several biosignals in conjunction or use additional emotional measurement indicators, such as voice and facial expressions. Emotion recognition technology using brainwaves is one of the most actively studied fields in biotechnology. Unlike other biosignals, brainwaves show the state of the central nervous system. The alpha and beta waves that occur according to the state of brain activity are known to be related to emotion. Alpha waves generally indicate a stable or relaxed state, and these increase during positive emotions. On the other hand, beta waves increase during negative emotional states [14]. In addition to brainwaves, autonomic nervous system reactions occur when commands are received from the brain, and these indicate bodily changes when a person is surprised or encounters danger. These reactions include the heart beating rapidly, the facial muscles stiffening, the palms and back of the neck sweating, and changes in body temperature or the temperature at certain parts of the body. These bodily changes are controlled by the autonomic nervous system, and therefore emotional states can be inferred from biosignals by detecting autonomic nervous system reactions. Biosignals that can be measured following autonomic nervous system reactions include ECG, EMG, GSR, and SKT. Studies have used the heart rate variability (HRV) and heart rate (HR) to recognize emotions. A low heart rate indicates a relaxed state, and a high heart rate can indicate stress, frustration, and dissatisfaction. As such, the heart rate is often used for data analysis. Emotions can be classified by calculating the HRV from the heartbeat and extracting features. However, it is difficult to accurately recognize emotions from the heartbeat alone [15]. Emotion recognition via EMG has mainly consisted of studies on the recognition of emotions through the movement of facial muscles. The movement of facial muscles can be measured through an EMG of the face, and this can be used to measure high-stress tension and so on, and recognize emotions. In a tense/aroused state, the EMG increases, whereas in a relaxed state it decreases. However, the absolute levels of muscle tension vary according to the part of the muscle that is measured. As such, this signal requires precise measurement. GSR essentially refers to skin conductivity, which increases when sweat is present on the skin. This can be used as an indicator of stimulation or stress. The GSR amplitude increases during arousal or negative emotions. The reaction speed (latency) is quick during sensitive stimulation. The reaction sensitivity (slope) is large during sudden or sensitive stimulation. Therefore, GSR can be used as a good indicator for measuring negative emotions. SKT is different from body temperature, in that it is a temperature index for certain body parts. This is not a normal core biosignal indicator, but it can be used as a slow indicator of changes in emotional states. It is significantly affected by external environmental factors. In general, when the amplitude of the SKT signal is large, this indicates relaxed and pleasant positive emotions. When it is small, this indicates tense or uncomfortable negative emotions. Recently, there has been a sudden interest in many different fields in attempts to monitor image, voice, biometric, brainwave, and body data and extract emotions to provide emotional application services. Emotional application service technology is evolving, and exhibits significant potential for use in a variety of fields, such as entertainment, healthcare, market analysis, online education, automobiles, customer marketing, and general home use.

2.4. Stress and Bio-Information. Previous studies have used biosignals as a method of identifying stress states in individuals. Bakker et al. [16], Healey and Picard [17], and Jung and Yoon [18] have collected biosignals from workers, drivers, and senior citizens, respectively, to evaluate stress levels. Setz et al. [19], Melillo et al. [20], and Kurniawan et al. [21] administered tests that required learning capabilities to create experimental environments, and then identified stress states. However, these previous studies have been limited by the fact that they considered the identification of a posteriori states, and did not consider combinations of features that are appropriate for stress relief using emotions. In this chapter, previous studies on the identification of stress states using biosignals are listed by their purpose, the biosignals that they used, and their analysis techniques, as shown in Table 1. Previous studies have identified stress states using various combinations of biosignals such as HR, GSR, ECG, EMG, and brainwaves. Of these, the heartbeat data has been used most often to identify stress. However, because heartbeat variability can occur in a variety of contexts other than stress, this is used in combination with other data [22]. Sun et al. [23] used the heartbeat data along with GSR and accelerometer data to identify stress during physical activities. However, because GSR is more sensitive to movement than other biosignals, it is difficult to obtain accurate data in situations with a lot of movement. Kurniawan et al. performed experiments using voice data to analyze the words uttered by subjects and identify stress [21]. In studies by Setz et al. and Melillo et al., subjects were given tests that required learning capabilities to create experimental environments. In this manner, previous studies have faced difficulties in adequately measuring and analyzing user stress without using a variety of biometrics. In addition, the goals of studies on stress that have used biometrics have only been focused on measurement and analysis. As such, there has been a lack of studies on stress and emotions that are measured via biometrics. This study considers the limitations of previous work, and employs user biometrics such as brainwaves, pulse, and blood pressure (systolic and diastolic blood pressure). In addition, it proposes a system that uses biosignals to identify a user's stress states and infer emotions, to suggest colors and sounds according to the user's emotional state, i.e., their stress index.

3. System Configuration and Design

To infer and classify the emotional state of a user according to the user's stress, this study designed a fuzzy system by using the EEG data after acquiring the EEG, pulse rate, and blood pressure from the sensors, and the pulse rate and blood pressure (systolic and diastolic blood pressure) data were produced in single packets separately and sent to the database. The SVM

Table 1: Summary of related studies.

Objects (topics)	Signals used	Analysis methodologies	References
Automatic identification of stress causes of employees	GSR	Adaptive windowing	Bakker et al. [16]
Detecting real-world driving stress	HR, EMG, Respiration	Continuous correlations	Healey and Picard [17]
Multi-level assessment model for monitoring elder's health condition	HR, EEG, ECG	SVM, DT, Expectation maximization	Jung and Yoon [18]
Personal health system for detecting stress	GSR	Latent dirichlet allocation, SVM	Setz et al. [19]
Stress elicitation by examination	HR Voice, GSR	Latent dirichlet allocation DT, SVM, K-Means	Melillo et al. [20] Kurniawan [21]
Activity-aware mental stress detection (sitting, standing and walking)	HR, GSR, Accelemeter	DT, SVM, Bayes network	Sun et al. [23]
Automatic cry detection in early childhood	Voice	Gentle-boost	Ruvolo and Movellan [24]
Automatic classification of infant crying for early disease detection	Voice	Genetic selection of a fuzzy model	Rosales-Pérez et al. [25]
Automatic detection of the expiratory and inspiratory phases in newborn cry signal	Voice	Hidden markov model	Abou-Abbas et al. [26]

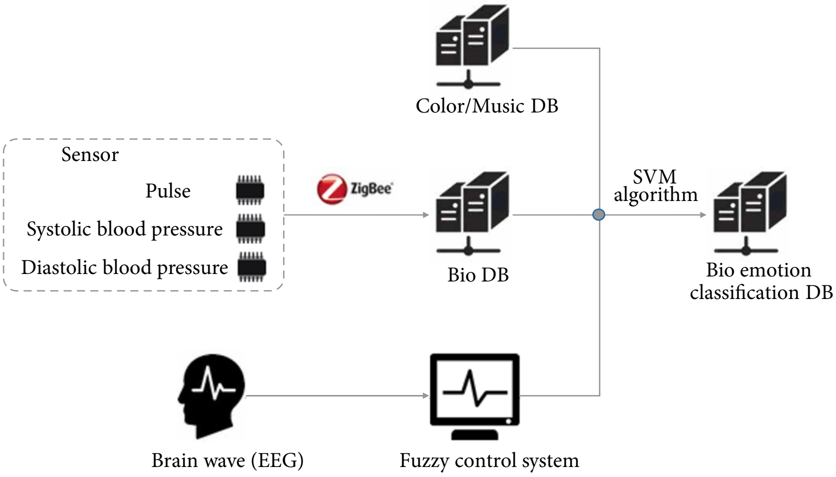


Figure 1: The system configuration diagram.

algorithm was used to classify the biometric emotion information according to the stress index by using the acquired EEG, pulse rate, and blood pressure data of the fuzzy control system. The biometric emotion information is classified into color values corresponding to the emotion based on 20 color emotion models selected through HP's "the Meaning of Color," and as for the music, the music pieces provided by "Samsung Idea" for music therapy were collected, classified, and used [27]. Figure 1 shows the configuration diagram of the system proposed in this paper, which performs emotional classification and recommendation based on biometrics and stress index.

This study designed a fuzzy system to classify the biometric emotion information according to a user's stress index by using the EEG data. The stress emotion information expressed by the fuzzy control system can be divided into four types: stable, relaxed, tensed, and excited. BIOPAC MP 150 was used to measure the EEG of the user and the measurement was conducted according to the 10–20 International System of Electrode Placement [28, 29]. When conducting the measurement, sampling was performed at 256 Hz for only the EEG data that passed through a notch filter of 60 Hz in terms of hardware. The sampled data were filtered in the frequency band 0.5–50 Hz. To analyze the EEG data, the data for 60 s, which is considered the stable-state analysis section for the analysis of EEG, were used excluding each 30 s part of starting and ending. First, the EEG data, which are the data in time domain, were converted into frequency domain by using fast Fourier transform, as shown in Equation (1) in order to analyze them.

$$H(f_n) = \sum_{k=0}^{N-1} h_k e^{-j2\pi kn/N}. \tag{1}$$

To extract an absolute size by each frequency, the data were converted into the frequency domain through the Fourier

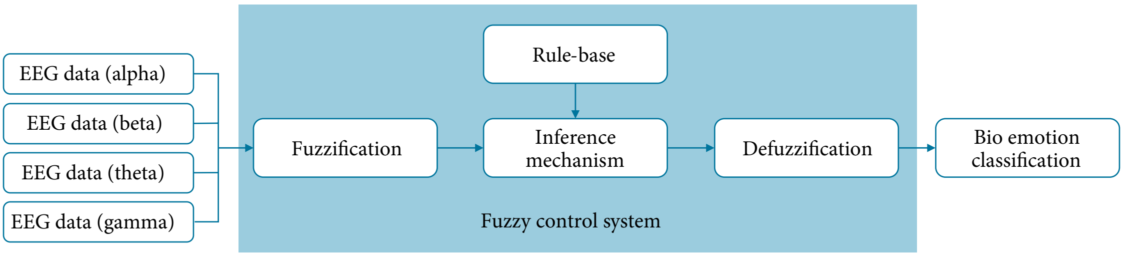


Figure 2: The fuzzy control system using EEG data based on bio emotion classification.

transform process. Moreover, the power spectrum analysis method was used for comparison and analysis [30]. The power spectrum analysis method is a widely used analysis method in many areas including biosignals. Depending on the expression method, it is divided into one-side and two-side. One-side shows 0 and positive frequency domain only, and two-side shows the frequency domain of {negative, 0, positive}. This study obtained the absolute values of power spectrum for the band frequencies of α, β, θ, and γ by using the one-side power spectrum analysis method. The Fourier inverse transform is shown in Eq. (2)

$$h_k = \frac{1}{N}\sum_{k=0}^{N-1} H_n e^{-j2\pi kn/N}. \tag{2}$$

After obtaining the absolute values on both sides of Equation (2), if they are squared and added, then Equation (3) is induced. Here, it can be confirmed that the sum of square of the signal that was subjected to the Fourier transform is the same as the sum of square of the original signal. Here, the sum of Fourier transform or the sum of square of the original signal is called the total power value. In other words, it indicates that the total power value is identical in both the frequency space and the temporal space. This is called Parseval theorem [31].

$$\text{Total Power} = \sum_{k=0}^{N-1} \left|h_k\right|^2 = \frac{1}{N}\sum_{n=0}^{N-1} \left|H_n\right|^2. \tag{3}$$

The one-side spectrum analysis method that satisfies the Parseval theorem is shown in

$$\begin{gathered} P(f_0) = P(0) = \frac{1}{N^2}\left|H_0\right|^2, \\ P(f_0) = \frac{1}{N^2}\left[\left|H_n\right|^2 + \left|H_{N-n}\right|^2\right], \quad n = 1, 2, \ldots, \left(\frac{N}{2}-1\right), \\ P\left(f_{n/2}\right) = P(f_c) = \frac{1}{N^2}\left|H_{N/2}\right|^2. \end{gathered} \tag{4}$$

This study extracted the absolute values of the power spectrum of theta (4–8 Hz), alpha (8–14 Hz), beta (14–30 Hz), and gamma (30–50 Hz) domains through the one-side power spectrum analysis, and used it as an input value of the fuzzy control system to classify the emotion.

Table 2: The fuzzy condition using EEG data.

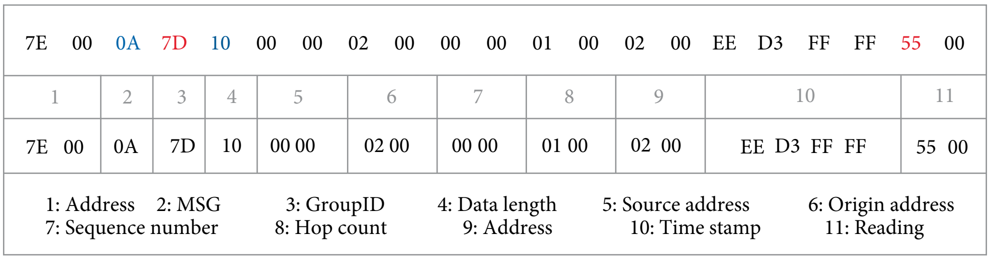
If theta is high and alpha, beta, gamma is low then emotion is stability
If alpha is high, and beta, theta, gamma is low then emotion is relaxation
If beta is high and alpha, theta, gamma is low then emotion is tension
If gamma is high and alpha, beta, theta is low then emotion is excitement

To classify the emotion of the user, the fuzzy model should be constructed by quantifying the physical information that can identify the emotion of the user from the EEG data. To this end, the correlation of α, β, θ, and γ values, i.e., the absolute power values acquired from the user with the user emotion, was investigated to define the rules. Then, the fuzzy model was implemented as shown in Figure 2. In other words, the biometric emotion classification (stability, relaxation, tension, and excitement) was performed by applying the four fuzzy rules using the absolute power values α, β, θ, and γ obtained from the EEG data of the user as the input values. The fuzzy control rules are expressed as linguistic control rules of "IF Ai THEN Bi" format. To create a set of control rules, multiple rules are collected and by making an inference based on the rules, an output value is obtained. Table 2 shows the fuzzy condition for expressing the output value according to the input variable by using the EEG data.

