IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 6, 2019, accepted December 28, 2019, date of publication January 6, 2020, date of current version January 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2964029

A New Vehicular Fog Computing Architecture for Cooperative Sensing of Autonomous Driving

HAO DU, SUPENG LENG, FAN WU, XIAOSHA CHEN, AND SUN MAO
School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Corresponding author: Supeng Leng (spleng@uestc.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFE0117500, in part by the Science and Technology Program of Sichuan Province, China, under Grant 2019YFH0007, and in part by the EU H2020 Project COSAFE under Grant MSCA-RISE-2018-824019.

ABSTRACT The sensing coverage and accuracy of vehicles are vital for autonomous driving. However, the current sensing capability of a single autonomous vehicle is quite limited in the complicated road traffic environment, which leads to many sensing dead zones or frequent misdetection. In this paper, we propose to develop a Vehicular Fog Computing (VFC) architecture to implement cooperative sensing among multiple adjacent vehicles driving in the form of a platoon. Based on our VFC architecture greedy and Support Vector Machine (SVM) algorithms are adopted respectively to enhance the sensing coverage and accuracy in the platoon. Furthermore, the distributed deep learning is processed for trajectory prediction by applying the Light Gated Recurrent Unit (Li-GRU) neural network algorithm. Simulation results based on real-world traffic datasets indicate the sensing coverage and accuracy by the proposed algorithms can be significantly improved with low computational complexity.

INDEX TERMS Intelligent vehicles, vehicular fog computing, cooperative sensing, autonomous driving.

I. INTRODUCTION

Autonomous driving has received wide attention from the academy and industry. Benefiting from the high accuracy, small size and low cost of on-board sensors, the perception ability of intelligent vehicles can be highly improved, making autonomous driving safe and promising [1]. However, the sensing ability of a single autonomous vehicle could be quietly limited due to dead zones and misdetection, which has led to some traffic accidents according to recent news reports. Fortunately, with the help of the internet of vehicles (IoV), autonomous vehicles can be connected and share their sensing information, thus the driving satiety and traffic efficiency can be improved significantly [2].

Meanwhile, grouping vehicles into platoon is recognized as a promising method to enhance the safety of autonomous driving [3]. Autonomous vehicles with similar driving speed towards the same direction can be organized as a connected platoon, in which each platoon member can communicate with each other with low latency and high transmission rate in IoV [4], [5]. In this case, sensing data can be shared in the platoon so that the platoon can maintain safe and harmonious driving. However, autonomous driving will generate massive sensing tasks with high computational complexity and high delay sensibility [6] in a dynamic IoV scenario [7].

The previous studies have proved the edge computing to be adopted for autonomous driving [8]. Edge computing has been extensively studied to handle the large latency, unstable connection, and network congestion of conventional cloud computing-based approaches [9]–[11]. With edge computing, the computational tasks of vehicles are offloaded to Road Side Units (RSUs) through a multi-access network [12]–[14]. Nevertheless, the enormous data from cooperative sensing of the autonomous platoon will put a heavy burden on the cellular networks which cannot satisfy the low delay requirement of autonomous driving.

Furthermore, vehicular fog computing (VFC) provides a promising solution to leverage the computational abilities and reliable wireless connectivity of intelligent vehicles [15]. VFC facilitates nearby intelligent vehicles to carry out a substantial amount of communication and computation cooperatively. Different from other existing techniques, the advantages of VFC include proximity to end-users, dense geographical distribution, and good mobility support [16], [17]. Although VFC enables the computation offloading

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

among intelligent vehicles, the edge server (RSU) or remote cloud is required to assist the offloading scheduling in the existing VFC architecture.

Motivated by the above observations, we propose a new VFC architecture and intelligent algorithms to perform cooperative sensing to improve the sensing coverage and accuracy of autonomous driving vehicles, as well as driving safety. The real traffic data from the practical environment, namely NGSIM I-80 data and US 101 data [18], is used to evaluate our proposed methods. The contributions of this paper are summarized as follows.

