Zero-to-Stable Driver Identification: A Non-Intrusive and Scalable Driver Identification Scheme

Mussadiq Abdul Rahim, Liehuang Zhu, Xin Li, Jiamou Liu, Zijian Zhang, Zhan Qin, Salabat Khan, and Keke Gai

***Abstract*—Driver identification faces various challenges in real-time applications. These challenges include high dimensional input data, moderate accuracy, scalability issues and need for custom-built in-vehicle sensors. The conventional biometric solutions are either less accurate, unscalable, or costly for practical application. In those solutions, high accuracy comes at the cost of the preservation of privacy. Similarly, scalability and cost-effectiveness are inversely proportional to one another. This paper proposes a driver identification scheme to pare these challenges. The scheme uses data from the global positioning system (GPS) to learn an individual's driving pattern, which is commonly deployed in in-car navigation systems and can also be found in general-purpose hand-held devices in the prevailing market. The proposed scheme is innovative in terms of providing a single solution for the existing challenges. It is more practical and scalable with large numbers of the drivers also being cost-effective and accurate, which makes it applicable in real-time applications with the least overhead costs for different resources. The consequent analysis and empirical results show that the scheme could identify drivers with significant accuracy given only GPS data. To assess the scheme in-depth, we perform experiments both on the collected data and the real-world open datasets. The average accuracy approximates above 96% for up to 25 drivers.**

***Index Terms*—Driver identification, driver classification, machine learning, applied artificial intelligence.**

I. Introduction

IN RECENT times, driver behavior modeling and driver identification have evolved substantially. Driver identification problem is defined as classification or identification of a single driver out of a set of drivers given some knowledge base of drivers, and certain input for a single driver. In modern research, the underlying knowledge base differs from solution to solution. Different biometrics and driver identification solutions work on different kinds of data – where the data is something a driver knows, has or is. In the context of this research, the focus is on behavioral data, which qualifies as what a driver is. Behavioral data can come through different kinds of sensors, where a combined set of sensory data is processed to derive or quantify some driving behavior of a person. Multiple works are available in the research community with the core interest in solving the driver identification problem by learning drivers' skill or behavior. Hallac *et al.* [1] used in-vehicle sensors, Yang *et al.* [2] used wearable sensors, and Martinez *et al.* [3] used multiple kinds of sensors available in the car. Aforementioned and other solutions followed different approaches to solve the driver identification problem by using different sources of information. However existing works suffer from different challenges, especially there exists trade-off in the triad of practical application, scalability, and accuracy. We discuss the features, pros, and cons of the major related works in the following section. Moreira-Matias and Farah [4] studied the strengths and weaknesses of different standard biometrics referred to as in-vehicle data recorders. A recent review by Chowdhury *et al.* [5] showed the need for a solution with improved accuracy and least data dependencies consequently more cost-effective.

The need for cost-effective, non-intrusive and scalable (in terms of the number of drivers) solution is observed. Leads to research questions raised as – Can driver identification be more compact, accurate and scalable? If so, what is an effective approach which has minimum data dependencies? Will such compact scheme be accurate enough to be applicable for large-scale implementation? To conquer these questions in this research, we propose *Zero-to-Stable* (ZTS) *identification* scheme, which requires only GPS data and outperforms the existing techniques on the perspective of accuracy in the practical scenarios with a large number of drivers. The main contribution of our work is many-fold:

- We present a novel event-driven driver identification scheme, with only data dependency on GPS.
- The proposed scheme uses the least number of data features derived from available GPS data.
- We validate the scheme with a detailed set of experiments over different machine learning methods and analyze the performance of ZTS $identification$ against these methods.

Manuscript received June 23, 2019; revised September 12, 2019; accepted November 5, 2019. Date of publication November 25, 2019; date of current version January 15, 2020. This work was partially supported by in part by China National Key Research and Development Program 2016YFB0800301 and in part by the National Natural Science Foundation of China "NSFC" under Grants 61300177. The review of this article was coordinated by Dr. Z. Ma. *(Corresponding authors: Liehuang Zhu; Zijian Zhang.)*

M. A. Rahim, L. Zhu, X. Li, S. Khan, and K. Gai are with the School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100811, China (e-mail: mussadiq.ar@gmail.com; liehuangz@bit.edu.cn; xinli@bit.edu.cn; salabatwazir@gmail.com; gaikeke@bit.edu.cn).

J. Liu is with the Department of Computer Science, The University of Auckland, Auckland 1010, New Zealand (e-mail: jiamou.liu@auckland.ac.nz).

Z. Zhang is with the School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100811, China, and is with the Department of Computer Science at The University of Auckland, Auckland 1010, New Zealand (e-mail: zhangzijian@bit.edu.cn).

Z. Qin is with the Institute of Cyberspace Research, Zhejiang University, Zhejiang 310027, China (e-mail: zhan.qin@utsa.edu).

Digital Object Identifier 10.1109/TVT.2019.2954529

- We evaluate our scheme, over open-access GPS data set and furthermore carry out field work on public buses, by collecting real-time data using the indigenously built mobile app.

We further remark that our work is novel and significant in the following senses: *a)* Our discovery is remarkable especially given the compact-sized feature definition, and the fact that they can be easily captured from a simple GPS receiver available in smart-phone or in-car navigation system. *b)* It is observed that the scalability is neither essentially discussed nor focused in prior research. Since we validate our scheme for a large number of drivers where empirically it yields feasible accuracy, so this scheme shall provide as scalability criterion in future driver identification schemes. *c)* The proposed scheme is applicable to real-time scenarios with high accuracy at the minimum cost of hardware and data processing. The accuracy and cost-effectiveness along with least dependency on data sources are salient features of the proposed scheme.

This research work is oriented in such a manner that it answers the aforementioned research questions while keeping the real-time constraints under consideration. Such constraints are quite crucial in nature for application where the data is limited to GPS only or availability of other data sources is hurdled by budget or intrusiveness nature of the data. The compactness of the proposed feature vector is the essential need and base of the solution to such problems.

Paper organization: Section II summarizes existing approaches on driver identification and related problems. Section III discusses basic preliminary details of supervised machine learning algorithms used in this research. Section IV specifies ZTS $identification$ in detail with features, extraction of feature and core identification scheme. Section V discusses the experiments, analyzes empirical results and provides a comparison to prior methods. Section VI concludes this research article and gives a highlight to future work.

II. Related Works

The study of human driving behaviors and automated driver behavior modeling are well-established topics of research [6], [7]. In previous decade interests emerged related to safety and privacy aspects of smart vehicles [8]. These interests further lead the research on automatic driver identification, initial studies focused on biometrics-based methods such as facial recognition [9]. Riener *et al.* proposed driver identification based on drivers' sitting posture [10]. This required use of custom sensor arrays placed under the driver seat, which led to many constraints and extra costs. Wu and Ye [11] used artificial neural network to identify a driver's voice. However, this approach comes with the constraint of the driver speaking while in the car. Later on, Wu and Ye also proposed [12] driver identification using finger-vein pattern. This biometric method is applicable to scenarios such as when the driver opens the car door, but not applicable when the car is on the road. Usually, such biometric approaches for driver identification suffer from drawbacks of high costs, physical restraints of sensors, and interferences.