Defuzzification refers to a process of converting a fuzzy quantity result, which is obtained from the fuzzy inference, to a representative value. In this study, the center of gravity method was used for the fuzzy inference [32, 33]. Equation (5) shows the inference output function defuzzificated by the four fuzzy rules.

$$C' = \frac{\sum_i b_i \int \mu c_i'(z)}{\sum_i \int \mu c_i'(z)}. \tag{5}$$

To measure the pulse rate and blood pressure data of the user, this study used the ZigBee wireless sensor network. ZigBee of IEEE 802.15.4 refers to a short-range wireless communication technology that focuses on applications requiring low power consumption, low speed, and low cost [34]. A sensor module of integrating pulse rate and blood pressure sensors was used to acquire the data. Furthermore, the data were measured by using a Telos platform series as the process board, MSP430 MCU, and CC256XQFNEM. If the pulse rate and blood pressure data are produced in single packets separately and used,



7E 00 0A 7D 10 00 00 02 00 00 00 01 00 02 00 EE D3 FF FF 55 00

1	2	3	4	5	6	7	8	9	10	11
7E 00	0A	7D	10	00 00	02 00	00 00	01 00	02 00	EE D3 FF FF	55 00

1: Address 2: MSG 3: GroupID 4: Data length 5: Source address 6: Origin address
7: Sequence number 8: Hop count 9: Address 10: Time stamp 11: Reading

Figure 3: The sensed sensor data structure.

energy is consumed based on additional traffic and data transmission. Therefore, they were bundled in one packet and then sent to the database. Figure 3 shows the structure of sensed biometric data. MSG shows the type of biometric information and enables to isolate the type of pulse rate, i.e., systolic or diastolic blood pressure. GroupID shows the sensor information and each sensor has a GroupID. Timestamp is the time when the sensor measured the data. Reading expresses the actually measured data values in hexadecimal 2 bytes.

Various learning algorithms are used to match the biometric emotion information according to the stress index. Nevertheless, this study used the SVM algorithm, which can solve the nonlinear discrimination problem of multilayer perceptron structure because the EEG, pulse rate, and blood pressure data used in this study consist of nonlinear data structures. The goal of the SVM algorithm is to find a classification boundary by maximizing the margin between the closest observation values (support vectors) in two classes. Furthermore, even when linear separation is difficult, classification is facilitated by using a nonlinear hyperplane through a kernel function, and consequently, it is used in diverse areas such as biology, and image and text recognition. To optimize the performance level of learning data, most traditional pattern recognition methods are based on a risk minimization method. In the case of the SVM algorithm, it is based on a structural risk method to minimize the probability of incorrect classification of the data having fixed but unknown probability distribution [35]. When data cannot be linearly separated like the data of this study, i.e., when they have a pattern that cannot be completely separated because they are overlapped with each other at the linear separation boundary, a slack variable ξ_i is used considering a case of incorrect classification, as shown in eqs. (6) and (7)

$$\text{minimize}: J(w, \xi) = \frac{1}{2}|w|^2 + C\sum_{i=1}^{N}\xi_i, \tag{6}$$

$$\text{subject to}: \begin{cases} y_i\left(w^T x_i + b\right) \geq 1 - \xi_i, \ \forall i, \\ \xi_i \geq 0, \ \forall i, \end{cases} \tag{7}$$

However, C is a trade-off parameter and the Lagrangian function is expressed by

$$\begin{aligned} J(w, b, \alpha) = &\frac{1}{2}w^T w + C\sum_{i=1}^{N}\xi_i \\ &- \sum_{i=1}^{N}\alpha_i\left(y_i\left(w^T x_i + b\right) - 1 + \xi_i\right) - \sum_{i=1}^{N}\mu_i\xi_i. \end{aligned} \tag{8}$$

Table 3: The SVM algorithm configuration.

Algorithm: SVM
Number of data for learning: N
Inputs: sample x to classify dataset : I_i
I_{i1} : EEG, I_{i2} : Pulse, I_{i3} : Systolic Blood Pressure,
I_{i4} : Diastolic Blood Pressure
Output : decision $y \in \{-1, 1\}$
Classify using SVM Algorithm, get the result in the form of a real number.

The above Equation (8) can be converted to Equations (9) and (10) by applying a Lagrangian optimization method.

$$\text{minimize}: \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{i=j}^{N}\alpha_i\alpha_j y_i y_j x_i^T x_j, \tag{9}$$

$$\text{subject to}: \begin{cases} \sum_{i=1}^{N}\alpha_i y_i = 0, \\ 0 \leq \alpha_i \leq C, \quad \forall i = 1, \ldots, N, \end{cases} \tag{10}$$

If the above equations are solved, a decision function can be obtained, as shown in Eq. (11)

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{N} y_i\alpha_i\left(x_i^T x\right) + b\right). \tag{11}$$

The SVM algorithm developed for binary classification has many difficulties when solving problems having many classes in a real environment, one-against-all and one-against-one methods have been proposed. Among them, the one-against-one method consists of $k(k-1)/2$ quantity of the SVM algorithm when k classes are inputted. Furthermore, for respective learning data that consist of data showing two groups they belong to, as the number of learning data used is small when performing the learning, the learning speed is fast [35, 36]. Therefore, this study conducted experiments by using the one-against-one method to improve the learning performance and composed the SVM algorithm as shown in Table 3.

As various kinds of emotions can be expressed according to the external environment, it is effective to predefine the emotion colors and music to be used. Accordingly, for the classification of emotion colors, this study selected 20 color emotion models as the representative elements from HP's "the Meaning of Color." Through HP's color table and the emotion vocabulary matching result investigated in this study, the

Table 4: The common emotional words analysis according to color.

Color	Meaning	Color	Meaning
Bright red	Optimistic, dynamic, energizing, exciting, sexy, intense, stimulating, aggressive, powerful, energetic, dangerous	Orange	Ambition, fun, happy, energetic, balance, flamboyant, warmth, enthusiasm, generosity, vibrant, expansive, organic
	International significance: China = good luck; India = purity; Eastern cultures = signifies joy when combined with white		International significance: Ireland = religious significance(protestant)
Burgundy	Vigor, elegance, richness, refinement, leadership, maturity, expensive	Light blue	Peace, tranquillity, quiet, cool, clean, soft, pure, understanding
Blue	Truth, healing, tranquility, stability, peace, harmony, wisdom, trust, calm, confidence, protection, security, loyalty	Purple	Spirituality, royalty, mystery, wisdom, transformation, independence, enlightenment, respect, wealth
	International significance: China = immortality; Hindus = color of Krishna	Navy	Dignity, credibility, strength, authority, conservative, trustworthiness, traditional, quiet, confident, serene
Green	Nature, envy, healing, fertility, good luck, hope, stability, success, generosity	Beige	Earthy, classic, neutral, warm, soft, bland, melancholy
	International significance: China & France = negative significance for package goods; India = color of Islam; some tropical countries = danger	Greenish yellow	Tart, fruity, acidy, jealousy
		Lime	Tart, fruity, acidy, refreshing, lively, revitalizing
Brown	Stability, masculinity, reliability, comfort, endurance, simplicity, friendship	Terracotta	Wholesome, earthy, country, welcoming, warmth, stability, fall, harvest
	International significance: Colombia = discourages sales; India = the color of mourning	Lavender	Enchantment, nostalgia, delicacy, floral, sweet, fashion
Light pink	Love, romance, softness, delicacy, sweetness, friendship, tenderness, fidelity, compassion	Teal blue	Emotional healing, pleasing, rich, protection, unique, expensive
Bright yellow	Cheeriness, joy, action, optimism, happiness, idealism, summer, hope, imagination, sunshine, philosophy, youth,	Olive green	Traditional color of peace, camouflage, classic, adventure
	International significance: Asia = sacred, imperial		International significance: Military
Fuchsia	Hot, sensual, exciting, bright, fun, energetic, feminine	Neural gray	Neutral, corporate, classic, practical, cool, timeless, quiet, quality

classification was performed based on the common emotion vocabulary, as shown in Table 4.

This study classified the colors and music corresponding to the biometric emotion information according to the stress index by using wavelength, which is a common characteristic of emotion information, color, and sound. As wavelength and frequency have an inversely proportional relationship in physics and mathematics, they can be converted to each other mathematically. Based on C (on the musical scale), the relationship with the wavelength ratio of D and E is 1 : 4/5 : 2/3. This ratio is consistent with the ratio of the respective wavelength of three primary colors, i.e., red, green, and blue: 650 nm, 520 nm, and 433 nm. As such, C, D, and E are similar to the three primary colors, which can be used to create countless colors through appropriate mixing. Therefore, if the wavelength ratio of dodecatonic scale based on the equal temperament is corresponded sequentially to the frequency of colors that can be created with the combination of the three primary colors, the colors and sounds can be linked [37].

By analyzing the color measurement values and the musical scale of music, the optimal colors and list of music corresponding to the biometric emotion information are determined according to the stress index. Music, which is an important emotion information for understanding and analyzing the emotions, can heal the emotion of the user and relieve stress. Therefore, to recommend music based on the biometric emotional state according to the stress index, this study composed a music list based on the data collected from "Samsung Idea" for musical therapy. Table 5 classifies the biometric emotion information according to the stress index and shows the classification of the corresponding colors and music.

4. Performance Evaluation and Experiment Results

In this study, the experiments were conducted by using the EEG, pulse rate, and blood pressure as input data to classify the biometric emotion information according to the stress index. Table 6 classifies the four states (stability, relaxation, tension, and excitement) based on the classification standard of biometric emotion information according to the stress index [38, 39].

This study used the SVM algorithm to design and evaluate the classification of respective emotions. The SVM algorithm is a classification algorithm that determines the discrimination boundary in order to have the largest distance between the discrimination boundary and each class. The kernel of the SVM algorithm used in this study was the radial basis function

Table 5: The corresponding emotional colors and corresponding emotional music according to stress index.

Stress index	Bio emotion information	Corresponding emotion colors	Corresponding emotion music
Stage 1 (00–30)	Stability	Yellow, Brown	Franz Peter Schubert—10 songs besides the lullabies
Stage 2 (31–60)	Relaxation	Red	Antonio Vivaldi—10 other songs besides the four seasons (spring)
Stage 3 (61–70)	Tension	Blue	Franz Peter Schubert—10 songs besides the Ave Maria
Stage 4 (71–100)	Excitement	Green	Robert Alexander Schumann—10 songs besides the dream

Table 6: The classification of bio emotion information according to stress index.

Stress index	Bio emotion information	EEG	Pulse	Blood pressure	
		Hz	Times/Minute	Diastolic	Systolic
Stage 1	Stability	Theta (4–7)	70–80	80	120
Stage 2	Relaxation	Alpha (8–14)	60–70	81–89	121–139
Stage 3	Tension	Beta (14–30)	80–90	90–99	140–159
Stage 4	Excitement	Gamma (30–50)	Under 60, Over 90	Over 100	Over 160

Table 7: The confusion matrix.

Type		True condition			
		Stability	Relaxation	Tension	Excitement
Prediction condition	Stability	90.4	11.4	0.0	0.0
	Relaxation	9.6	83.4	3.4	0.5
	Tension	0.0	5.2	84.5	13.8
	Excitement	0.0	0.0	12.1	85.7

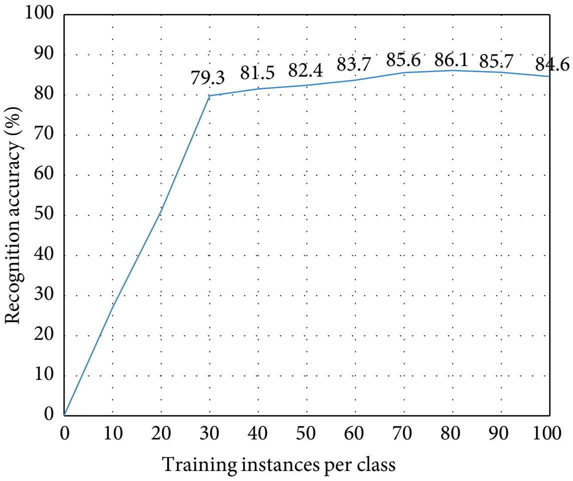


Figure 4: The bio emotion classification accuracy.

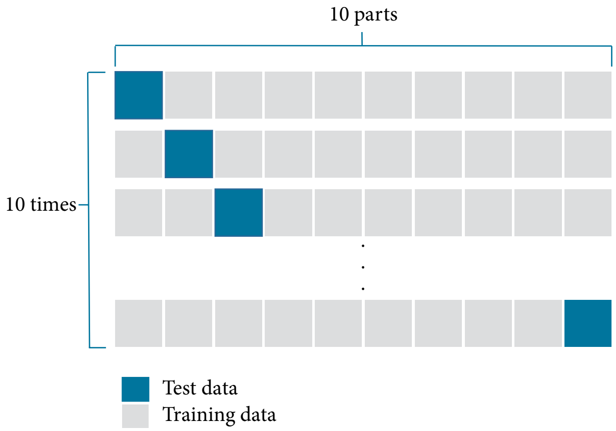


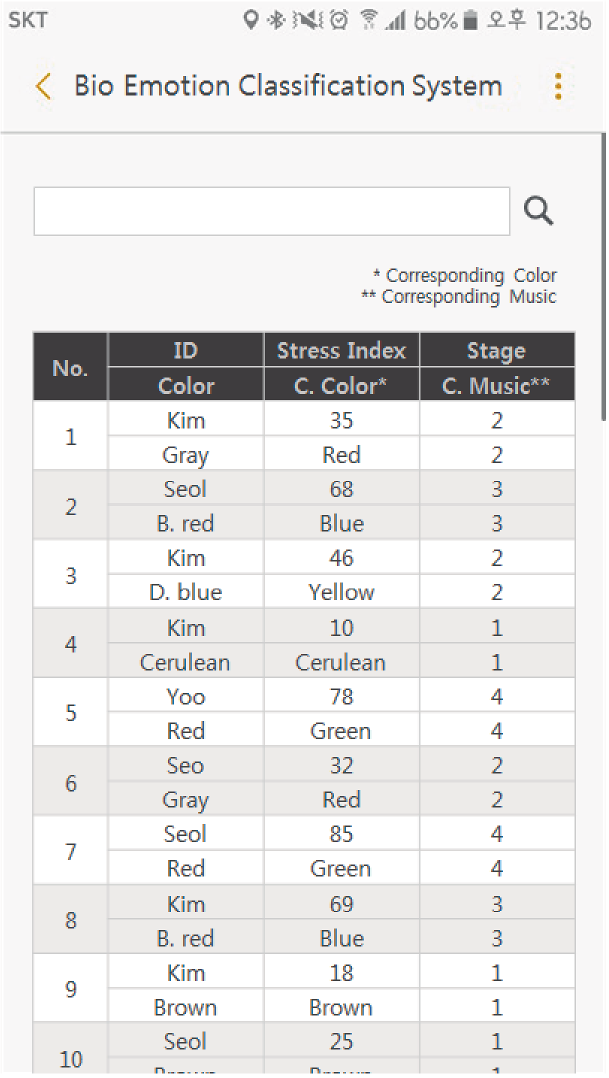
Figure 5: The 10-fold cross validation.

kernel; the radius of the kernel was set to 1 and the margin of the SVM algorithm was set to 1 [40, 41]. Figure 4 shows the accuracy result of classifying according to the four types of biometric emotion information through the SVM algorithm. When 80% of data were learned according to the ratios of learning data, the highest performance of 86.1% was shown. Furthermore, Table 7 shows the error matrix when 80% of data were learned. The classification results were 90.4% for stable, 83.4% for relaxed, 84.5% for tensed, and 85.7% for excited.