- In order to overcome the weakness of traditional VFC architecture, we proposed a new VFC architecture for cooperative sensing in a platoon, which can jointly utilize the sensing and the computational abilities of intelligent vehicles. This architecture takes full advantage of platoon driving. To the best of our knowledge, it is the first time that autonomous vehicle platoon organized as a vehicular fog to share their sensing, communication and computation resources for enhancing safety.
- We propose greedy and SVM algorithms respectively for cooperative sensing to enhance the sensing coverage and accuracy. Based on the proposed VFC architecture, cooperative sensing tasks are processed by multiple vehicles in the platoon. As a consequent, the computing complexity can be greatly reduced.
- We propose a distributed deep learning for Li-GRU neural network to predict lane change manoeuvre. The training task is offloaded to intelligent vehicles based on our proposed VFC architecture. The training time is much lower than traditional centralized learning. Furthermore, we make a comparison with the existing approaches for vehicle driving prediction. The Li-GRU neural network algorithm has a good performance in terms of prediction accuracy.

The remainder of this paper is organized as follows. Section II presents the survey of related work. In Section III, we introduce the VFC architecture for cooperative sensing. In Section IV, sensing cooperation based on our VFC architecture is discussed. In Section V, we propose distributed deep learning for the Li-GRU algorithm to enhance prediction accuracy. In Section VI, we provide numerical testing results of our VFC architecture for cooperative sensing. In Section VII, conclusion and future research of our work.

II. RELATED WORK

Recent works investigated the architecture of vehicular fog computing. Approaches for task allocation is widely studied. Cooperative sensing is also discussed extensively. Some state-of-the-art researches are discussed to motivate us to make deeply explore.

A. VEHICULAR FOG COMPUTING

With the rapid development of intelligent vehicles, the VFC has attracted considerable attention from both industry and academia. In [15], X. Hou et al. discussed the practicability of VFC and analyzed the mobility, connectivity, and capacity. A new framework named autonomous vehicular edge (AVE) was proposed in [8] to improve the computation abilities of vehicles. K. Xiong et al. proposed a machine-learning based framework to allocate the heterogeneous resources of VFC based on the requirements of the tasks in [17]. In [19], Zhu et al. proposed Fog following me (Folo), a dynamic task allocation solution for vehicular fog computing which aims at minimizing average service latency while reducing the overall quality loss. In [20], G. Qiao et al. propose a content cache scheme based on the deep deterministic policy gradient, which jointly optimizes content placement and content delivery in vehicular edge computing and networks. The authors in [21] presents a visionary concept on vehicular fog computing that turns connected vehicles into mobile fog nodes and utilizes the mobility of vehicles for providing cost-effective and on-demand fog computing for vehicular applications. Nevertheless, all the vehicles in these works are willing to act as fog nodes, and the incentive issues should take into account.

The work [22] formulated the negotiation between task publisher and fog nodes as an optimization problem. The optimal contract is the Nash equilibrium solution achieved by task publisher and fog nodes. J. Zhao et al. presented a collaborative approach based on MEC and cloud computing that offload services to automobiles in vehicular networks in [12]. In addition, [23] proposed a framework of content delivery with parked vehicles, where moving vehicles can obtain content from both the Road Side Unite (RSU) and parked vehicles according to the competition and cooperation among them. Then, based on a Stackelberg game, a pricing model including moving vehicles, RSU, and parked vehicles, can obtain their maximum utilities. Moreover, [24] and [25] considered the power allocation in cognitive radio networks. However, in most of the current works' frameworks, the scheduling of task offloading is dependent on the edge server or fog server (RSU). These existing frameworks are unable to satisfy the low-latency for autonomous driving of intelligent vehicles platoon and cannot cooperate with the sensing task with computational requirements to ensure the safety of autonomous driving.

B. COOPERATIVE SENSING

Meanwhile, sensing fusion of intelligent vehicles has been extensively studied in recent literature. The fusion model can be divided into three categories, i.e., fusion for homogeneous sensors of multi-vehicles [26], fusion for heterogeneous sensors of a single vehicle [27], and fusion for heterogeneous sensors of multi-vehicles. Most researches focus on the first two fusion methods. Few approaches addressing how to fusing the heterogeneous sensing information from different kinds of sensors of multiple vehicles.