Miyajima *et al.* [13] proposed one of the first works that incorporate driver habits into identification. In this research, they considered features such as pedal movements; they used a statistical prediction model of driver actions for driver identification. Their experiments involved data collected from a specially equipped car with up to 24 sensors of different kinds including video, voice, driving signal channels and multiple GPS measurements. Martinez *et al.* [3] proposed the use of extreme learning machine (ELM) and the nearest centroid classifier for driver identification based on driver behavior signals, which yielded about 75% accuracy for eleven drivers. Subsequently, Martinez *et al.* [14] applied ELM over fourteen different signals collected from (Controller Area Network) CAN-bus and other sources and achieved 85% accuracy for driver identification among a set of five drivers. However, the use of CAN-bus means that the required input measurements are not general in the application for all vehicles.

Hallac *et al.* [1] used 12 different data features collected from in-car sensors for driver identification. This method trained the Random Forest Classifier (RFC) using the turn data of the vehicle. This method yielded an accuracy of 76.9% for two drivers, whereas the accuracy dropped down to about 50% for five drivers. Consequently, this method is inapplicable for a large number of drivers. Zhang *et al.* [15] collected data from seven in-car sensors and three smart-phone sensors, tested with up to fourteen drivers. In naturalistic driving scenarios among two drivers, the average accuracy for this method was observed 80%, with variation in results for different kinds of cars. Burton *et al.* [16] identified driver based on driver behavior using SVM, they analyzed 12 features for ten drivers derived from data collected using OpenDS simulator. Equal error rate (EER) with SVM for different drivers was 24.9%, whereas the time for detection was 180 seconds. Yang *et al.* [2] used wearable technology in driver identification using Gaussian mixture model which results in the best accuracy of 70%. As the system requires drivers to use wearable sensors to monitor 18 different readings, it is intrusive and less practical in the real world.

Markwood and Liu [17] used two mobile sensors and GPS to collect data for 31 drivers. Trials were performed on the two-driver pair at a time, combining five different kinds of features. The accuracy of these individual trials varies between 50% to 97%. We point out that our proposed scheme depends on GPS data only and applicable to a large number of drivers in the real world. Chowdhury *et al.* [5] surveyed different techniques for GPS based driver identification. They proposed the use of only GPS data with RFC with a set of 137 statistical derived features from raw GPS data. With this method, the classification accuracy for four drivers varied from 57.7% to 91.4%. Ezzini *et al.* [18] proposed driver classification model, they derived 40 different features to classify drivers. They used various classifiers including SVM, kNN, Random Forest and, multi-layer perception. They selected different features from the feature set for different underlying sources of data, with no more than ten drivers. Qiao *et al.* [19] proposed a framework to analyze the user's mobility in highly populated area the framework uses mobile big data from cellular networks to form trajectories of the user.

To remark the fact that GPS based applications are growing in number and the accuracy of the GPS is often discussed in the research community. Researchers have studied the influence of weather conditions, also topological or terrain conditions on GPS data. Street canyon effect is one such problem which hinders the measurement of accurate GPS data. However recent literature has addressed this problem in detail. Cui and Ge [20] studied the effects of the urban canyon effect and has proposed that these inaccuracies can be mitigated through established solutions such as Kalman-Filter. Cho *et al.* [21] suggest that for specific scenarios where precision is most important, filters can be designed to cope with the errors.

Finally, we remark that smart driving and security-related technologies constitute a large body of work. There are numerous works which apply machine learning methods in different vehicle related applications not limited to driver identification. Zhang *et al.* [22] proposed a Bayesian network model based sensor fault detection scheme to detect attacks on body sensor network to avoid misdiagnosis consequently mistreatment. Pattern recognition and human-machine systems works such as Ma *et al.* [23] proposed end-to-end speech language identification using deep neural networks and long short term memory (LSTM) approach for conversations uttered in intelligent vehicles. Subsequently, Ma *et al.* [24] proposed channel max pooling for convolutional neural networks (CNNs) to identify vehicles, using this method CNNs tend to learn discriminative features in detail. Tang *et al.* [25] proposed Anti-Coordination game-based Partially Overlapping Channels Assignment algorithm to allow unmanned aerial vehicles (UAV) build an effective wireless network in disaster affected areas. Takaishi *et al.* [26] proposed resource allocation scheme for UAVs by creating virtual cells. Imani *et al.* [27] proposed a Bayesian decision making framework for control of Markov Decision Processes with unknown dynamics and large, continuous, state, action, and parameter spaces in data-poor environments. Subsequently, Imani *et al.* [28] proposed a multi-fidelity Bayesian optimization algorithm for the inference of general nonlinear state-space models, this method enabled simultaneous sequential selection of parameters and approximators. It led to the fast and accurate inference of parameters of nonlinear state-space models. Zhang *et al.* [29], [30] proposed deep learning approaches for image classification. Zhang *et al.* [31] proposed a LSTM based approach facial age of a person. Yin *et al.* [32] utilized deep learning to predict Quality of Service (QoS) in edge computing environments. Zeng *et al.* [33] proposed efficient anonymous user authentication protocol for IoT computing users and servers. Zhu *et al.* [34] proposed anonymous smart-parking and online payment system for users searching for parking slot. Li *et al.* [35] proposed use of private block-chain to store car-pooling for high efficiency and privacy preserving. Xu *et al.* [36] proposed a new mutual authentication key agreement protocol for global mobile networks. Liu *et al.* [37] discussed contemporary technologies for used in security of the intelligent vehicles. Wang *et al.* [38] surveyed networking and communication technologies used in autonomous vehicles.

For a general introduction of the use of GPS and machine learning in intelligent transportation systems, the reader may refer to other works such as [39].

III. Preliminary

Random Forest Classifier (RFC): This supervised classification method utilizes classification trees to grow a random forest. Multiple classification trees are combined to form a random forest, where each classification tree provides single classification also referred to as a vote, these votes decide the prediction for any given input. There is no limit to the size of a single tree also this algorithm does not prune the tree. For any two trees within a forest, the more correlation will lead to increased forest error rate. Similarly, the high strength of one tree would decrease forest error rate and will form a better classifier. Classification And Regression Tree (CART) method is well discussed [40] in the literature.

k-Nearest Neighbors (kNN): This classification algorithm belongs to family nearest neighbors [41] algorithms; it is a non-parametric classification algorithm. Training of this algorithm is performed by storing feature vectors for given classes. For a feature space, the consensus of its k nearest neighbors classifies an input. Nearest neighbors are defined based on some distance metric, typically Euclidean distance metric is used but other distance metrics like Manhattan, Minkowski or Hamming distance can be used for this purpose.