Furthermore, to evaluate the tracking performance of the proposed system that recommends colors and music according to the biometric emotion information and stress index, 10-fold cross validation was conducted, thereby minimizing the influence of training data and ensuring reliability. The 10-fold cross validation divides the entire data into ten equal parts and uses nine parts as training data and the remaining part as test data. The 10-fold cross validation repeats the training and testing ten times in total to evaluate the overall performance, and every time, one equal part of data used for the test is changed. Figure 5 shows an example of 10-fold cross validation.

Table 8: The evaluation of data performance for verification.

Fold no.	Accuracy (unit: %)
1	84.8
2	87.5
3	88.6
4	86.4
5	87.3
6	83.3
7	89.6
8	87.4
9	83.2
10	85.6



SKT 66% 오후 12:36

Bio Emotion Classification System

* Corresponding Color
** Corresponding Music

No.	ID	Stress Index	Stage
	Color	C. Color*	C. Music**
1	Kim	35	2
	Gray	Red	2
2	Seol	68	3
	B. red	Blue	3
3	Kim	46	2
	D. blue	Yellow	2
4	Kim	10	1
	Cerulean	Cerulean	1
5	Yoo	78	4
	Red	Green	4
6	Seo	32	2
	Gray	Red	2
7	Seol	85	4
	Red	Green	4
8	Kim	69	3
	B. red	Blue	3
9	Kim	18	1
	Brown	Brown	1
10	Seol	25	1

Figure 6: The corresponding emotion color and corresponding emotion music recommendation system according to bio emotion information and stress index.

This study divided the experimental data into evaluation data and validation data using the ratio 7 : 3. Furthermore, the system was optimized by using the leave-one-out method. Table 8 shows the average accuracy of 86.4% in the result of validation data obtained through the performance evaluation. Therefore, the performance of the recommendation system is high.

Figure 6 shows the result of implementing an application for the proposed system that recommends the colors and music according to the biometric emotion information and stress index. The emotion colors and the emotion music are classified into the corresponding items according to the biometric emotion information and stress index.

5. Conclusion

This study aimed to classify and recommend the emotion colors and the emotion music corresponding to the emotion of the user by measuring biometric information. Accordingly, after measuring the EEG, pulse rate, and blood pressure data (i.e., the biometric information), the emotion colors and emotion music are classified according to the current biometric emotion information and stress index of the user through real-time emotion analysis. To classify the EEG data into four types of biometric emotion information (stability, relaxation, tension, and excitement), a fuzzy control system was designed and the pulse rate and blood pressure data were composed in single packets separately and sent to the database. When 80% of data were learned according to the ratio of the training data by using the SVM algorithm to classify the EEG, blood pressure, and pulse rate data based on the biometric emotion information, the highest performance of 86.1% was shown. Moreover, the error matrix classification results obtained were 90.4% for stable, 83.4% for relaxed, 84.5% for tensed, and 85.7% for excited, confirming that the efficiency was high. Furthermore, when the 10-fold cross validation was performed to evaluate the tracking performance of the system that recommends the emotion colors and emotion music according to the biometric emotion information and stress index, an average accuracy of 86.4% was demonstrated, confirming that the performance of the recommendation system was high. This study proposed a biometric emotion information classification system using the stress index to classify the emotion colors and emotion music intelligently based on the emotion of the user. Therefore, it is expected to contribute to studies on HCI in the fourth industrial revolution era. In a future study, the emotions will be classified based on the situational and environmental factors as well as the biometric information of the user. Furthermore, the accuracy and efficiency of the proposed system will be enhanced by using various types of biometric information in addition to the EEG, pulse rate, and blood pressure used in this study, as measures for determining biometric emotions.

Data Availability

The BIOPAC MP 150 EEG data used to support the findings of this study have been deposited in the BrainAmp by Brain Products, Munich, Germany.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea Government (MSIT) (No. 2019R1F1A1041186). This research was supported by the Korean MSIT (Ministry of Science and ICT), under the National Program for Excellence in SW (2017-0-00137), supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation).

References

[1] A. Y. Kim, E. H. Jang, and J. H. Sohn, "Classification of negative emotions based on arousal score and physiological signals using neural network," *Korean Society for Emotion and Sensibility*, vol. 21, no. 1, pp. 177–186, 2018.

[2] L. Santamaria-Granados, M. Munoz-Organero, G. Ramirez-Gonzalez, E. Abdulhay, and N. Arunkumar, "Using deep convolutional neural network for emotion detection on a physiological signals dataset (AMIGOS)," *IEEE Access*, vol. 7, pp. 57–67, 2017.

[3] R. W. Picard, "Affective computing: challenges," *International Journal of Human–Computer Studies*, vol. 59, no. 1-2, pp. 55–64, 2003.

[4] S. M. Park, Y. H. Lee, K. E. Ko, and K. B. Sim, "Development of EEG signals measurement and analysis method based on timbre," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 20, no. 3, pp. 388–393, 2010.

[5] W. Apriadi, S. K. Wijaya, and L. O. H. Prawito Zilullah, "Development of electroencephalogram (EEG) based on ADS1 299EEGFE-PDK and LaunchPad MSP432P401R," in *2017 5th International Conference on Instrumentation, Communications, Information Technology, and Biomedical Engineering (ICICI-BME)*, IEEE, Indonesia, 2017.

[6] Y. H. Cho and K. S. Park, "A study on the improvement of emotion recognition by gender discrimination," *Journal of the Institute of Electronics Engineers of Korea*, vol. 45, no. 4, pp. 107–114, 2008.

[7] M. V. Villarejo, B. G. Zapirain, and A. M. Zorrilla, "Algorithms based on CWT and classifiers to control cardiac alterations and stress using an ECG and a SCR," *Sensors (Basel, Switzerland)*, vol. 13, no. 5, pp. 6141–6170, 2013.

[8] M. R. Ogiela and L. Ogiela, "On using cognitive models in cryptography," in *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, Institute of Electrical and Electronics Engineers, pp. 1055–1058, Crans-Montana, Switzerland, 2016.

[9] L. Ogiela and M. R. Ogiela, "Insider threats and cryptographic techniques in secure information management," *IEEE Systems Journal*, vol. 11, no. 2, pp. 405–14, 2017.

[10] N. Mitsuo, "Kansei engineering: a new ergonomic consumer-oriented technology for product development," *International Journal of Industrial Ergonomics*, vol. 15, no. 1, pp. 3–11, 1995.

[11] W. Hongwei and M. M. Jerry, "Classification of battlefield ground vehicles using acoustic features and fuzzy logic rule-based classifiers," *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 1, pp. 56–72, 2007.

[12] J. Nicholson, K. Takahashi, and R. Nakatsu, "Emotion recognition in speech using neural networks," *Neural Computing and Applications*, vol. 9, no. 4, pp. 290–296, 2000.

[13] C. C. Lee, E. Mower, C. Busso, S. Lee, and S. Narayanan, "Emotion recognition using a hierarchical binary decision tree approach," *Speech Communication*, vol. 53, no. 9-10, pp. 1162–1171, 2011.

[14] K. E. Ko, H. C. Yang, and K. B. Sim, "Emotion recognition using EEG signals with relative power values and Bayesian network," *International Journal of control Automation and Systems*, vol. 7, no. 5, pp. 865–70, 2009.

[15] V. Shusterman and O. Barnea, "Sympathetic nervous system activity in stress and biofeedback relaxation," *IEEE Engineering in Medicine and Biology Magazine*, vol. 24, no. 2, pp. 52–57, 2005.

[16] J. Bakker, M. Pechenizkiy, and N. Sidorova, "What's your current stress level? Detection of stress patterns from GSR sensor data," in *Proceedings of the IEEE International Conference on Data Mining Workshop*, pp. 573–580, IEEE, 2011.

[17] J. A. Healey and R. W. Picard, "Detecting stress during real-world driving tasks using physiological sensors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 2, pp. 156–66, 2005.

[18] Y. C. Jung and Y. I. Yoon, "Monitoring senior wellness status using multimodal biosensors," in *Proceedings of the 2016 International Conference on Big Data and Smart Computing (BigComp)*, pp. 435–438, IEEE, Hong Kong, China, 2016.

[19] C. Setz, B. Arnrich, J. Schumm, R. La Marca, G. Tröster, and U. Ehlert, "Discriminating stress from cognitive load using a wearable EDA device," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 2, pp. 410–417, 2010.

[20] P. Melillo, M. Bracale, and L. Pecchia, "Nonlinear heart rate variability features for real-life stress detection. Case study: students under stress due to university examination," *BioMedical Engineering OnLine*, vol. 10, no. 1, p. 96, 2011.

[21] H. Kurniawan, A. V. Maslov, and M. Pechenizkiy, "vStress detection from speech and galvanic skin response signals," in *Proceedings of the 26th IEEE International Symposium on Computer Based Medical Systems*, pp. 209–214, IEEE, 2013.

[22] N. Sharma and T. Gedeon, "Objective measures, sensors and computational techniques for stress recognition and classification: a survey," *Computer Methods and Programs in Biomedicine*, vol. 108, no. 3, pp. 1287–1301, 2012.

[23] F. T. Sun, C. Kuo, H. T. Cheng, S. Buthpitiya, P. Collins, and M. Griss, "Activity-aware mental stress detection using physiological sensors," in *Proceedings of the International Conference on Mobile Computing, Applications, and Services*, pp. 211–230, Springer, 2010.

[24] P. Ruvolo and J. Movellan, "Automatic cry detection in early childhood education settings," vol. 7, in *Proceedings of the IEEE International Conference on Development and Learning*, IEEE, pp. 204–208, Monterey, CA, USA, 2008.

[25] A. Rosales-Pérez, C. A. Reyes-García, J. A. Gonzalez, O. F. Reyes-Galaviz, H. J. Escalante, and S. Orlandi, "Classifying infant cry patterns by the genetic selection of a fuzzy model," *Biomedical Signal Processing and Control*, vol. 17, pp. 38–46, 2015.

[26] L. Abou-Abbas, H. F. Alaie, and C. Tadj, "Automatic detection of the expiratory and inspiratory phases in newborn cry signals,"

Biomedical Signal Processing and Control, vol. 19, pp. 35–43, 2015.

[27] T. Y. Kim, D. W. Seo, and S. H. Bae, "The study of bio emotion cognition follow stress index number by multiplex SVM algorithm," *The Journal of Korea Institute of Information, Electronics, and Communication Technology*, vol. 5, no. 1, pp. 45–51, 2012.

[28] S. Fazli, J. Mehnert, J. Steinbrink et al., "Enhanced performance by a hybrid NIRS–EEG brain computer interface," *NeuroImage*, vol. 59, no. 1, pp. 519–529, 2012.

[29] M. H. Lee, S. Fazli, J. Mehnert, and S. W. Lee, "Hybrid brain-computer interface based on EEG and NIRS modalities," in *2014 International Winter Workshop on Brain-Computer Interface (BCI)*, IEEE, Jeongsun-kun, South Korea, 2014.

[30] G. Peharz and A. Ulm, "Quantifying the influence of colors on the performance of c-Si photovoltaic devices," *Renewable Energy*, vol. 129, no. A, pp. 299–308, 2018.

[31] L. Y. Hu and H. Y. Fan, "Inversion formula and Parseval theorem for complex continuous wavelet transforms studied by entangled state representation," *Chinese Physics B*, vol. 19, no. 7, Article ID 074205, 2010.

[32] I. H. Bae, C. H. Kim, and H. T. Noh, "Design and evaluation of a fuzzy logic based multi-hop broadcast algorithm for IoT applications," *KSII Transactions on Internet and Information Systems*, vol. 17, no. 6, pp. 17–23, 2016.

[33] C. Wu, S. Ohzahata, Y. Ji, and T. Kato, "Joint fuzzy relays and network-coding-based forwarding for multihop broadcasting in VANETs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1415–1427, 2014.

[34] K. Pat, "IEEE 802.15 WPAN™ Task Group 4," 2019, http://ieee802.org/15/pub/TG4.html.

[35] T. W. Rauber, F. D. A. Boldt, and F. M. Varejãao, "Heterogeneous feature models and feature selection applied to bearing fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 1, pp. 637–646, 2015.

[36] S. U. Jan, Y. D. Lee, J. P. Shin, and I. S. Koo, "Sensor fault classification based on support vector machine and statistical time-domain features," *IEEE Access*, vol. 5, pp. 8682–8690, 2017.

[37] T. Y. Kim, B. H. Song, and S. H. Bae, "A design and implementation of music & image retrieval recommendation system based on emotion," *Journal of the Institute of Electronics Engineers of Korea*, vol. 47, no. 1, pp. 73–79, 2010, UCI : G704–A00451, 2010.47.1.002.

[38] H. J. Kim, S. H. Lee, and H. G. Lee, "Development of stress index model and u-SMC (stress management center) business model from the context-aware computing perspective," *Journal of Intelligent Information System*, vol. 14, no. 2, pp. 21–44, 2008, UCI : G704–000721, 2008.14.2.001.

[39] S. A. Hosseini, M. A. Khalilzadeh, and S. Changiz, "Emotional stress recognition system for affective computing based on bio-signals," *Journal of Biological Systems*, vol. 18, no. 1, pp. 101–114, 2010.

[40] J. S. Kang, G. J. Jang, and M. H. Lee, "Stress status classification based on EEG signals," *The Journal of the Institute of Internet Broadcasting and Communication*, vol. 16, no. 3, pp. 103–108, 2016.