Recently, the occupancy grid mapping and filtering algorithm are widely adopted for mapping environments state into grid states. A SLIC superpixels based clustering for

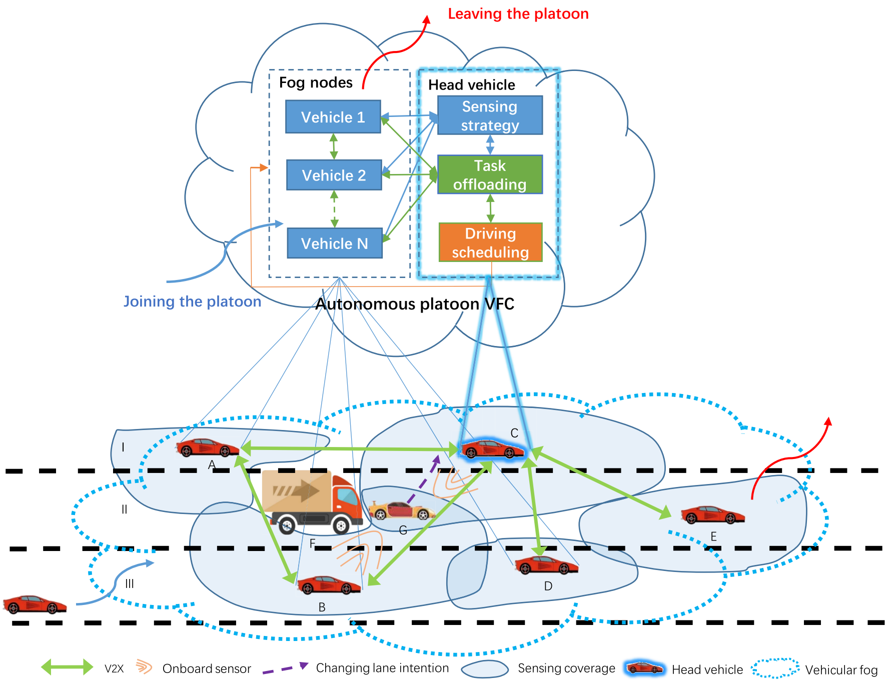

FIGURE 1. **VFC architecture for cooperative sensing.**

a grid cell cluster (GCC) to particles measured using light detection and ranging and dense depth maps extracted from a stereo vision and LIDAR sensor [28]. Unfortunately, the OGF from LIDAR and stereo vision sensor data are based on ego-vehicle. This method cannot render the dead zone and the accuracy cannot meet the autonomous driving requirement. The fast occupancy Grid Filtering (OGF) is an effective method for sensing data fusion proposed by [29], where the LIDAR and stereo camera information can be fused efficiently and quickly. Unfortunately, the sensing range of the OGF method is limited and unable to detect and predict the platoon scenario. Nevertheless, multi-vehicle cooperative sensing with heterogeneous sensors is a promising solution for these problems. The work in [26]proposed an occupancy probability distribution for OGF map fusion, but the accuracy is not sufficient to detect whether the grid is occupied. Furthermore, based on the cooperative sensing data, the GRU algorithm that is a modified LSMT algorithm has better accuracy and lower computational complexity than LSTM [30]. However, traditional centralized training for the neural network is hard to satisfy the time delay of autonomous driving. Training tasks can be offloaded to the intelligent vehicles in the platoon based on our VFC architecture which enables the distributed learning to reduce training time.

In summary, VFC is a promising technique to improve the sensing accuracy for autonomous driving. The existing works consider much about the Fog Radio Access Networks (FRAN) and the task offloading of VFC. The traditional VFC architecture concern little about the mobility of the autonomous driving platoon and the independent cooperation of the communication, computation, and sensing in the platoon.

III. ARCHITECTURE OF VFC FOR COOPERATIVE SENSING

In this section, we introduce the autonomous platoon scenario and our new architecture of VFC for cooperative sensing of autonomous driving. The cooperative sensing scenario map is shown in FIGURE 1. Autonomous vehicles A, B, . . . , E consist of a cooperative sensing platoon and each vehicle has a sensing coverage. Assume vehicle G intends to change lane, but vehicle G is in the dead zone of Vehicle A. Besides, suppose that there doesn't exist any RSU with MEC to assist these vehicles, the cooperative sensing process sensing is shown as follow.

A couple of autonomous vehicles run at a relatively stable speed and distance. Consequently, they can be connected based on IoV to inform an autonomous platoon. With the communication and computation abilities of autonomous vehicles, this platoon can be organized as a vehicular fog. Meanwhile, the vehicle with powerful ability of computation and communication is appointed to be the head vehicle to act as the server and the other vehicles act as the fog nodes. The head vehicle can manage the resource of this platoon. Different from the traditional VFC architecture, the task offloading application can be deployed on intelligent vehicles. Then, the head vehicle chooses proper cooperative sensing and task offloaded strategies according to the sensing requests. Subsequently, the surrounding states are fed into the neural network to predict the trajectory of the surrounded vehicles. The training process can also be offloaded to the vehicles in this platoon. Finally, the warning messages are distributed to each autonomous vehicle sensing platoon immediately.