Support vector machine (SVM): SVM is one of supervised machine learning algorithm, fundamentally it is a binary classifier. For $\mathbf{N} = \{x_1, \ldots, x_t\}$ training set, given t number of samples. The vector $\vec{w} = (w_1, w_2, w_3, w_4)$ is the normal vector of the target hyperplane. $\phi(x)$ defines kernel function used, b is the margin distance which the classifier aims to ensure between support vectors. For k classes LIBSVM [42] defines $\frac{k*(k-1)}{2}$ classifier problems each classifier optimizes for i and j class as following subject to constraints for two classes

$$\min_{w^{ij}, b^{ij}, \xi^{ij}} \quad \frac{1}{2}(w^{ij})^T w^{ij} + C \sum_t (\xi^{ij})_t$$

IV. Zero-to-Stable Driver Identification Scheme

The first step towards our intelligent driver identification scheme is to identify a set of features that truthfully represent and differentiate individuals' driving behaviors. An ideal feature set would need to satisfy the following criteria: a) The set must be compact, so the resulting driver identification system is cost-effective and efficient. b) Data required to derive these features must be easy to collect. c) The data, once collected, must contain enough information to distinguish between individuals' behavioral characteristics. Only then the corresponding identification scheme would achieve high accuracy. Different drivers, in general, have different habits in maneuvering their cars. When a certain driver is behind the steering wheel – different attributes such as speed, acceleration, and change of direction reflect driving habits of that driver. Here we use a continuous series of measurements to capture these habits. All of these data elements are available through GPS sensor and do not require any other sensor or hardware for their collection of these.

Definition 1: A *drive instance* is a 4-tuple $\iota = (\mathbb{T}, \mathbb{O}, \mathbb{S}, \mathbb{A})$ of timestamp $\mathbb{T}$, orientation $\mathbb{O}$, speed $\mathbb{S}$, and acceleration $\mathbb{A}$.

Where a) *timestamp* is the instance of time at which the measurement takes place – seconds; b) *orientation* is the angle

between a car's moving direction and north – degrees; c) *speed* is the current speed of the car – meter per second; d) *acceleration* is the current change in speed of the car, which may be zero, positive or negative (in the case of deceleration) – meter per second squared. A drive of the car consists of a sequence of drive instances, as defined in definition 2.

Definition 2: Let $\iota_0, \iota_1, \iota_2, \ldots, \iota_k$ be a sequence where each $\iota_i = (\mathbb{T}_i, \mathbb{O}_i, \mathbb{S}_i, \mathbb{A}_i)$ is a drive instance. We call ι

- a *zero instance* if $\mathbb{S}_i = 0$ *or* $\mathbb{S}_i \approx 0$, and $\mathbb{A}_i = 0$;
- a *stable instance* if $\mathbb{A}_i = 0$ and $\mathbb{S}_{i-1} > 0$ and $|\mathbb{S}_{i-1} - \mathbb{S}_i| = 0$ or $|\mathbb{S}_{i-1} - \mathbb{S}_i| \approx 0$.

By a ZTS event of the car, we indicate some likely situations, e.g., a parked car moves from a carport, a car stopped at an intersection and then moves forward, or a taxi picks up or drops off passengers and drives ahead. In such scenarios, the vehicle starts from static (no speed), accelerates (and decelerates) until it reaches a constant speed. The constant speed depends on the speed limit, the real-time traffic condition as well as the driver's capability. We posit that it is the period between the car at a zero instance to the stable instance that distinguishes the behaviors among drivers. Abstractly, we conceptualize a Zero-to-Stable routine as the sequence of drive instances which starts from the zero instance and ends at the first stable instance in chronical order. While if the car is stuck in a very slow-moving traffic and needs to constantly switch from the zero to the more or less constant speed. It would make the time from the zero to the constant speed too short and would trivialize the sample. To exclude such samples from the model, we introduce a threshold $\mu > 0$ and require that the duration of the process must be longer than μ seconds. In this paper, we will always set $\mu = 10$ as it provides good empirical results.

Definition 3: A *Zero-to-Stable (ZTS) period* is a sequence of drive instances $p = (\iota_0, \iota_1, \iota_2, \ldots, \iota_k)$ such that

- $\mathbb{T}_k - \mathbb{T}_0 > \mu$;
- ι_0 is a zero drive instance;
- ι_k is a stable instance;
- none of the drive instances ι_i where $1 \leq i \leq k-1$ and $\mathbb{T}_i > \mu$ is a zero nor a stable instance.

A trip of the vehicle may thus consist of multiple ZTS periods. In ZTS $identification$, a ZTS period is quantified as a data point referred to as ZTSObject.

Definition 4: A ZTSObject is a data point ω which is composed of four features (t, o, s, and α) as defined in Table I.

Given a ZTS period $p = (\iota_0, \iota_1, \iota_2, \ldots, \iota_k)$, we define the following features (see Table I):

- t is the *time elapsed* during this ZTS period p, i.e., $\mathsf{t} = \mathbb{T}_k - \mathbb{T}_0$.
- o is the *orientation change*, defined as the variance of orientations of all drive instances in p, i.e., $o = \mathsf{variance}(\mathbb{O}_0, \ldots, \mathbb{O}_k)$.
- s is the *stable speed*, i.e. $\mathsf{s} = \mathbb{S}_k$.
- α is the *total acceleration*, i.e., $\alpha = \sum_{i=0}^{k} \mathbb{A}_i$.

The next question is how one could derive ZTS periods from a sequence of drive instance and generate a data set containing ZTSObjects. We now describe an efficient algorithm that extracts the required information at run-time. In other words, the algorithm takes an input stream of driving instances, going through each instance once and dynamically identifies ZTS periods. Once the scheme finds a ZTS period, the algorithm analyzes the instances in p and derives the corresponding ZTSObject. The process of ZTSObject extraction is described step-by-step in algorithm 1.

Algorithm 1: ZTS-Extract.