[41] L. Saeed, S. Shima, K. Reza, A. Alireza, and H. Boshra, "Support vector machine classification of brain states exposed to social stress test using EEG-based brain network measures," *Biocybernetics and Biomedical Engineering*, vol. 39, no. 1, pp. 199–213, 2019.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 9139074, 14 pages
https://doi.org/10.1155/2020/9139074

Research Article

An Improved Automatic Traffic Incident Detection Technique Using a Vehicle to Infrastructure Communication

Muhammad Sameer Sheikh[1,2] **Jun Liang**[2] **and Wensong Wang**[3]

[1] *School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China*
[2] *Department of Automotive and Transportation Engineering, Automotive Engineering Research Institute, Jiangsu University, Zhenjiang 212013, China*
[3] *School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore*

Correspondence should be addressed to Muhammad Sameer Sheikh; sameer@ujs.edu.cn and Jun Liang; liangjun@ujs.edu.cn

Received 12 June 2019; Accepted 9 October 2019; Published 13 January 2020

Guest Editor: Kandaraj Piamrat

Traffic incident detection is one of the major research areas of intelligent transportation systems (ITSs). In recent years, many megacities suffer from heavy traffic flow and congestion. Therefore, monitoring traffic scenarios is a challenging issue due to the nature and the characteristics of a traffic incident. Reliable detection of traffic incidents and congestions provide useful information for enhancing traffic safety and indicate the characteristics of traffic incidents, traffic violation, driving pattern, etc. This paper investigates the estimation of traffic incident from a hybrid observer (HO) method, and detects a traffic incident by using an improved automatic incident detection (AID) technique based on the lane-changing speed mechanism in the highway traffic environment. First, we developed the connection between vehicles and roadside units (RSUs) by using a beacon mechanism. Then, they will exchange information once the vehicles get access to a wireless medium. Second, we utilized the probabilistic approach to collect the traffic information data, by using a vehicle to infrastructure (V2I) communication. Third, we estimated the traffic incident by using an HO method which can provide an accurate estimation of an event occurring. Finally, in order to detect traffic incident accurately, we applied the probabilistic data collected through V2I communication based on lane-changing speed mechanism. The experimental results and analysis obtained from simulations show that the proposed method outperforms other methods in terms of obtaining a better estimation of traffic incident which agrees well with the theoretical incident, around 30% faster detection of traffic incidents and 25% faster dissipation of traffic congestion. With regard to duration of an incident, the proposed system obtained a better Kaplan–Meier (KM) curve, influenced by the shortest duration of time to clear the traffic incident, in comparison with the other methods.

1. Introduction

In recent years, intelligent transportation systems (ITSs) draw a great deal of attention for the researcher of wireless and communication technology background. This raises concern to the transportation authorities because of the large number of vehicles on the road causing traffic incidents, congestions, road bottlenecks, etc. ITS integrates wireless communication technology with the transport networks in order to provide traffic safety, reduce traffic congestion, and improve traffic management [1, 2]. In addition to the traffic safety, ITS also provides entertainment services on vehicles such as climate information, internet access, etc. In many global cities, people are using private cars, taxi and bus to commute to their destination. Because traffic conditions on the road can rapidly become severe, it can affect the transport operations. Specifically, many metropolitan cities are suffering from severe traffic congestions in the urban and highway traffic environments, which are caused by the traffic incident [3]. As a consequence, the loss and disturbance caused by the traffic incident, is directly associated with the duration of the traffic incident which, may further deteriorate the traffic flow. In this context, early detection of incidents is necessary to investigate and to implement the traffic strategy for an ITS. These early incident detections can alleviate traffic congestion and hence improve the traffic flow for real-time traffic monitoring system in ITS [4].

The traffic incident is referred to as an abrupt change in traffic flow, which reduces the road capacity and increases traffic congestion. In the past, a traffic accident was very difficult to analyze due to the nature of an incident which is always changing and this makes the detection more complex for transportation authorities. Such complexity can cause the failure of the transportation management system. Therefore, there is a great need of designing a sophisticated algorithm, which is able to estimate and detect traffic incident. The challenging issue in ITS is to estimate and detect an incident from traffic congestion scenarios [5]. Generally speaking, an incident is referred to as an occurrence of an event that creates disturbance to the normal traffic flow [6]. Surveillance cameras are placed on the road to detect the traffic incident. An incident detection and the classification are a very important aspect in traffic management system. The traffic management system has the ability to conduct automatic incident classification (AIC) to evaluate different types of incidents [7].

As mentioned, in the past, many methods have been proposed which are used to collect data from the detectors such as loop detector, radar detector, video detector, etc. In this context, V2I communication system is used to collect data from vehicles to detect traffic incidents. V2I communication is a robust system, which is considered as a highly contributive in ITS [8, 9]. The collected data are processed by AID algorithms, which can generate incident alerts in case of any traffic incident and violations. Traffic incidents can be detected by using global positioning system (GPS) detector. Asakura et al. [10] presented the properties of traffic flow dynamics under incident by using floating data gathered from GPS. In this scheme, a proposed method is able to predict the time and location of traffic congestion influenced by a traffic incident. Surveillance cameras are placed on the road to detect the traffic incident. Incident detection and classification are a very important aspect of the traffic management system. The traffic management system has the ability to conduct automatic incident classification (AIC) to evaluate different types of incidents [7]. Ren et al. [11] presented a video-based technique to monitor and detect the traffic incident by evaluating the distribution characteristics of the traffic states on the road section. An incident detection system (IDS) play an important role in ITS which gains a lot of attention from the research community in recent years. The IDSs are designed to detect incident or unpleasant situation such as traffic incident, traffic violation and traffic congestion by using communication technology [5, 6]. The main challenging issue for an ITS is to obtain an early and accurate detection of traffic incident [12].

In past decades, machine learning techniques have been widely utilized to detect traffic incident. Many artificial neural network (ANN) were discussed [7]. Ritchie and Cheu [13] introduced an ANN technique which is able to detect the traffic incident with a better performance. However, the collection of ANN parameters is very complex and difficult to obtain. A hybrid approach is introduced by combing time series analysis and machine learning schemes to detect incidents [14]. This approach may detect traffic incidents accurately. Jin et al. [15] introduced the constructive probabilistic neural network (CPNN) in a highway traffic environment. This model is tested on I-880 and evaluated by considering online and offline traffic situations. However, this approach is able to detect only small traffic incidents.

To enhance the performance of AID, support vector machine (SVM) was introduced to detect traffic incident [4, 16, 17]. In [4], two SVMs were trained and simulated on the traffic incident data. This method did not produce a robust result because the selection criteria of SVM parameters and kernels are always very complex during the training process in order to construct a sample. Xiao [18] introduced SVM and *k*-Nearest Neighbor (KNN) ensemble learning method to detect traffic incident. This model trains SVM and KNN learning, and combine them to obtain better results. Wang et al. [7] introduced an incident classification of traffic data by using SVM method. However, this method cannot support small traffic data and require a much longer time to process the traffic data due to the characteristics of the ST signals. In social media such as Twitter, it has become the most famous tool used to gather information and have a large user's account database, which can share a portion of data to the public using APIs [19]. In recent years, many works have been proposed to detect the traffic incident by analyzing the location and time of an incident from tweets. In Ref. [19], Gu et al. introduced a real-time traffic detection from Twitter using the REST API. The proposed method utilizes Semi-Naive Bayes (SNB) method to detect five different incidents and obtain better performance. However, the processing of tweets from different incident situations often required large computational time. Schulz et al. [20] introduced a method to detect small incident by analyzing microblog. This method obtains a better detection of an incident, but only applicable for the low-level applications.

Dabiri and Heaslip [21] proposed a framework which is able to monitor and detect traffic incidents based on Tweeter, by using deep learning method. The proposed method utilized the numerous amount of Tweets to evaluate the traffic event condition and required a large computational time to process these tweets. Zhang et al. [22] introduced a new method to detect traffic incidents from Tweets using a deep learning method. The proposed method utilized millions of Tweets and was applied in two megacities. It also achieved better traffic incident detection, while consuming a large amount of time to process millions of Tweets. Paule et al. [23] proposed a method for geo-localization tweet, by utilizing a weight-voting algorithm where the weight of tweets votes depend on the user's reliability. The proposed method obtains a better detection of real-time traffic incident. However, due to an increase in the voting users, the proposed method is limited to the lower coverage and also limited users.

AID algorithm is used to calculate new parameters values from collected data and then compare these values to the threshold values to identify the incident detection. Several famous methods falls in this category such as McMaster Algorithm [24] and California Algorithm [25]. Recently, many approaches have been used to enhance the performance of existing AID schemes, such as integration of V2I communications with Bayesian-based scheme [26], which is focused on less traffic flow to detect an incident. He et al. [27] proposed a hybrid tree-based quantile regression method to predict and evaluate the incident duration. The presented method produced better results as compared to other predictive models. Peeta et al. [28] also proposed a variable message sign (VMS) scheme that only focused on the prediction of incident clearance time because of the delay caused by an accident.

Lu et al. [29] proposed a method to detect traffic incident based on nFoil. This method was implemented on the real traffic data and simulated traffic data generated from Singapore highway. The proposed method produced a better detection of a traffic incident. However, it required a longer time to process traffic data. Wang et al. [30] introduced an efficient multiple model particle filter (EMMPF) to estimate and detect traffic incident. The main idea is to implement an EMMPF to reduce the large computational time, which occurred in traditional AID techniques during the training of datasets. The proposed system is able to reduce the large computational time and the proposed method is only limited to the hybrid system which contains a large model. Based on GPS analysis, D'Andrea and Marcelloni [5] proposed a method to detect traffic incident and congestions to obtain a better incident detection rate. However, the proposed system was unable to differentiate between the traffic incident and congestion event due to the correlation of GPS data gathered from the moving or slow vehicles. Fogu et al. [31] introduced e-Notify system, which is able to detect traffic accident rapidly and also reduce the incident duration time, by implementing efficient communication through the combination of V2I and V2V, respectively. In the past, improved nonparametric regression based model was proposed to detect traffic incidents [32]. Popescu et al. [33] introduced an AID scheme, in which the lane changing distance and lane changing speed mechanisms were utilized to detect the traffic incident based on the collection of traffic-information data by using V2I communication. However, this method required a longer time to process the traffic data in terms of vehicle lane changing distance and also this scheme cannot distinguish the road bottleneck caused by a traffic incident.

The estimation and detection of traffic incident are one of the main challenging issues in the ITS. In the past, previous studies revealed that AID is a well-known and robust technique to detect traffic incident. Also, an AID technique can overcome the traffic congestion at the location where the occurrence of an incident caused the traffic difficulties such as road bottleneck, accident, and disabled vehicles, electronics equipment malfunctioning and other issues which can disrupt the traffic flow. Significant monitoring, estimation, and detection of traffic incident provide relevant traffic-related information to enhance the traffic safety and driving experiences, by providing a driver with real-time traffic information to assist decision. In particular, traffic monitoring, traffic incident management and traffic safety management are the main pillar for enhancing the ITS. This inspired us to further investigate the estimation and detection of the traffic incident. Specifically, the estimation of traffic incident and detection is somehow related to the pattern recognition problem, in which the incident and nonincident must be evaluated and classified. A sophisticated learning method can be applied to AID after training the data. So far, the support vector machine, neural network and deep learning techniques have been utilized to deal with this issue. These techniques depend on the propositional learning systems, which indicates that the data learned from these systems are propositional and not reliable. Also, a few AID algorithms such a McMaster algorithm [24] and California algorithm [25] are used to calculate the new parameter values from the collected data and compare these values to the threshold values to identify the incident detection. The estimation of traffic incident may not be accurate because of the hybrid modeling, in which the traffic incident can occur at any location with different traffic conditions. Estimation of traffic incident with HO method and incident detection with an improved AID technique are not covered well in the above studies. Also, the traffic data were not utilized to analyze the incident conditions.

In this paper, we presented an efficient ITS system, which is able to estimate and detect the traffic incident from the hybrid observer and an improved AID technique, respectively. The proposed system significantly utilized the PWSL observation to estimate the traffic incident and probabilistic collection of traffic data to detect the traffic incident. First, we developed the connection between vehicles and RSUs by using beacon mechanism. Once the connection is developed, they will exchange traffic-related information. Second, we employ the HO method to estimate the traffic incident, these estimations can provide an accurate estimation of an event occurring. Third, in order to detect traffic incident accurately, the proposed method exploits the probabilistic approach to collect the traffic information data by using V2I communication based on the lane changing speed mechanism.

The rest of this paper is structured as follows. Section 2 presents the system modeling in which vehicle signing and beacon signal mechanism have been discussed. Section 3 presents the probabilistic approach to obtain the traffic information data. Section 4 presents the proposed estimation of traffic incident and detection method. Section 5 discussed the comparison of the proposed model with different competent methods. Simulation results are presented in Section 6. Finally, Section 7 concludes this paper.

2. System Model

In this work, we assumed that the vehicles are equipped with a wireless module, which is used to communicate with the RSUs that are placed on the road to exchange traffic-related information with any passing vehicles. In addition, vehicles are also assumed equipped with the event data recorder (EDR) [34], which is used to monitor fast acceleration, speed, and lane information of the vehicles.

Figure 1 shows the system model of highway traffic flow on the road with the movement of vehicles in the forward direction. The RSUs are placed on the road apart from each other with a distance of nearly 1.5 km. These RSUs are able to provide equal coverage in its vicinity. Also, the RSUs situated in the adjacent and on the opposite side of the road, are used to construct infrastructure. Each RSU contains a GPS device to obtain the exact location of vehicles, a radio transceiver for developing a connection between passing vehicles and a computing device that processed traffic information data gathered from vehicles, such as lane changing speed and distance.