▷ INPUT: A stream of drive instances $\{\iota_0, \iota_1, \ldots\}$ such that each $\iota_i = (\mathbb{T}_i, \mathbb{O}_i, \mathbb{S}_i, \mathbb{A}_i)$
▷ OUTPUT: A stream of ZTSObjects $\omega_0, \omega_1, \ldots$ such that each $\omega_i = (\mathsf{t}_i, o_i, \mathsf{s}_i, \alpha_i)$
Initialize two counters: start $\leftarrow 0$ and end $\leftarrow 0$
Initialize $\mu \leftarrow 10$ ▷ For our experimental settings
repeat
  Read the next input $\iota_i = (\mathbb{T}_i, \mathbb{O}_i, \mathbb{S}_i, \mathbb{A}_i)$
  if $\mathbb{S}_i \approx 0$ **then**
    start $\leftarrow i$;
    end $\leftarrow i$;
  end if
  if $\mathbb{S}_i > 0$ & $|\mathbb{S}_i - \mathbb{S}_{i-1}| \approx 0$ & $\mathbb{A}_i \approx 0$ **then**
    end $\leftarrow i$
    if $\mathbb{T}_i - \mathbb{T}_{\mathsf{start}} > \mu$ and start $<$ end **then**
      $\mathsf{t} \leftarrow \mathbb{T}_{\mathsf{end}} - \mathbb{T}_{\mathsf{start}}$
      $o \leftarrow \mathsf{variance}(\mathbb{O}_{\mathsf{start}}, \mathbb{O}_{\mathsf{start}+1}, \ldots, \mathbb{O}_{\mathsf{end}})$
      $\mathsf{s} \leftarrow \mathbb{S}_{\mathsf{end}}$
      $\alpha \leftarrow \mathsf{sum}(\mathbb{A}_{\mathsf{start}}, \mathbb{A}_{\mathsf{start}+1}, \ldots, \mathbb{A}_{\mathsf{end}})$
      output ZTSObject $\omega = (\mathsf{t}, o, \mathsf{s}, \alpha)$
    end if
  end if
until Termination (all drive instances are processed)

TABLE I
THE FEATURES COMPOSING A ZTSObject

Notation	Feature Name	Range
t	Time elapsed	$\mathbb{R}^+$
o	Orientation change	$\mathbb{R}^+$
s	Stable speed	$\mathbb{R}^+$
α	Total acceleration	$\mathbb{R}^+$

The ZTS-extract algorithm processes a stream of input drive instances and outputs the sequence of all ZTSObject in linear time. It is observed in this research that vehicle comes across zero to stable event less than 30 times in routine drive. Sample space is over-fit efficiently as minimum as possible; extrapolation is limited for only time period $10 \leq \mathsf{t} \leq 30$ since μ is set to 10 for these experiments and generally, the ZTS period is less than a minute. The ZTS driver identification scheme comprises a series of key phases. Initially, input data is collected in real-time from a GPS-enabled mobile device (e.g. smartphone). Each reading of the GPS data, which contains the location, orientation and speed information of the vehicle, is then converted to a driving instance (as in definition 1). At the same time, the ZTS-extract algorithm starts to process the stream of driving instances into ZTSObjects.

V. Experiments

We perform a series of experiments to evaluate the ZTS driver identification scheme. The first experiment consists of field research conducted on city buses. The next two experiments are performed on two open-access real-world GPS datasets, respectively. We perform these experiments to exhibit proofs of concept against different real-time scenarios. Also, to validate this scheme as a practically feasible scheme for driver identification. The focus of these experiments is to test the accuracy of driver identification in the real world scenario using ZTS $identification$. Accuracy refers to driver identification rate in the rest of the paper. The equal setup for the experiments involves 3-trials per n-sized driver sets for each $n \in \{2, 3, \ldots, 5\}$ for experiment 1 and 2; whereas for each $n \in \{2, 3, \ldots, 25\}$ for experiment 3. Each trial involves an n-driver subset of the available drivers where each driver is randomly selected. Then features are extracted and data is extrapolated to ensure a significant sized training data, whereas extrapolation is limited to 10 to 30 seconds size of t in ZTSObjects for target data. The data is randomly split into training and testing part on runtime, training comprises 80% of the data and testing data comprises 20% data. For validation of ZTS $identification$ against each setup of the experiment, three supervised classification approaches were used – a) Random Forest Classifier, b) k-Nearest Neighbor classifier, and c) Support Vector Machine multi-class classifier.

A. Experiment 1: Field Research on Beijing Public Buses

We conducted field research by collecting ten city bus drivers' driving data on different bus routes in Beijing. Driver set sizes were set between two and five drivers. Data collection was carried out in real-time on the buses using indigenously built smart-phone application over a week's period. The data collection periods for different drivers vary between 1.5 to 7.2 hours. We treat drivers in this n-driver subset as a group and perform identification of drivers belonging to the group. To minimize minor errors caused due to environmental reasons e.g weather conditions, interference, or urban canyon effect affecting GPS data, we set a few constraints while applying ZTS $identification$:

1) The starting speed is zero in principle; in our tests, we treat any speed less than 1 kmph also as zero speed.
2) The final speed is the constant speed in principle; in our tests, we identify a stable speed whenever we observe that the speed difference between two consecutive records is less than 1 kmph.
3) The minimum duration of a ZTS-period was set to $\mu = 10$ seconds to deal with the situation when the vehicle is in a very slow-moving traffic.

The experiment result shows that in general, ZTS $identification$ yielded significant accuracy. For 2-drivers tests, it was observed that the highest recorded accuracy was 100%, with all three classification methods. Whereas, the average accuracy recorded for 2-driver pairs was recorded at 97%. The average accuracy for 5-drivers set was observed as 89.37%, 86.99% and 84.58% for RFC, kNN, and SVM respectively. The lowest, average, and highest accuracies achieved for each size of the set of drivers can be seen in Fig. 1.

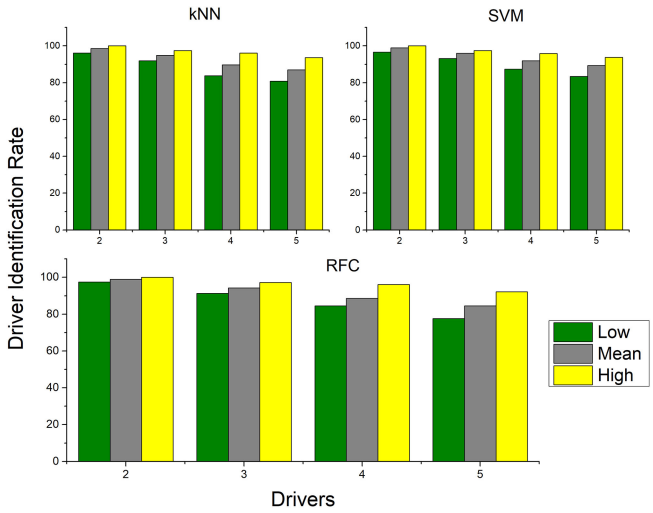

Fig. 1. The accuracy of driver identification for Beijing Public Buses dataset. (Experiment 1).

B. Experiment 2: hciLab Dataset Experiment

We use hciLab real-world driving dataset [43],[1] which provides real driving records of 10 different drivers. It contains a set of GPS data along with drivers' physiological variables, as well as other variables that are relevant to driving behaviors. To apply ZTS $identification$, we use only data elements relevant to ZTS from the data and perform this experiment. For each driver, we uniformly select 30 minutes of their driving time to produce our ZTSObjects. We also apply the constraints on the zero speed, stable speed and $\mu = 10$ as described above for experiment 1. We tested our experiments with the same three supervised machines learning algorithms as for experiment 1. Similarly, to see the overall performance we plot high, average and low driver identification rates observed from all experimental results. The highest accuracy for 2-drivers pair was observed at 97.6% with SVM classifier, whereas for 5-drivers group it was observed 90.52% with kNN classifier. Point to be noted that these results yielded on a selection of entirely random process of selection of drivers and train-test data split, hence some fluctuation in accuracy rates is subject to random data selection. Fig. 2 shows us that change in accuracy follows a trend and with an increasing number of drivers the results are not largely affected. The average accuracy for 5-drivers set was observed as 90.06%, 89.01% and 89.16% for RFC, kNN, and SVM respectively.