As shown in Figure 2, Figure 2(a) illustrates the vehicles are moving in the forward direction with the constant speed,

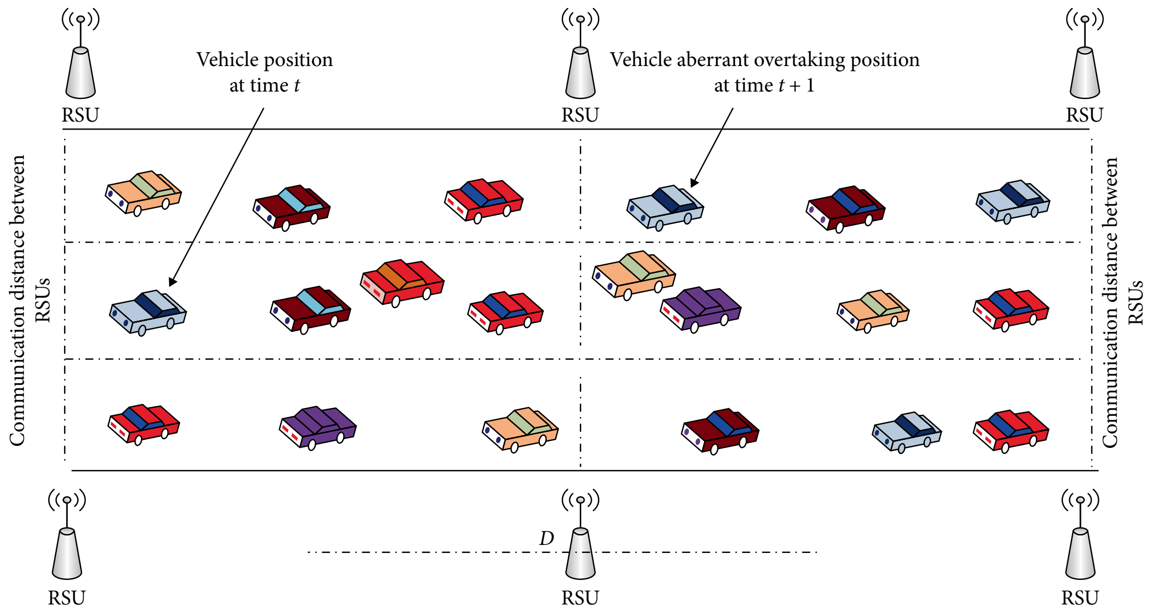
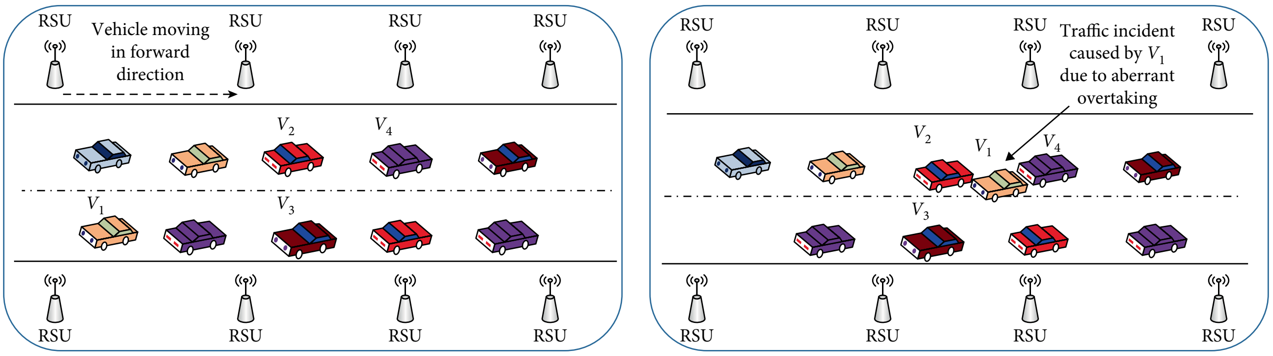

RSU
Vehicle position
at time t
RSU
Vehicle aberrant overtaking position
at time t + 1
RSU
Communication distance between
RSUs
Communication distance between
RSUs
RSU
D
RSU
RSU

Figure 1: Traffic incident influenced by the vehicle aberrant overtaking.



(a)

(b)

Figure 2: Traffic incident scenario. (a) Movement of vehicle in the forward direction, and (b) movement of vehicle with aberrant overtaking.

and Figure 2(b) illustrates an aberrant change in the speed of a vehicle V_1 during lane changing caused the traffic incident.

2.1. Vehicle Signing. In the proposed method each vehicle needs to sign up and register their details with the transport authority (TA). TA is responsible for managing database of the vehicles such as, vehicle ID, personal information of the drivers, and also providing certificates to the vehicles. It is important that all the vehicles must be connected with the TA.

A time-dependent secret $S_{v_{ri}}(t)$ act on behalf of the TA for verifying the identity of the vehicle v_{ri} and when it issued a message last time. The secret can be computed and encrypted as below.

Lets assume the vehicle v_{ri} sent a request for sign-up, to the TA at time t. First, TA will check the identity, *id* of v_{ri}, and then generate a reply which contains three parameters $(K_{v_{ri}}, e_{v_{ri}}, f_{v_{ri}})$, where $K_{v_{ri}}$ is a symmetric key, $e_{v_{ri}}$ and $f_{v_{ri}}$ are two integer values.

$$S_{v_{ri}}(t) = E_{kv_{ri}}\left\{e_{v_{ri}} + nf_{v_{ri}}\right\}. \tag{1}$$

Both the vehicle and TA must initialize the counter to the value of $e_{v_{ri}}$ and increment it by $f_{v_{ri}}$ at every message received by the vehicle v_{ri}.

2.2. Beacon Signal Mechanism. In this section, we discuss the beaconing signal process. After sign-in, each vehicle sends a beacon signal, periodic message, at every time τ_m second to share the detail information such as recent location, pseudo-identity and type of vehicle to the RSU.

Lets assume a vehicle v_{ri} establishs a beacon and transmits, the established beacon is expressed as below.

$$B_{v_{ri}} = \left[\mathrm{PID}_{v_{ri}}, t_i, \mu, S_{v_{ri}}(t), E_{kv_{ri}}[l(t)]\right], \tag{2}$$

where $\mathrm{PID}_{v_{ri}}$ is the pseudo-identity of the vehicle v_{ri}, t_i is used to protect from replay attack, $S_{v_{ri}}(t)$ is the sign-in process of the vehicle. $E_{kv_{ri}}[l(t)]$ is encrypted location and μ is the beacon signal using hash function $H()$ [35].

$$\mu = H\left(\mathrm{PID}_{v_{ri}}, t_i, S_{v_{ri}}(t), E_{kv_{ri}}[l(t)]\right). \tag{3}$$

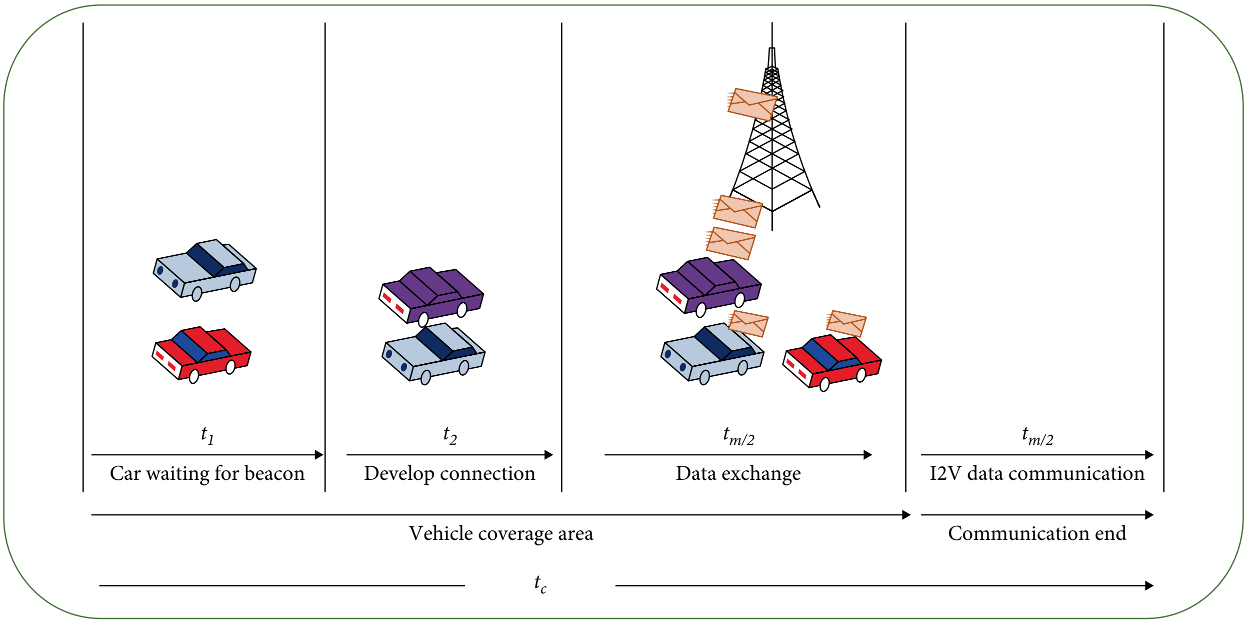


Figure 3: Developing communication between RSU and passing by vehicle.

Equation (3) is used to calculate the beacon signal, lets assume when a witness vehicle v_w received a message, it immediately checks the time-stamp t_i function of the received beacon, and then verify the beacon signal μ. If both values match, then the beacon content is correct. Thereore, the beacon is considered valid within the vicinity of the incident vehicle.

As shown in Figure 3, the total time required to exchange the traffic information between vehicle and RSU is t_c, t_1 is the time when the vehicle are waiting to receive beacon, and at the time t_2 vehicle and RSU start to develop connection. Once the vehicle gets access to the wireless medium, it will exchange information such as ID, speed, acceleration, with RSU at time $t_{m/2}$, and exchange the information from RSU to vehicle at time $t_{m/2}$, respectively.

3. Collection of Traffic Information Data

Due to an enormous amount of traffic flow in the urban traffic environment, data from externalities on the road are often interlinked with other vehicles. Consequently, it is very difficult to collect and manage data from each vehicle passing through RSU. In order to obtain the accurate detection of traffic incident. Firstly, we applied the probabilistic approach to collect the data from the passing vehicles [36].

We assumed that the RSUs are active and able to collect the traffic-related information from the vehicles passing through the RSU with the probability of p_i. In particular, the vehicles are capable of maintaining a database of reliable traffic information to identify the traffic conditions that shows the sign of traffic incidents. The data collected from k number of vehicles with the probability of $1-\beta^n$ can lead to provide optimum aggregation. For some application β ranges from $0<\beta<1$ [36].

Lets assume A_n is an event caused by incident vehicle, i.e. vehicle A (see Figure 4), n number of vehicles passing the road for successful aggregation. Let B be the random variable which traces the number of vehicles that provided traffic data information among the n passing vehicles. So that the equation can be written as follows.

$$\begin{aligned} P_a[A_n] &= \sum_{k=0}^{n} P_a[A_n|B=k]P_a[B=k] \\ &= \sum_{k=0}^{n}\left(1-\beta^k\right)\binom{n}{k}p_i^k\left(1-p_i\right)^{n-k} \\ &= \sum_{k=0}^{n}\binom{n}{k}p_i^k\left(1-p_i\right)^{n-k} - \sum_{k=0}^{n}\binom{n}{k}\left(p_i\beta\right)^k\left(1-p_i\right)^{n-k}. \end{aligned} \tag{4}$$

Equation (4) observed as a binomial function,

where

$$\sum_{k=0}^{n}\binom{n}{k}p_i^k\left(1-p_i\right)^{n-k} = 1, \tag{5}$$

$$\sum_{k=0}^{n}\binom{n}{k}\left(p_i\beta\right)^k\left(1-p_i\right)^{n-k} = \left[1-p_i(1-\beta)\right]^n. \tag{6}$$

Equation (4) can be expressed as below.

$$\mathrm{P_a}[A_n] = 1-\left[1-p_i(1-\beta)\right]^n. \tag{7}$$

Lets assume Φ is the incident target caused by A_n. In order to detect the event which must satisfy the below condition.

$$\Phi \le 1-\left[1-p_i(1-\beta)\right]^n. \tag{8}$$

The transpositions can be yields as below.

$$1-\left[1-p_i(1-\beta)\right]^n \le 1-\Phi. \tag{9}$$

After applying natural logarithm to the Equation (9) and then divide by $\ln[1-p_i(1-\beta)]$, so the final equation can be expressed as below.

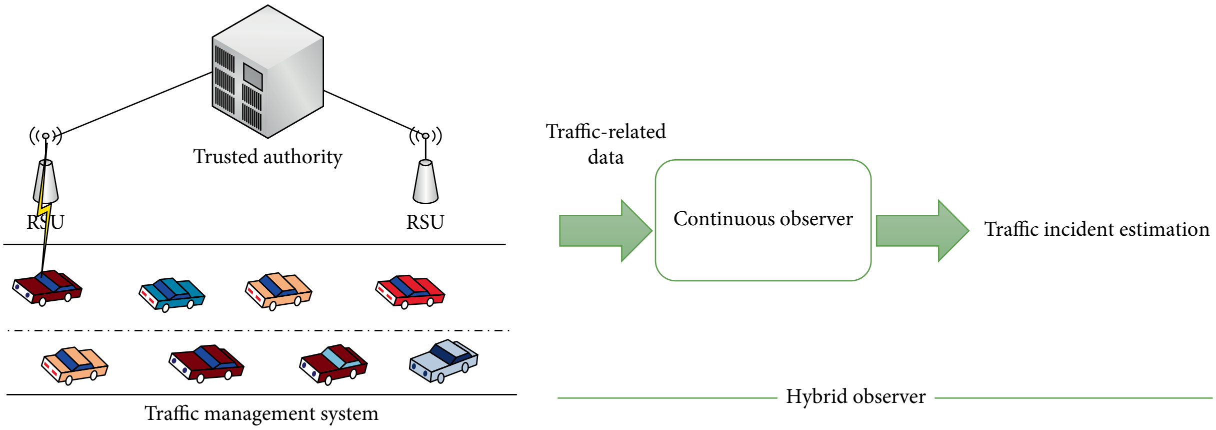


Figure 4: Estimation of traffic incident.

$$n = \left\lceil \frac{\ln(1-\Phi)}{\ln[1-p_i(1-\beta)]} \right\rceil. \tag{10}$$

Equation (10) is used to determine the number of vehicles which is necessary for meaningful aggregation at the event A_n caused by the incident vehicle in the highway traffic environment.

4. Proposed Traffic Incident Estimation and Detection

4.1. Designing of PWSL Hybrid Observer. A designing observer is used to estimate and reconstruct the traffic states of the dedicated system by using measurable variables. As the nature and characteristics of traffic systems are complex, a hybrid observer is able to estimate the possibility of occurring an event that detects the traffic incident. In order to obtain the accurate estimation of traffic incident, a PWSL model and hybrid observer are combined together to produce a better estimation [37].

To estimate the traffic incident, we have developed the structure of the hybrid observer with PWSL, which is written as.

$$\grave{t}(k+1) = M_s\grave{t}(k) + N_s v(k) + D_s + G_s(q_i(k) - \grave{q}_i(k)), \tag{11}$$

where G_s is the observer gain with traffic incident mode $\grave{t}$. G_s is linked with the PWSL to ensure the accurate estimation of traffic incident $\grave{t}(k)$ with the theoretical incident $t(k)$ under any traffic condition. Therefore, the observed gains ensure the convergence of estimated error and stablize the matrices $(M_s - G_s c)$.

Estimated traffic incident $\grave{t}(k)$ of the continuous state converge with the theoretical incident $t(k)$ using the continuous traffic flow $v(k)$, and the continuous output $\grave{q}_i(k)$ can be expressed as below.

$$\grave{q}_i(k) = c\grave{t}(k), \tag{12}$$

where c is the matrix with structures that depends on traffic conditions or state.