C. Experiment 3: Beijing Taxi Dataset Experiment

We use the dataset of Beijing taxis which is compiled at Tsinghua University [44], [45]. The dataset contains extensive anonymized data of roughly 42% of all taxis in the urban area of Beijing, providing a whole month of driving data of taxis.

[1]The dataset can be downloaded at `https://www.hcilab.org/research/hcilab-driving-dataset/`

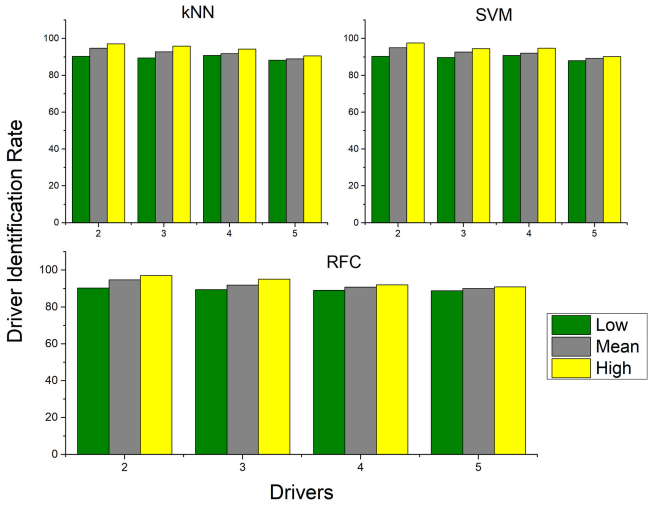

Fig. 2. The accuracy achieved with hciLab data. (Experiment 2).

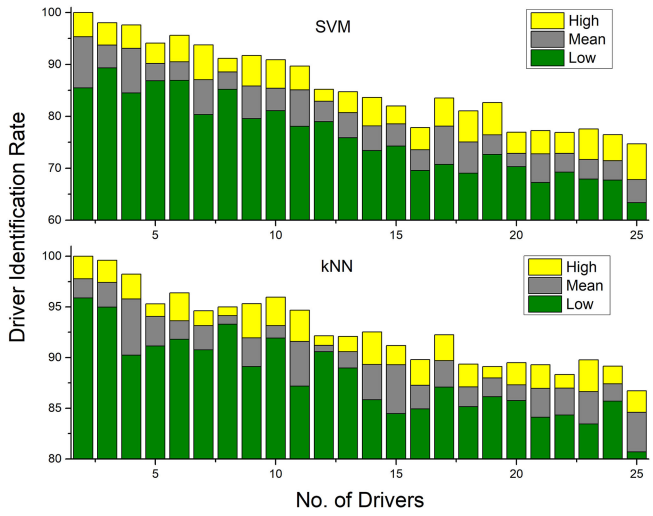

Fig. 3. The accuracy achieved with Beijing taxis data (Experiment 3) – (upper) using SVM classifier – (lower) using kNN classifier.

TABLE II
LIST OF THE DATA ELEMENTS IN EACH RECORD WITH DESCRIPTION

Data Element	Description
Car ID	Unique Car ID
Time	UTC time
Orientation	Degrees Orientation
Speed	Speed of Car
Occupant	Passenger Status

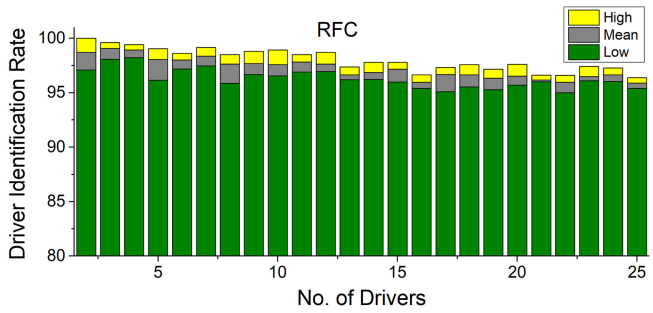

Fig. 4. The accuracy achieved with Beijing taxis data (Experiment 3) – using Random Forest classifier – on scale of 85 to 100.

The data elements used for this experiment from this dataset are stated in Table II. A taxi in Beijing typically is driven by two drivers; one driver is in charge of day-shifts while the other is in charge of night-shifts [46]; this understanding allows us to assign ground truth labels for our training sets. The night-shift data is observed to be of a smaller proportion, so it is excluded from the experiment. Then, we filter out any kind of redundant data which may affect performance. Additionally, we incorporate a new constraint to verify that the effect of high rise buildings does not affect the accuracy largely. To do so we assembled a dataset of a significant number of high rise buildings (90-meter and high) available at the time when the dataset was built through publicly available data [47], [48].

It was ensured for each driver included in the experiment, that at least 5% of the ZTSObjects were within 500 meters of identified high-rise buildings. As the dataset contains data for a large number (8500+) of drivers in Beijing, we are able to analyze data for large sets of drivers. Similar to previous experiments each trial uses a randomly selected n-drivers from the dataset. For 2-drivers pair, all three classification methods yielded 100% driver identification rate. It was observed for the 25-drivers group the lowest performing classification method among three was SVM with lowest accuracy of the 63.37% and the average of 67.81%. Whereas, kNN yielded an average accuracy of 84.59% with highest of 96.37% and RFC outperformed both algorithms with lowest 95.39% and an average of 95.89%. In Fig. 3 and Fig. 4 it can be seen that RFC is very scalable with this approach as it is least affected by the increasing size of the set of drivers. Whereas the least performing is SVM but still the lowest accuracy exceeds 60% with proposed ZTS $identification$.