The estimation of traffic incident depends on several factors such as traffic scenarios, road conditions, traffic flow, etc. Figure 4 illustrates the estimation of the traffic incident, which rely on the traffic-related data and continuous observer. By using the continuous observer mechanism, we can obtain the accurate estimation of traffic incidents.

4.2. Traffic State Estimation. In most of the estimation approaches, the utilization of Lyapunov function ensures the asymptotic convergence of the estimated error [38]. A multiple Lyapunov function introduced a piecewise Lyapunov function due to the nature of the piecewise hybrid system to ensure the guarantee of error reduction [39].

To solve the HO problem which relies on determining the observer gain G_s, the estimated traffic incident $\grave{t}(k)$ could converge with the theoretical incident $t(k)$. Therefore, the difference of possible error between theoretical and estimated incident can be expressed as below.

$$e_i(k) = t(k) - \grave{t}(k), \tag{13}$$

$$e_i(k+1) = [M_s - G_s c]e_i(k). \tag{14}$$

The convergence of the traffic incident estimation error is required to obtain the gain G_s of the hybrid observer of (11), which ensure that $[M_s - G_s c]$ is a Hurwitz matrix.

4.3. Detection of Traffic Incident. To detect traffic incident based on lane changing speed, RSUs first collect the traffic information related to speed changing between the vehicles and then analyze and evaluate the vehicle speed in the incident and nonincident conditions. In other words, at the nonincident conditions, the average change in speed in shorter average time during changing lane as compared with the incident conditions. In this method, we used the collected traffic information from the RSU related to vehicle speed changing, to evaluate whether or not the incident has occurred when the vehicle changing lane speed falls in the critical region of the defined threshold values.

Figure 5 shows the vehicle lane-changing process which identified that the three vehicles A, B, and C vehicles are travelling along the road. Vehicle A is switching from lane 2 at "point a" to lane 1 at "point b" to pass and cross the vehicle B. The aberrant change in speed from "a" to "b" caused by the driver behavior which creates disturbance to

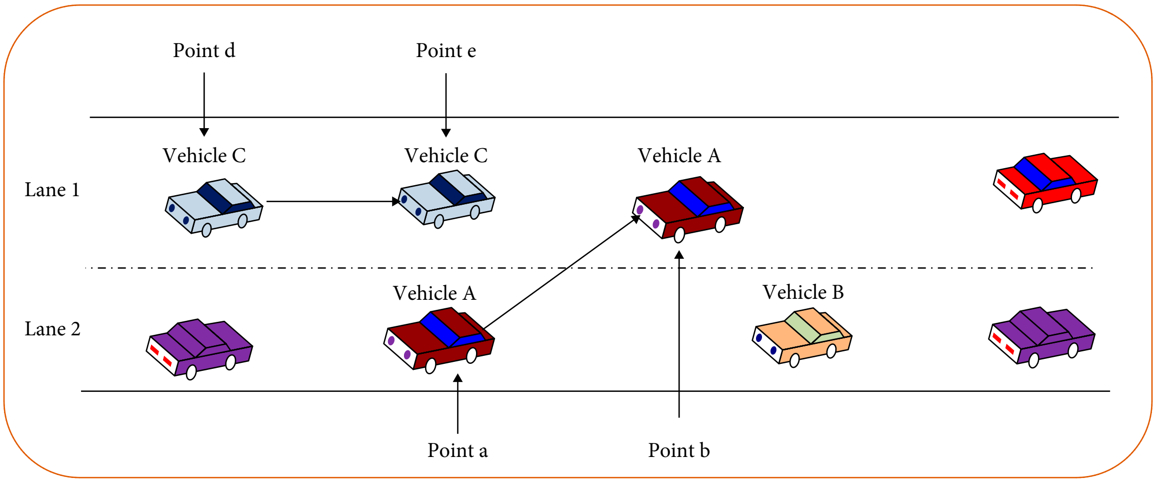


Figure 5: Traffic incident detection based on lane changing speed.

the other vehicles on the road, which may subsequently cause traffic incident. We assumed that the aberrant change in speed while changing lane by the vehicle A also causes variation and disturbance of the vehicle C speed. The RSU calculates the average speed variation γ_{va} which occurred during the lane-changing and the associated average time γ_{ta}. Based on these parameters, we have defined the threshold level, if the speed variation falls under the incident threshold region, then it clearly indicates that an incident has occurred, which is caused by the aberrant overspeeding during lane change.

5. Model Comparison

In ITS, model validation is considered as an important parameter because it is able to evaluate the effectiveness of the presented method. Though, the study and empirical investigation revealed that the detection of traffic incidents is more complicated and challenging than the other traditional incidents due to the nature and characteristics of the traffic incident. These characteristics depend on the traffic structure, pattern, and collection of traffic information data from infrastructure. In the past, many traditional AID techniques have been proposed which investigates the traffic incidents in different traffic scenarios [24, 25]. In order to evaluate the performance of the proposed method, we have compared the results with traditional AID techniques. Also, the presented method is further validated by KM estimate [40], which is used to evaluate the duration and clearance of traffic incidents with other competent methods.

5.1. Incident Threshold Region. Figure 6 illustrates the average change in speed with the average time for lane changing mechanism in incident detection threshold region, i.e., incident and nonincident scenarios. It can be seen that, when a lane is congested due to road bottleneck, the average vehicle speed changes in much shorter average time under nonincident conditions as compared with the incident conditions. In this method, the proposed method uses the collected traffic-related data from the RSU related to vehicle speed changing to evaluate whether or not the incident has occurred when the changing lane speed falls in the critical region of the defined threshold values.

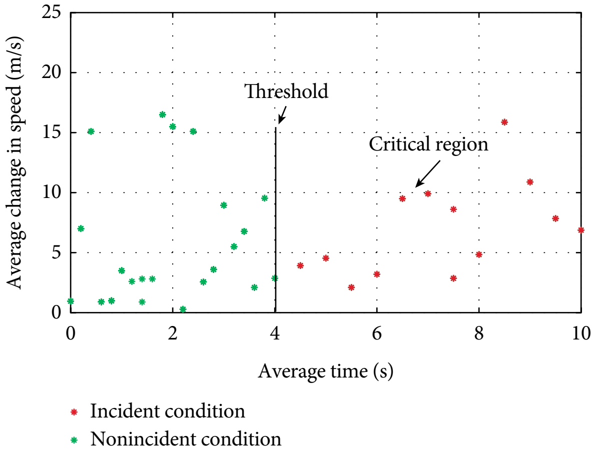
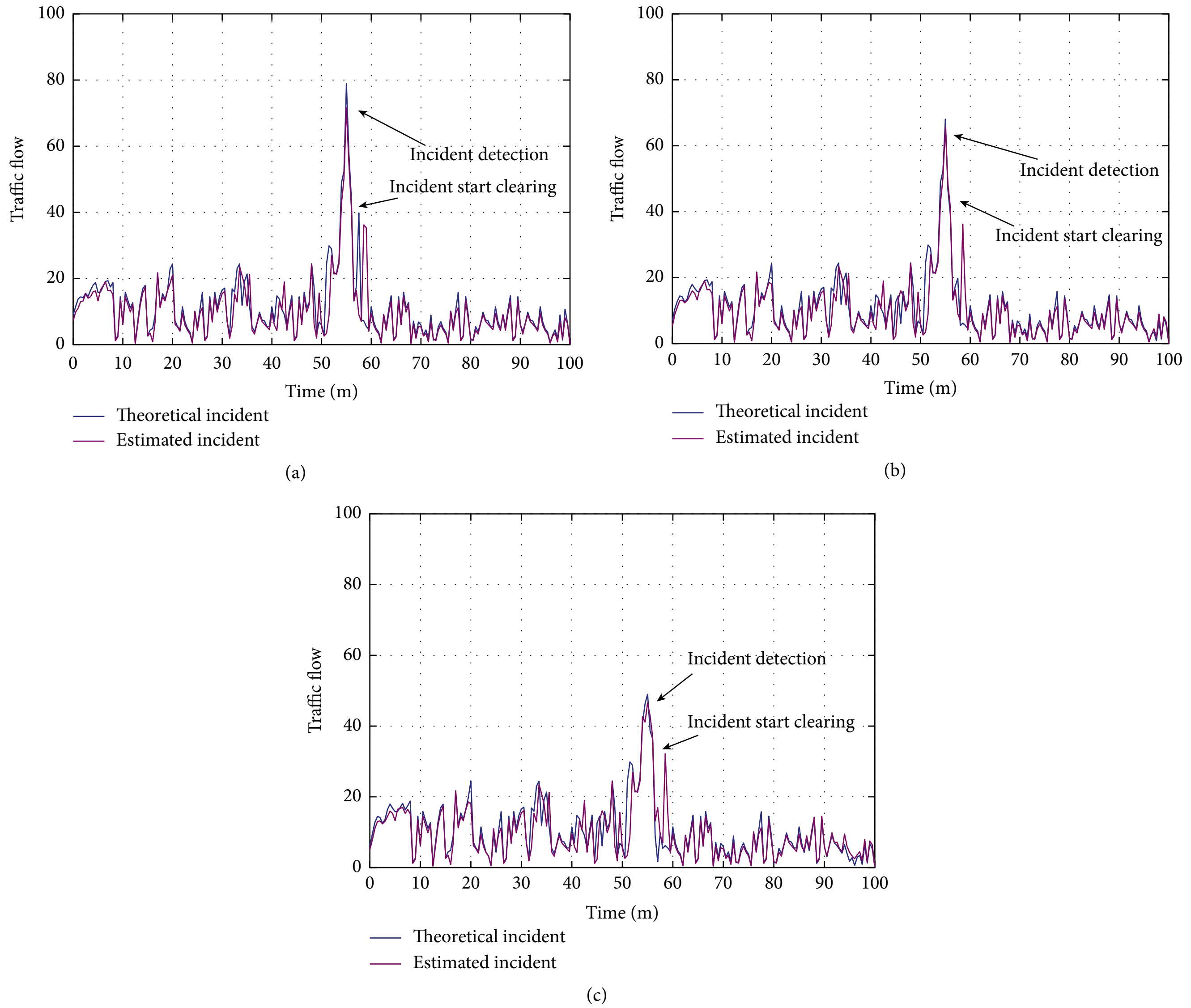


Figure 6: Incident detection threshold region for changing lane speed.

5.2. Estimation of Traffic Incident. Figure 7 shows the comparison of the theoretical incident with the estimated incident. In the simulation, we have considered three cases such as low, moderate and high traffic densities. Figure 7(a) illustrate the comparison of the theoretical and estimated traffic incident with low traffic density. From Figure 7(a), it can be seen that the detection of traffic incident takes a longer time to detect and clear an incident. However, the estimation of traffic incident is close to the theoretical incident, which indicated that the proposed method is able to estimate the traffic incidents in case of low traffic density.

When the traffic density is moderate, the performance of the proposed method in terms of estimation of traffic incidents shown in Figure 7(b). Clearly improved estimation of traffic incident has been obtained. Also, it has the fastest clearance



(a)

(b)

(c)

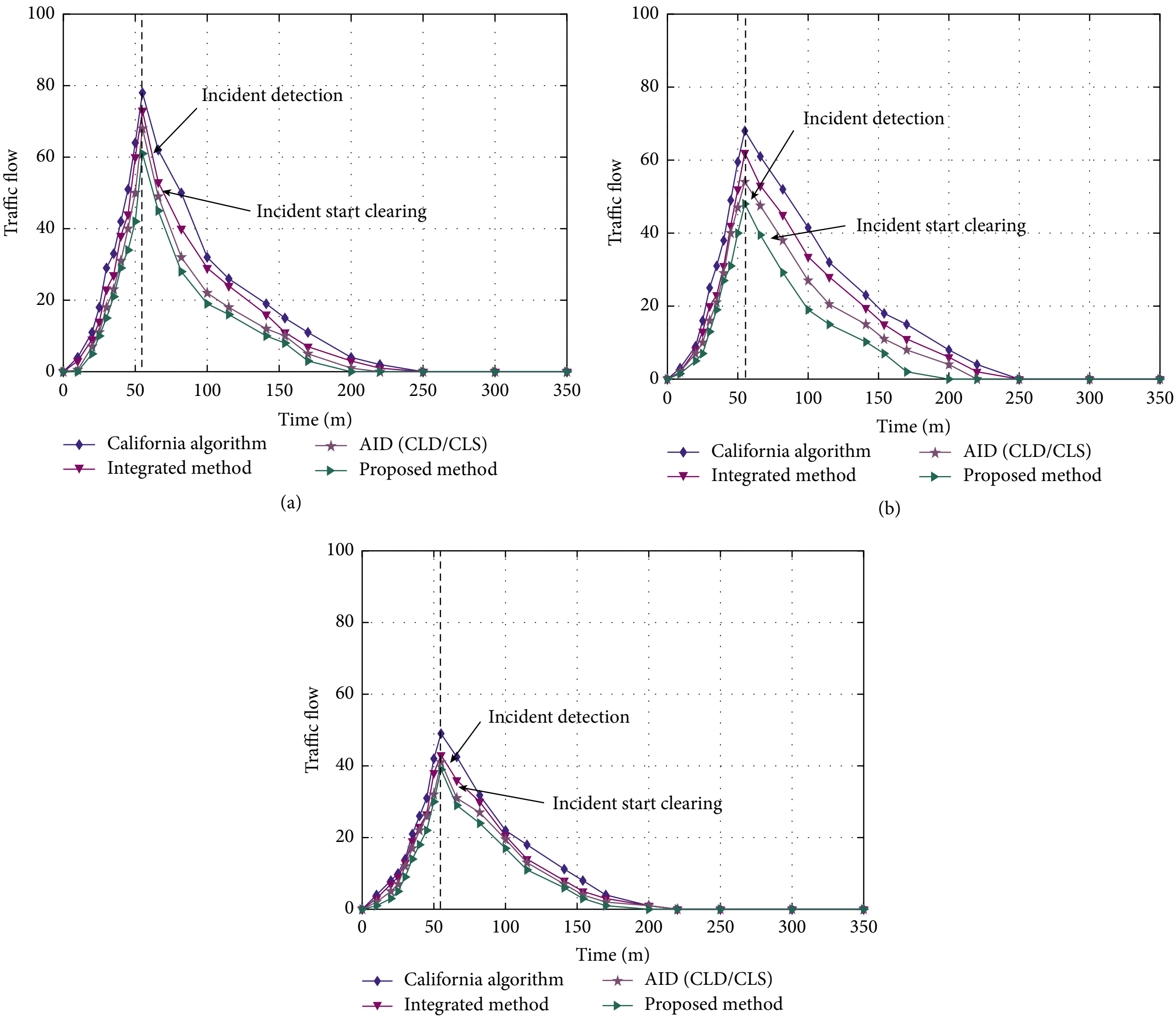
Figure 7: Comparison of the theoretical incident and estimated incident with three different cases. (a) Low traffic flow, (b) moderate traffic flow, and (c) heavy traffic flow.

of incident detection. More specifically, the proposed method obtained a better estimation of traffic incident than the low traffic density, and the estimation of traffic incident has somehow agreed with the theoretical traffic incident.