D. Performance Analysis

We use receiver operating characteristic (ROC) graph [49] to analyze the performance of our proposed scheme. ROC graphs are two-dimensional graphs which express the tradeoff between false negative rate and the false positive rate. Here we plot the false negative rate on the vertical axis and false positive rate on the horizontal axis. For this, we express the following metrics

$$False\ Negative\ Rate(FNR) = 1 - TP/(TP + FN) \quad (1)$$

$$False\ Positive\ Rate(FPR) = FP/(TN + FP) \quad (2)$$

The *false negative rate* is the percentage of misclassifications where an authentic driver is not correctly identified and the *false positive rate* is the misclassification rate where an unauthentic driver is recognized as an authentic driver. In these metrics, TP denotes true positives, TN is true negatives, FN is false negatives and, FP is false positives. We compute these metrics and calculate EER using results from experiment 3. We calculate FNR and FPR for each class in each model with each n-sized set of drivers, then calculate the mean FNR and FPR for each

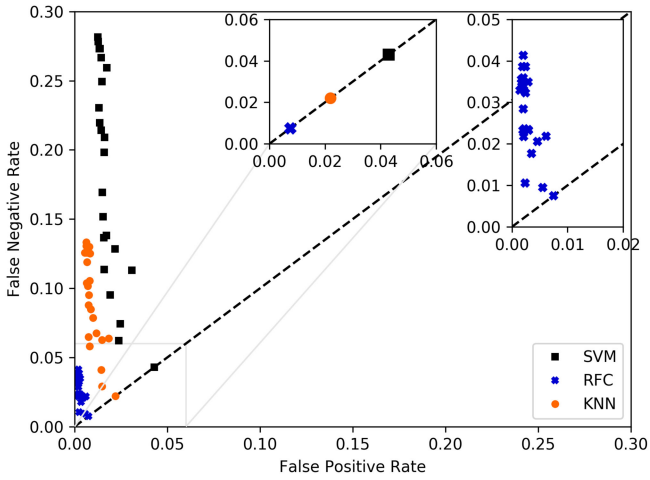

Fig. 5. The ROC graph of proposed scheme showing EERs – with Beijing taxis data – marked inset shows EER for all classification methods – second inset shows all ROC points for RFC method.

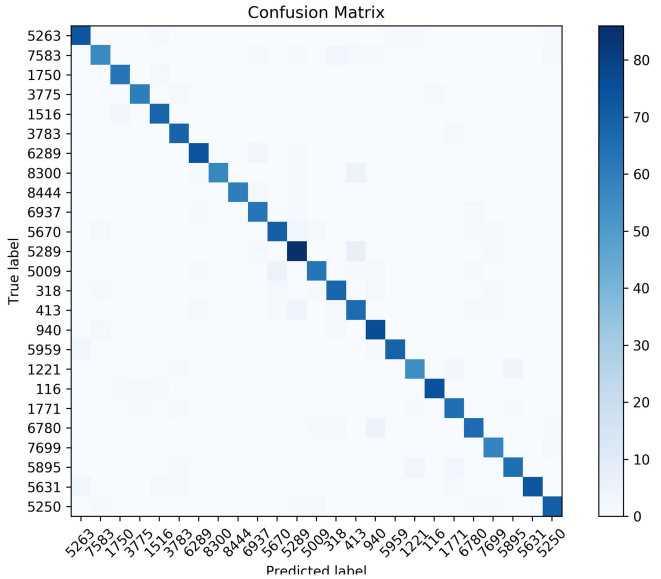

Fig. 6. Confusion matrix for a 25-drivers test with ZTS *identification* using RFC. (Experiment 3).

model. Fig. 5 shows two kinds of points and one EER line. The EER line is reference line for all points that represent equality between FNR and FPR. Whereas each point as seen in the figure represents a pair of FNR, FPR for all three classification methods where FNR is equal to FPR. The EER for RFC was recorded at 1.2% and for kNN and SVM it was observed as 2.1% and 4.6% respectively. This shows the high achieving accuracy of this approach with these algorithms, it is observed that ZTS *identification* performs best with RFC classification method. We also present one of confusion matrix to show further detail of accuracy of the ZTS *identification*, a confusion matrix is composed of predicted versus true labels for a prediction test of a system. Each cell of the matrix accounts for one true and one predicted label represents a portion of tests where system was tested for the *true* label and prediction was the *predicted* label. Confusion matrix for a test case with RFC classification method as seen in Fig. 6, shows the accuracy of the proposed scheme for large number of drivers. As advised by Fig. 4, error in average performance is approximately $\pm 0.5\%$, hence such confusion matrix is highly occurring for ZTS *identification*.

E. Discussion and Results Comparison

We compare the results of our proposed scheme with different existing driver identification schemes. The comparison of accuracy is based on our experimental results and the results specified in the respective previous works. Table III lists the feature vector to present the complex nature of data mining through different sources which are cost-ineffective for high-scale implementation in real-time.

Hallac *et al.* [1] used 12 sensor signals to identify two to five drivers, with an average accuracy (over 2-drivers) of 76.9%. Van Ly *et al.* [50] used six individual signals and four combinations of those signals for pairs of drivers, the best result achieved by this method was approximately 60%. Zhang *et al.* [15] tested their approach on a predefined route and naturalistic driving, the data used was collected through three phone sensors and seven car signals. Car signals were collected using the on-board diagnostics (OBD) which is not available in all kinds of cars. This method achieved an average accuracy of 80.83% for natural driving and 77.3% average accuracy for a predefined route. The highest accuracy observed in prior works was scored by Martinez *et al.* [14], where fourteen different signals yielded an average of 96.95% accuracy. Point to be considered, while comparing the results of our method and this method the data used in this method was recorded from a CAN-bus. It requires specific kind of hardware to be available on vehicles, and the

TABLE III
LIST OF THE FEATURE-VECTORS USED IN THE COMPARED RESEARCHES

Research	Feature Vector
[1]	Steering – wheel angle, velocity, acceleration; Vehicle – velocity, heading, engine resolution per minute, gas pedal position. brake pedal position, throttle position, forward acceleration, lateral acceleration, torque;
[5]	Mean, Median, Skewness, Kurtosis, Standard deviation, Maximum, Minimum , 97.5th percentile, 1st and 3rd Quartiles, Interquartile range and 2.5th percentile of ten unique data sources in total 137;
[14]	Audio & video recordings; CAN-bus signals; gas pedal & brake pedal sensor recordings; a frontal laser scanner; an inertial measurement unit (IMU) with XYZ accelerometers; measures of rotation rates – pitch, roll, and yaw;
[15]	Histogram; Power Spectrum; Cepstrum; Entropy of – Histogram, Power Spectrum and Cepstrum; Standard Deviation of – Power Spectrum and Cepstrum; Max, Min, Mean, Median, and Variance of the data obtained from seven car sensors and three phone sensors;
[18]	51 data elements collected through CAN-bus of the vehicles; 9 GPS data elements; 11 inertial sensors data elements; 22 different data elements [41] – 40-element feature vector.
[50]	Various combinations of Histogram, Minimum, Maximum, Mean of Y-Acceleration and Gyroscope Signal; Duration of Y-Acceleration;
Proposed	Listed in Table I

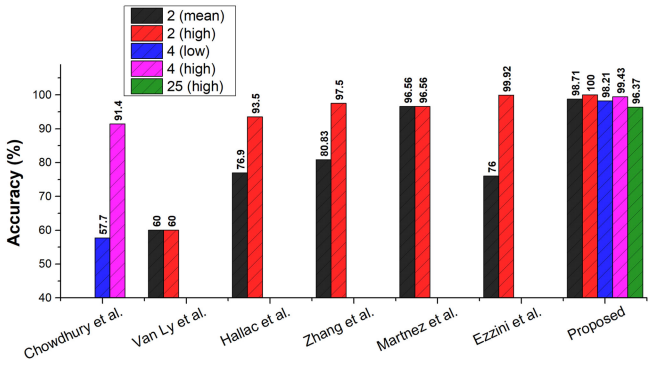

Fig. 7. Comparison of driver identification rate among different methods. (right) the zoomed graph for a better understanding of values.

method involves in-depth readings such as brake and gas pedal monitoring. While our method requires data only from GPS sensor for data collection. Scheme by Chowdhury *et al.* [5], achieved its highest accuracy at 91.4% for 4-drivers with 137 statistical features derived from GPS. The average accuracy for 2-drivers yielded by our proposed scheme is approximately 97% whereas the highest accuracy of driver identification was 100%. The scheme by Ezzini *et al.* [18], achieved 99% using 40 features derived from different data sources, while the accuracy for GPS-only data was recorded 76%.