Figure 7(c) illustrates the comparison of the theoretical incident with the estimated incident when the traffic density is very high. The proposed method obtained the most robust estimation of the traffic incident in comparison with the low traffic and moderated traffic densities. And, the estimated traffic incident is also very close to the theoretical incident. This indicates that the proposed method agrees well with the theoretical incident.

5.3. Traffic Incident Detection. Figure 8 shows the comparison of the proposed method with other methods. In our simulation, we assumed that when the incident occurred, it introduces traffic congestion. The simulation results demonstrated in Figure 8 shows that the traffic congestion is influenced by the traffic incident. More specifically, the traffic congestion depends on the number of vehicles which take alternative routes when the incident has occurred. From Figure 8, it can be observed that the proposed system achieved the fastest detection of the traffic incident. Also, when the incident is cleared by the police, the proposed method obtains the fastest dissipation of traffic congestion.

Figure 8(a) shows the comparison of the proposed method with the heavy traffic congestion of 25% of traffic diverts. It can be seen that the improved AID technique is able to detect the fastest traffic incident as compared with the other methods. Figure 8(b) shows the comparison of the proposed method with 35% of traffic diverts. At the time, when the incident has occurred, 35% of the vehicles take other routes. By means of simulation, it can be observed that the proposed method obtains better incident detection and the fastest dissipation of traffic congestion. Figure 8(c) shows the comparison of the proposed method with 45% of traffic diverts. The proposed



(a)

(b)

(c)

Figure 8: Shows the comparison of the proposed method with California algorithm, integrated algorithm, and AID (CLD/CLS) method with three different cases. (a) Heavy traffic congestion, (b) moderate traffic congestion, and (c) less traffic congestion.

improved AID technique is able to give the fastest incident detection and obtains the fastest dissipation of traffic congestion.

From Figure 9, we can evaluate the performance of the KM curves between the proposed method and other AID techniques. It can be seen that the duration of the incident of California algorithm and integrated method are similar to each other. The characteristics of these algorithms required a longer time to notify the police for the incident situation and also need more time to tow and clear an incident. The KM curve for the incident duration with the AID (CLD/CLS) technique obtained better performance than the Integrated and California methods. After receiving complaint, the police arrived at the incident location in the shortest time to towing the incident vehicles and subsequently clear the incident. By using the proposed method, the police were able to clear the incident in the shortest time by towing away the incident vehicles. The validation results reveal that the proposed method has the capability to estimate and detect the traffic incident with the fastest detection rate. With regard to duration of an incident, it can be seen from Figure 9 that the proposed model obtained a better KM curve by achieving the shortest duration of time to clear an incident among all other schemes.

6. Simulation Results

6.1. Exchange Communication between Vehicle and RSU. In our simulations, we assumed that the average speed of the vehicles passing from the RSU was varying from 20 mph to 80 mph in the highway traffic environment. The probability of exchange traffic information between vehicle and RSU was determined using the low data rates such as 512 kbps and

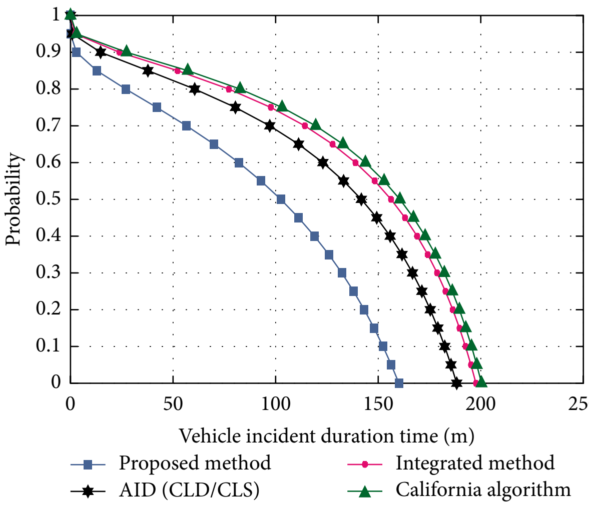


FIGURE 9: Show the comparisons of the KM curves of the proposed method with other AID techniques.

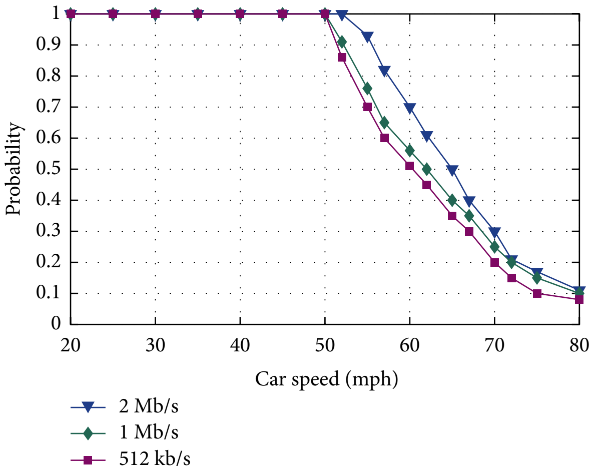


FIGURE 10: Probability of successful exchange traffic information with different data rates.

1 Mbps, and the high data rates such as 2 Mbps. As shown in Figure 10, the results obtained from the simulation revealed that the probability of exchange traffic information decreases with the increase of the average speed of the vehicles passing by the RSU. As the vehicle travels at low speed 50 mph and below, it will remain in the coverage area of the RSU for a longer time and be able to exchange the accurate traffic information at lower data rates such as 512 kbps. When the average vehicle speed exceeds 55 mph, the probability of successful exchange traffic information takes place between the passing vehicles and RSU at higher data rate such as 2 Mbps, which lead to provide the higher probability of exchange traffic information.

6.2. Vehicle Communication with Each RSU Location. The simulation test is carried out to examine the influence of vehicles traveling to each RSU on the probability of successful exchange information as illustrated in Figure 11. We have placed four RSUs with a distance of 1.5 km apart from each other that is able to detect the change in vehicle speed which is influenced by the aberrant overtaking and may lead to causes of the traffic incident. These RSUs are able to obtain the traffic information and violence of each vehicle within their range. Four types of vehicles such as vehicle *A*, vehicle *B*, vehicle *C*, and other vehicles are used in a one-way three-lane traffic scenario. It can be observed that the vehicle *A* overtaking the other vehicle while changing lane at the distance of RSU location about 5 km–10 km. From Figure 11, it can be seen that the probability of traffic information increases as the vehicles are heading towards the next RSU location. Therefore, the strong connection between the vehicle and RSU is developed, which successfully lead to exchange of the traffic information between the passing vehicles and each RSU locations.

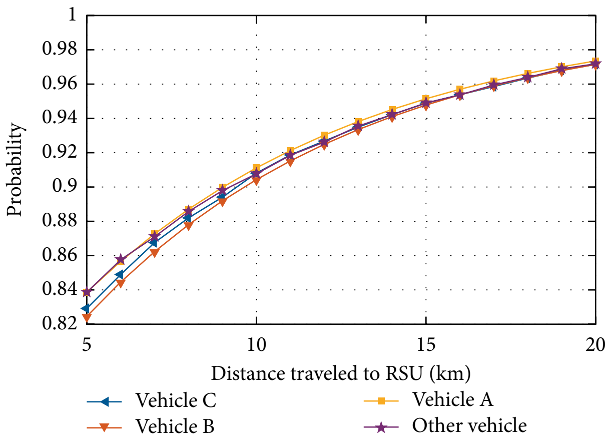


FIGURE 11: Communication between vehicle to each RSU location.

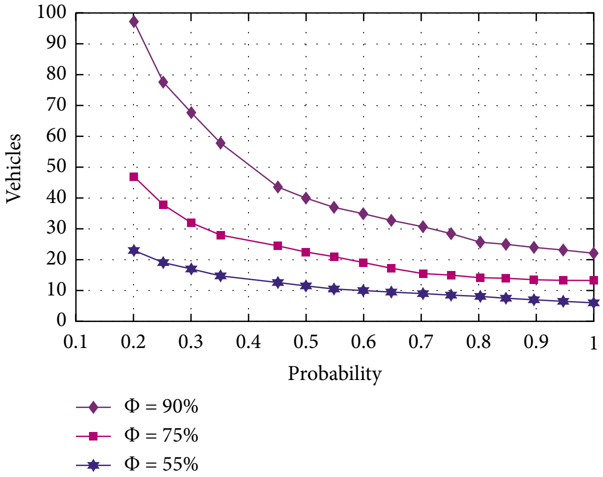
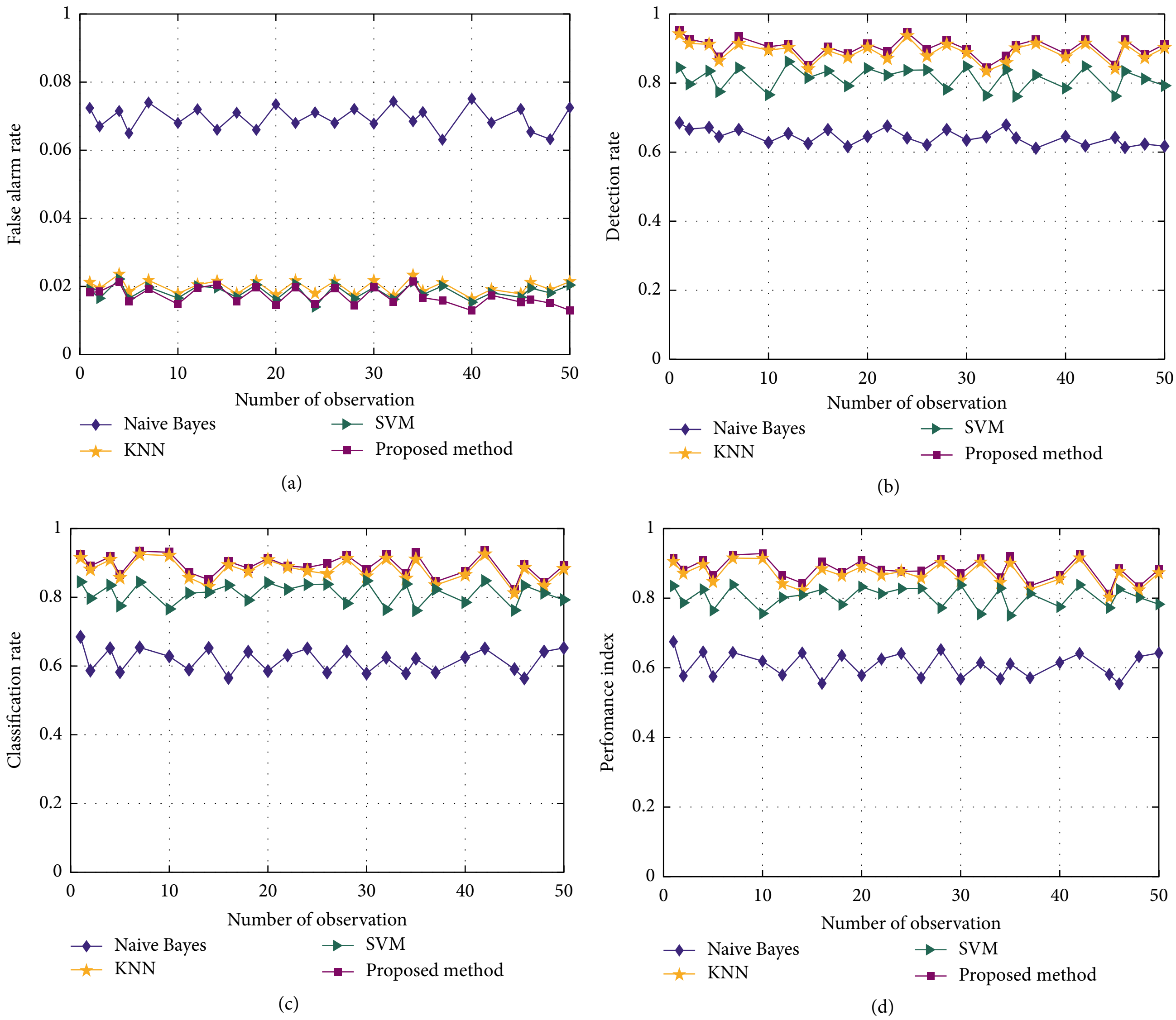


FIGURE 12: Influence of number of vehicles on the probability of traffic information data.

6.3. Probabilistic Comparison of Traffic Information Data. Figure 12 shows the effects of the number of vehicles on the probability of data collection. In order to accurately detect traffic incidents, we have assumed several parameters, the incident target $\Phi = 0.9$, application parameter which is able to detect an incident accurately $\beta = 0.90$, and the probability of



(a)

(b)

(c)

(d)

FIGURE 13: Comparisons of proposed method with other competent methods (a) FAR, (b) DR, (c) CR, and (d) PI.

a vehicle to communicate with the RSU is $p_i = 0.7$. Substitute all these values in equation (10), after solving we obtained that 32 vehicles can communicate with RSU, with the probability of 90%. Therefore, it indicates that RSU is able to acquire the traffic information data of 32 vehicles including incident vehicle, and it is also used to detect the traffic incidents caused by vehicle A with the higher accuracy of $\beta = 0.90$.

6.4. Performance Test Criteria. In this section, we evaluate the performance of the proposed method with other well-known techniques such as Naive Bayes, SVM, and KNN using three criteria i.e. false alarm rate (FAR), detection rate (DR) and classification rate (CR) [41].

$$\text{FAR} = \frac{F_n}{I_{ni}} \times 100\%, \tag{15}$$

where FAR is the false alarm rate, F_n is the number of false alarm cases, and I_{ni} is the total number of nonincident cases.

$$\text{DR} = \frac{I_d}{A_i} \times 100\%, \tag{16}$$

where DR is the incident detection rate, I_d is the number of incident cases detected and A_i is the total number of incident cases reported.

$$\text{CR} = \frac{T_i}{C_i} \times 100\%, \tag{17}$$

where CR is the classification rate, which is used to determine the incident detection, T_i is the number of events correctly classified and C_i is the total number of events.