Fig. 7 summarizes the results in the form of a graph with different available results for other methods and ZTS $identification$. It should be kept under consideration that the results presented in this article were achieved through a random selection of drivers and randomized selection of train and test data with least extrapolation required, as well as the no test data was extrapolated. The proposed scheme has the potential to yield more depending on the target computation capability. The EER of ZTS $identification$ was measured at 1.2%, which is the lowest among the different methods.

F. Cost-Effectiveness and Scalability Overview

The size of feature vector and the hardware resources which make the data available for processing are important for various critical reasons. Following are few costs to consider which may make a system feasible or not feasible for high-scale implementation.

- Individual costs of the hardware units of different sensors which help collect data from the vehicle and environment.
- Computing resources required for deriving the corresponding feature space from raw data.
- Data-warehouse storage costs to cater large-scale implementation.

The attributes which constitute proposed feature vector are dependant *only* on GPS for data collection, which reduces the cost by larger proportion in comparison to schemes which depend on different hardware resources. The GPS data elements which play important role in the proposed scheme can be stored in *only 23 bytes* given that $\mathbb{T}$ is stored as ten-digit unsigned integer; $\mathbb{O}$ as a short integer; $\mathbb{S}$ and $\mathbb{A}$ as floating point numbers. The t element can be stored in an unsigned small integer, whereas o, s, and α are floating point numbers and can be stored in float type variables. In terms of database storage a single ZTSObject with an integer label can be stored within *only 26 bytes* of memory having significant precision. These factors combined with accuracy for the larger number of drivers make the proposed scheme applicable, cost-effective and scalable.

VI. Conclusion and Future Work

In this research, we proposed ZTS $identification$ which profiles driver behavior to identify a driver among a set of drivers. It uses the GPS data to extract features, without the need for any specific hardware apart from a smart-phone or in-vehicle GPS. The proposed set of features in ZTSObjects is easy to derive and small in number which makes this scheme more practical and cost-effective. On the contrary, to the most previous works on driver profiling which focuses exclusively on the accuracy, this work's novel contribution lies in the emphasis on ease of data collection, non-intrusiveness, and cost-effectiveness. Proposed scheme yielded the highest accuracies using RFC classification approach, besides that we compared the performance of scheme with kNN and SVM classifiers. Three sets of experiments performed, on different data sets, including a field research on Beijing buses and two open-access datasets. The highest driver identification accuracy for two drivers was observed 100%, whereas, for drivers up to 25, the accuracy is observed more than 96%. Furthermore, the addition of an impostor detection mechanism to the scheme is part of the continuation of this work. We further point out that, in future features used in the proposed scheme can be utilized to solve other problems related to driver profiling, such as detection of drunk driving, or rash driving.

References

[1] D. Hallac *et al.*, "Driver identification using automobile sensor data from a single turn," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst.*, 2016, pp. 953–958.
[2] C.-H. Yang, D. Liang, and C.-C. Chang, "A novel driver identification method using wearables," in *Proc. 13th IEEE Int. Annu. Conf. Consum. Commun. Netw.*, 2016, pp. 1–5.
[3] M. V. Martinez, I. D. Campo, J. Echanobe, and K. Basterretxea, "Driving behavior signals and machine learning: A personalized driver assistance system," in *Proc. IEEE 18th Int. Conf. Intell. Transp. Syst.*, 2015, pp. 2933–2940.
[4] L. Moreira-Matias and H. Farah, "On developing a driver identification methodology using in-vehicle data recorders," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2387–2396, Sep. 2017.
[5] A. Chowdhury, T. Chakravarty, A. Ghose, T. Banerjee, and P. Balamuralidhar, "Investigations on driver unique identification from smartphone's GPS data alone," *J. Adv. Transp.*, vol. 2018, 2018, 11 pages.
[6] L.-K. Chen *et al.*, "Identification of a driver steering model, and model uncertainty, from driving simulator data," *Trans. ASME J. Dyn. Syst. Meas. Control*, vol. 123, no. 4, pp. 623–629, 2001.
[7] C. C. MacAdam, "Understanding and modeling the human driver," *Vehicle Syst. Dyn.*, vol. 40, no. 1–3, pp. 101–134, 2003.
[8] J.-P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *IEEE Secur. Privacy*, vol. 2, no. 3, pp. 49–55, May/Jun. 2004.
[9] Z. Liu and G. He, "Research on vehicle anti-theft and alarm system using facing recognition," in *Proc. Int. Conf. Neural Netw. Brain*, 2005, vol. 2, pp. 925–929.

[10] A. Riener and A. Ferscha, "Supporting implicit human-to-vehicle interaction: Driver identification from sitting postures," in *Proc. 1st Annu. Int. Symp. Veh. Comput. Syst.*, 2008.

[11] J.-D. Wu and S.-H. Ye, "Driver identification based on voice signal using continuous wavelet transform and artificial neural network techniques," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 1061–1069, 2009.

[12] J.-D. Wu and S.-H. Ye, "Driver identification using finger-vein patterns with radon transform and neural network," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 5793–5799, 2009.

[13] C. Miyajima *et al.*, "Driver modeling based on driving behavior and its evaluation in driver identification," *Proc. IEEE*, vol. 95, no. 2, pp. 427–437, Feb. 2007.

[14] M. V. Martinez, J. Echanobe, and I. Del Campo, "Driver identification and impostor detection based on driving behavior signals," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst.*, 2016, pp. 372–378.

[15] C. Zhang, M. Patel, S. Buthpitiya, K. Lyons, B. Harrison, and G. D. Abowd, "Driver classification based on driving behaviors," in *Proc. 21st Int. Conf. Intell. User Interfaces*, 2016, pp. 80–84.

[16] A. Burton *et al.*, "Driver identification and authentication with active behavior modeling," in *Proc. IEEE 12th Int. Conf. Netw. Service Manage.*, 2016, pp. 388–393.

[17] I. D. Markwood and Y. Liu, "Vehicle self-surveillance: Sensor-enabled automatic driver recognition," in *Proc. 11th ACM Asia Conf. Comput. Commun. Secur.*, 2016, pp. 425–436.

[18] S. Ezzini, I. Berrada, and M. Ghogho, "Who is behind the wheel? driver identification and fingerprinting," *J. Big Data*, vol. 5, no. 1, 2018, 9 pages.

[19] Y. Qiao, Y. Cheng, J. Yang, J. Liu, and N. Kato, "A mobility analytical framework for big mobile data in densely populated area," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 1443–1455, Feb. 2017.

[20] Y. Cui and S. S. Ge, "Autonomous vehicle positioning with GPS in urban canyon environments," in *Proc. Int. Conf. Robot. Autom.*, 2001, vol. 19, no. 1, pp. 15–25.

[21] S. Y. Cho, B. D. Kim, Y. S. Cho, and W. S. Choi, "Observability analysis of the INS/GPS navigation system on the measurements in land vehicle applications," in *Proc. Int. Conf. Control, Autom. Syst.*, 2007, pp. 841–846.

[22] H. Zhang, J. Liu, and N. Kato, "Threshold tuning-based wearable sensor fault detection for reliable medical monitoring using Bayesian network model," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1886–1896, Jun. 2018.

[23] Z. Ma, H. Yu, W. Chen, and J. Guo, "Short utterance based speech language identification in intelligent vehicles with time-scale modifications and deep bottleneck features," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 121–128, Jan. 2019.

[24] Z. Ma *et al.* "Fine-grained vehicle classification with channel max pooling modified CNNs," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3224–3233, Apr. 2019.

[25] F. Tang, Z. M. Fadlullah, N. Kato, F. Ono, and R. Miura, "AC-poca: Anticoordination game based partially overlapping channels assignment in combined uav and d2d-based networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1672–1683, Feb. 2018.

[26] D. Takaishi, Y. Kawamoto, H. Nishiyama, N. Kato, F. Ono, and R. Miura, "Virtual cell based resource allocation for efficient frequency utilization in unmanned aircraft systems," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 3495–3504, Apr. 2018.

[27] M. Imani, S. F. Ghoreishi, and U. M. Braga-Neto, "Bayesian control of large mdps with unknown dynamics in data-poor environments," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 8146–8156.

[28] M. Imani, S. F. Ghoreishi, D. Allaire, and U. Braga-Neto, "Mfbo-SSM: Multi-fidelity Bayesian optimization for fast inference in state-space models," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, pp. 7858–7865.

[29] K. Zhang, M. Sun, T. X. Han, X. Yuan, L. Guo, and T. Liu, "Residual networks of residual networks: Multilevel residual networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 6, pp. 1303–1314, Jun. 2018.

[30] K. Zhang, L. Guo, C. Gao, and Z. Zhao, "Pyramidal ror for image classification," *Cluster Comput.*, vol. 22, pp. 1–11, 2017.

[31] K. Zhang *et al.*, "Fine-grained age estimation in the wild with attention LSTM networks," *IEEE Trans. Circuits Syst. Video Technol.*, to be published, doi: 10.1109/TCSVT.2019.2936410.

[32] Y. Yin, L. Chen, Y. Xu, J. Wan, H. Zhang, and Z. Mai, "QoS prediction for service recommendation with deep feature learning in edge computing environment," *Mobile Netw. Appl.*, pp. 1–11, 2019.

[33] X. Zeng, G. Xu, X. Zheng, Y. Xiang, and W. Zhou, "E-AUA: An efficient anonymous user authentication protocol for mobile IoT," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1506–1519, Apr. 2019.

[34] L. Zhu, M. Li, Z. Zhang, and Z. Qin, "ASAP: An anonymous smart-parking and payment scheme in vehicular networks," *IEEE Trans. Dependable Secure Comput.*, to be published, doi: 10.1109/TDSC.2018.2850780.

[35] M. Li, L. Zhu, and X. Lin, "Efficient and privacy-preserving carpooling using blockchain-assisted vehicular fog computing," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4573–4584, Jun. 2019.

[36] G. Xu, J. Liu, Y. Lu, X. Zeng, Y. Zhang, and X. Li, "A novel efficient maka protocol with desynchronization for anonymous roaming service in global mobility networks," *J. Netw. Comput. Appl.*, vol. 107, pp. 83–92, 2018.

[37] Y. Li, Q. Luo, J. Liu, H. Guo, and N. Kato, "TSP security in intelligent and connected vehicles: Challenges and solutions," *IEEE Wireless Commun.*, vol. 26, no. 3, pp. 125–131, Jun. 2019.

[38] J. Wang, J. Liu, and N. Kato, "Networking and communications in autonomous driving: A survey," *IEEE Commun. Surv. Tut.*, vol. 21, no. 2, pp. 1243–1274, Secondquarter 2019.

[39] J. Engelbrecht, M. J. Booysen, G.-J. Van Rooyen, and F. J. Bruwer, "Survey of smartphone-based sensing in vehicles for intelligent transportation system applications," *IET Intell. Transport Syst.*, vol. 9, no. 10, pp. 924–935, 2015.

[40] W. Loh, "Classification and regression trees," *Wiley Interdisciplinary Rev. Data Mining Knowl. Discovery*, vol. 1, no. 1, pp. 14–23, 2011.

[41] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967.

[42] C. Chang and C. Lin, "Libsvm: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, 2011, 27 pages.

[43] S. Schneegass, B. Pfleging, N. Broy, F. Heinrich, and A. Schmidt, "A data set of real world driving to assess driver workload," in *Proc. 5th Int. Conf. Autom. User Interfaces Interactive Veh. Appl.*, 2013, pp. 150–157.

[44] X. Yu, H. Zhao, L. Zhang, S. Wu, B. Krishnamachari, and V. O. Li, "Cooperative sensing and compression in vehicular sensor networks for urban monitoring," in *Proc. Commun., IEEE Int. Conf.*, 2010, pp. 1–5.

[45] X. Yu, H. Zhao, L. Zhang, S. Wu, B. Krishnamachari, and V. O. Li, "Datasets, sensors research lab, electronic engineering department," 2018. [Online]. Available: http://sensor.ee.tsinghua.edu.cn/datasets.html, Accessed: Dec. 14, 2016.

[46] D. Zhang *et al.*, "Understanding taxi service strategies from taxi gps traces," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 123–135, Feb. 2015.

[47] Emporis, "High-rise buildings in beijing," 2018. [Online]. Available: https://www.emporis.com/city/100214/beijing-china/type/high-rise-buildings, Accessed: Apr. 28, 2018

[48] Emporis, "Tallest buildings in beijing," 2018. [Online]. Available: https://www.emporis.com/statistics/tallest-buildings/city/100214/beijing-china, Accessed: Apr. 28, 2018

[49] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, 2006.

[50] M. Van Ly, S. Martin, and M. M. Trivedi, "Driver classification and driving style recognition using inertial sensors," in *Proc. Intell. Vehicles Symp.*, 2013, pp. 1040–1045.