We further evaluate the performance of the proposed method by using performance index (PI), which can be written as below.

$$\text{PI} = w_d\text{DR} + w_f(1 - \text{FAR}) + w_c\text{CR}, \tag{18}$$

where w_d, w_f, and w_c are the weight of the DR, FAR, and CR, respectively. We assumed that the values of the weight of DR,

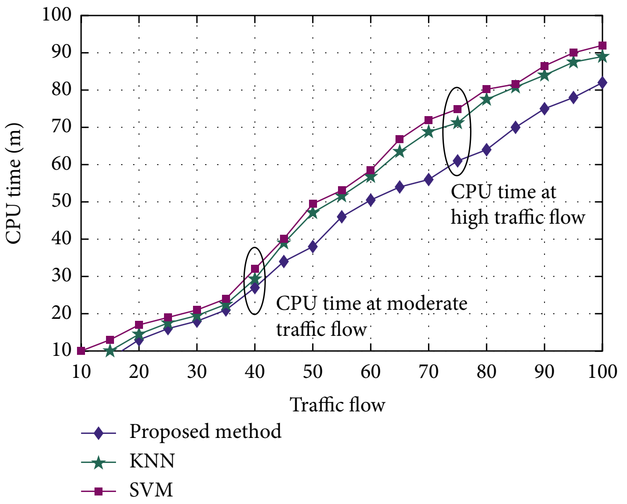


Figure 14: CPU comparison of the proposed method with other methods.

FAR, and CR are all 1/2. The larger value of PI indicates that the proposed method has obtained a better detection of traffic incidents. More specifically, the smaller the FAR values, the higher the possibility of detecting traffic incident accurately. The performance of DR indicates that, when the DR values approaching 100%. It clearly indicates that the algorithm is able to detect the traffic incident well. However, the higher DR values may generate some false alarm.

To evaluate the effectiveness of the proposed method, we further demonstrate the performance of the proposed method on I-880 datasets for evaluating FAR, DR, CR, and PI values, and compare these values with the other methods. Figure 13 shows the comparison of the proposed method with other competent methods in terms of FAR, DR, CR, and PI values. From Figure 13(a) it can be observed that the proposed method has obtained less FAR values as compared with the SVM method. Also, we can observe that the Naive Bayes and KNN methods produce worst FAR values as compared with the SVM method.

As shown in Figure 13(b), the proposed method has obtained the highest DR values which indicate that the accurate detection of traffic incident. Also, it can be seen that the KNN method has obtained better DR values to accurately detect the traffic incident as compared with the Naives Bayes and SVM methods. Due to the nature and characteristics of incident, the Naive Bayes and SVM methods are unable to detect traffic incident.

It can be noted from Figure 13(b) and Figure 13(c), the results of KNN method in DR and CR is very close to the proposed method. Also, from Figure 13(a) the FAR values of SVM are very close to the proposed method. Also from Figure 13(d), it can be observed that the proposed method has obtained better PI values as compared with other competent methods. Therefore, it clearly demonstrates that the improved values of FAR, DR, CR, and PI enhanced the performance of the proposed method, thus, the proposed method has an ability to detect traffic incident on I-880 datasets.

More specifically, from Figure 13, the observation values minimized FAR values when DR value was greater than 0.90. From the observation value, the proposed method obtained the FAR, DR, CR, and PI values of 0.018, 0.952, 0.925, and 0.915, respectively as compared with the KNN 0.1912, 0.937, 0.909, and 0.905, respectively. Clearly, the improved values indicate that the proposed method has obtained the better FAR, DR, CR, and PI values as compared with the other well-known techniques such as Naive Bayes, KNN, and SVM methods.

6.5. CPU Timing. Figure 14 shows the CPU comparison of the proposed method with KNN and SVM methods. When the traffic flow is 10, the CPU time of the proposed method and other methods were very low. As the traffic flow increases, the CPU required a longer time to process input dataset. From Figure 14, it can be observed that the KNN and SVM methods required a longer CPU time due to exploiting more input data, which subsequently required a longer time to process dataset. The proposed method obtains the less CPU time to process input dataset as compared with other competent methods.

7. Conclusion

The incident estimation and detection are the important parameters in the ITS for reducing the traffic congestion, improving traffic and road safety. However, the traffic incident may cause road bottlenecks, traffic congestion and disrupt the normal traffic flow. In this paper, we proposed an ITS model which is able to estimate the traffic incident from HO method, and then detect the traffic incident from an improved AID technique in the highway traffic environment. First, we present a hybrid observer method to estimate the traffic incident based on the combination of the PWSL model and a model estimation technique. Second, we designed a probabilistic approach to collect traffic information data by using a V2I communication based on lane-changing speed mechanism capable of detecting traffic incident accurately. The analysis shows that the proposed method can estimate the traffic incident more precisely and has the potential to estimate the traffic incident with three different traffic densities. By mean of the simulation, the proposed method has obtained a better estimation of the traffic incident, which agrees well with the theoretical incident. Furthermore, by comparing the effectiveness of the proposed method for detecting traffic incident, the proposed method outperforms other methods by obtaining the fastest incident detection rate, and the fastest dissipation of traffic congestion. The comparison with the other methods in terms of duration of an incident, the proposed method obtains the shortest duration of time to clear an incident. Moreover, we have further evaluated the performance of the proposed method with well-known techniques such as Naive Bayes, KNN, and SVM using I-880 traffic data. Experimental results show that the proposed method obtained a better performance compared with other methods.

In the future, we will enhance the proposed model by using joint learning and support vector machine (SVM) techniques which can produce the robust detection of traffic incidents. Also, we will utilize the 5G wireless network to obtain the robust accuracy of traffic information data which will be successfully applied to estimate the traffic incident.

Data Availability

The performance test data used to support the finding of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

This work was supported in part by National key research and development program under Grant 2018YFB1600503, in part by the Natural Science Foundation of China under Grants U1564201, 1664258, and 61773184.

References

[1] G. Anastasi, M. Antonelli, A. Bechini et al., "Urban and social sensing for sustainable mobility in smart cities," in *2013 Sustainable Internet and ICT for Sustainability (SustainIT)*, pp. 1–4, IEEE, Palermo, Italy, 2013.

[2] M. N. Mejri, J. Ben-othman, and M. Hamdi, "Survey on VANET security challenges and possible cryptographic solutions," *Vehicular Communications*, vol. 1, no. 2, pp. 53–66, 2014.

[3] J. Luk, C. Han, and D. A. Chin, "Freeway incident detection: technologies and techniques," 2010.

[4] F. Yuan and R. L. Cheu, "Incident detection using support vector machines," *Transportation Research Part C: Emerging Technologies*, vol. 11, no. 3-4, pp. 309–328, 2003.

[5] E. D'Andrea and F. Marcelloni, "Detection of traffic congestion and incidents from GPS trace analysis," *Expert Systems with Applications*, vol. 73, pp. 43–56, 2017.

[6] S. Kamran and O. Haas, "A multilevel traffic incidents detection approach: identifying traffic patterns and vehicle behaviours using real-time GPS data," in *2007 IEEE Intelligent Vehicles Symposium*, pp. 912–917, IEEE, Istanbul, Turkey, 2007.

[7] L. L. Wang, H. Y. T. Ngan, and N. H. C. Yung, "Automatic incident classification for large-scale traffic data by adaptive boosting SVM," *Information Sciences*, vol. 467, pp. 59–73, 2018.

[8] V. Milanes, J. Villagra, J. Godoy, J. Simo, J. Perez, and E. Onieva, "An intelligent V2I-based traffic management system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 49–58, 2012.

[9] J.-M. Chung, M. Kim, Y.-S. Park, M. Choi, S. Lee, and H. S. Oh, "Time coordinated V2I communications and handover for WAVE networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 3, pp. 545–558, 2011.

[10] Y. Asakura, T. Kusakabe, L. X. Nguyen, and T. Ushiki, "Incident detection methods using probe vehicles with on-board GPS equipment," *Transportation Research Part C: Emerging Technologies*, vol. 81, pp. 330–341, 2017.

[11] J. Ren, Y. Chen, L. Xin, J. Shi, B. Li, and Y. Liu, "Detecting and positioning of traffic incidents via video-based analysis of traffic states in a road segment," *IET Intelligent Transport Systems*, vol. 10, no. 6, pp. 428–437, 2016.

[12] D. Srinivasan, X. Jin, and R. L. Cheu, "Evaluation of adaptive neural network models for freeway incident detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 1, pp. 1–11, 2004.

[13] S. G. Ritchie and R. L. Cheu, "Simulation of freeway incident detection using artificial neural networks," *Transportation Research Part C: Emerging Technologies*, vol. 1, no. 3, pp. 203–217, 1993.

[14] J. Wang, X. Li, S. S. Liao, and Z. Hua, "A hybrid approach for automatic incident detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1176–1185, 2013.

[15] X. Jin, R. L. Cheu, and D. Srinivasan, "Development and adaptation of constructive probabilistic neural network in freeway incident detection," *Transportation Research Part C: Emerging Technologies*, vol. 10, no. 2, pp. 121–147, 2002.

[16] Z. Zhou and L. Zhou, "An automatic incident of freeway detection algorithm based on support vector machine," in *2010 International Symposium on Intelligence Information Processing and Trusted Computing*, pp. 543–546, IEEE, Huanggang, China, 2010.

[17] R. L. Cheu, D. Srinivasan, and E. T. Teh, "Support vector machine models for freeway incident detection," in *Proceedings of the 2003 IEEE International Conference on Intelligent Transportation Systems*, pp. 238–243, IEEE, Shanghai, China, 2003.

[18] J. Xiao, "SVM and KNN ensemble learning for traffic incident detection," *Physica A: Statistical Mechanics and its Applications*, vol. 517, pp. 29–35, 2019.

[19] Y. Gu, Z. Qian, and F. Chen, "From Twitter to detector: Real-time traffic incident detection using social media data," *Transportation Research Part C: Emerging Technologies*, vol. 67, pp. 321–342, 2016.

[20] A. Schulz, P. Ristoski, and H. Paulheim, "I see a car crash: real-time detection of small scale incidents in microblogs," *The Semantic Web: ESWC 2013 Satellite Events*, pp. 22–33, Springer, 2013.

[21] S. Dabiri and K. Heaslip, "Developing a Twitter-based traffic event detection model using deep learning architectures," *Expert Systems with Applications*, vol. 118, pp. 425–439, 2019.

[22] Z. Zhang, Q. He, J. Gao, and M. Ni, "A deep learning approach for detecting traffic accidents from social media data," *Transportation Research Part C: Emerging Technologies*, vol. 86, pp. 580–596, 2018.

[23] J. D. G. Paule, Y. Sun, and Y. Moshfeghi, "On fine-grained geolocalisation of tweets and real-time traffic incident detection," *Information Processing & Management*, vol. 56, no. 3, pp. 1119–1132, 2019.

[24] C. K. Lin and G. L. Chang, "Development of a fuzzy-expert system for incident detection and classification," *Mathematical and Computer Modelling*, vol. 27, no. 9–11, pp. 9–25, 1998.

[25] R. Weil, J. Wootton, and A. Garcia-Ortiz, "Traffic incident detection: sensors and algorithms," *Mathematical and Computer Modelling*, vol. 27, no. 9–11, pp. 257–291, 1998.

[26] M. Abuelela, S. Olariu, M. Cetin, and D. Rawat, "Enhancing automatic incident detection using vehicular communications," in *2009 IEEE 70th Vehicular Technology Conference Fall*, pp. 1–5, IEEE, Anchorage, AK, USA, 2009.

[27] Q. He, Y. Kamarianakis, K. Jintanakul, and L. Wynter, "Incident duration prediction with hybrid tree-based quantile regression," *IBM Research*, 2011.

[28] S. Peeta, J. L. Ramos, and S. Gedela, "Providing real-time traffic advisory and route guidance to manage Borman incidents on-line using the Hoosier helper program," Joint Transportation Research Program, Indiana Department of Transportation and Purdue University, West Lafayette, Indiana, 2000.

[29] J. Lu, S. Chen, W. Wang, and B. Ran, "Automatic traffic incident detection based on nFOIL," *Expert Systems with Applications*, vol. 39, no. 7, pp. 6547–6556, 2012.

[30] R. Wang, S. Fan, and D. B. Work, "Efficient multiple model particle filtering for joint traffic state estimation and incident detection," *Transportation Research Part C: Emerging Technologies*, vol. 71, pp. 521–537, 2016.

[31] M. Fogu, P. Garrido, F. J. Martinez, J. C. Cano, C. T. Calafate, and P. Manzoni, "Automatic accident detection: assistance through communication technologies and vehicles," *IEEE Vehicular Technology Magazine*, vol. 7, no. 3, pp. 90–100, 2012.

[32] S. Tang and H. Gao, "Traffic-incident detection-algorithm based on nonparametric regression," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 38–42, 2005.

[33] O. Popescu, S. Sha-Mohammad, H. Abdel-Wahab, D. C. Popescu, and S. El-Tawab, "Automatic incident detection in intelligent transportation systems using aggregation of traffic parameters collected through V2I communications," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 2, pp. 64–75, 2017.

[34] D. Sapper, H. Cusack, and L. Staes, "Evaluation of electronic data recorders for incident investigation, driver performance, and vehicle maintenance," National Center for Transit Research (NCTR) Center for Urban Transportation Research, University of South Florida, 2009.

[35] M. Özkul and I. Çapuni, "Police-less multi-party traffic violation detection and reporting system with privacy preservation," *IET Intelligent Transport Systems*, vol. 12, no. 5, pp. 351–358, 2018.

[36] S. El-Tawab, A. Alhafdhi, D. Treeumnuk, D. C. Popescu, and S. Olariu, "Physical layer aspects of information exchange in the NOTICE architecture," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 8–18, 2015.

[37] A. Zeroual, F. Harrou, and Y. Sun, "Road traffic density estimation and congestion detection with a hybrid observer-based strategy," *Sustainable Cities and Society*, vol. 46, p. 101411, 2019.

[38] W. Heemels and S. Weiland, "Observer design for a class of piecewise linear systems," *International Journal of Robust and Nonlinear Control*, vol. 17, no. 15, pp. 1387–1404, 2007.

[39] H. Lin and P. J. Antsaklis, "Stability and stabilizability of switched linear systems: a survey of recent results," *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 308–322, 2009.

[40] E. L. Kaplan and P. Meier, "Nonparametric estimation from incomplete observations," *Journal of the American Statistical Association*, vol. 53, no. 282, pp. 457–481, 1958.

[41] J. Xiao and Y. Liu, "Traffic incident detection using multiple-kernel support vector machine," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2324, no. 1, pp. 44–52, 2012.