Different from traditional VFC architecture, this new VFC architecture can take the full advantages of the communication, computation and sensing abilities of intelligent vehicles. At the same time, this VFC architecture enables the autonomous platoon organized as a vehicular fog and ensure driving safety independently. In the following sections, we discuss the benefits of cooperative sensing based on our VFC architecture.

IV. COOPERATIVE SENSING ENHANCEMENT

Based on our VFC architecture, we processed sensing cooperation for autonomous driving. We consider two kinds of sensing cooperation tasks which are accumulative sensing tasks such as [31] and best-quality sensing tasks such as [32]. We first adopt greedy algorithm to optimal cooperative sensing strategy to enhance sensing coverage. Then, we use SVM algorithm to fuse the multi-vehicles sensing data to get an accurate state of the vehicles in our sensing platoon. Finally, the sensing data are fed into a Li-GRU neural network algorithm to predict the lane changing manoeuvre of the target vehicle. In addition, we use real-world traffic data NGSIM I-80 Data and US 101 Data [18] to train our SVM algorithm.

A. COVERAGE ENHANCEMENT

According to the sensing task, the head vehicle selects vehicles in this platoon to process the sensing task and task allocation. We consider the best-quality sensing task for our cooperative sensing coverage enhancement. In other words, we select vehicles that have the best sensing coverage to process the sensing task. In particular, autonomous vehicles in the platoon has different sensing abilities. $N = \{1, 2, 3, \ldots, n\}$ denotes that there exists n autonomous vehicles in the platoon. We define s_i as the sensing coverage of vehicle i, $i \in N$ in the platoon. The sensing coverage set can be define as $S = [s_1, s_2, \ldots, s_n]$, where s_i is the coverage field of vehicle i. Then, we define the sensing strategy as $A = [a_1, a_2, \ldots, a_n]$, therein $a_i = [0, 1]$, where 1 means that the vehicle is selected to process sensing task and 0 means the vehicle is not selected. Accordingly, the full sensing coverage of the autonomous vehicles platoon is defined as:

$$S_f = \mathcal{M}\left(\left(\bigcup_{i\in N} s_i a_i\right) \bigcap S_R\right), \tag{1}$$

where $\mathcal{M}(s)$ is the area of the field s and S_R is the road field. Meanwhile, the cost of invalid coverage should take into account. In particular, there could exist sensing coverage overlaps between the nearby vehicles. As mentioned before, the coverage sensing task is a best-quality sensing task and the sensing overlap of the full platoon is:

$$S_o = \bigcup_{i,j\in N, i<j} \left\{s_i a_i \bigcap s_j a_j\right\} \tag{2}$$

Our optimal goal is to find the optimal A^* to maximum the sensing coverage and minimum the coverage overlap. We use greedy algorithm to get the optimal strategy A^* to enhance the sensing coverage. The optimal goal is shown as follow:

$$S_f{}^* = \arg\max_{a_i{}^*} \mathcal{M}\left(\bigcup_{i\in N} s_i a_i{}^*\right) \tag{3}$$

$$S_o{}^* = \arg\min_{a_i{}^*} \mathcal{M}\left(\bigcup_{i,j\in N, i<j} \left\{s_i a_i{}^* \bigcap s_j a_j{}^*\right\}\right) \tag{4}$$

Besides, we define three parameters to evaluate the efficiency of cooperative sensing coverage which are *AreaRatio*, *Effectness* and *TotalRatio*. The definition is as follow:

$$AreaRatio = \frac{S_f}{S_f + S_o} \tag{5}$$

$$Effectness = \frac{S_f}{\mathcal{M}\left(\sum_{n=1}^{N} s_i a_i\right)} \tag{6}$$

$$TotalRation = \frac{S_f}{S_R} \tag{7}$$

where the *AreaRatio* means the ratio of selected vehicles' sensing coverage to full coverage of all vehicles. Moreover, to cover more roads of this platoon, we pursue a high *TotalRatio*. A high *Effectness* means overlap of sensing coverage of selected vehicles is little.

B. GREEDY VEHICLE SELECTION ALGORITHM

To jointly solve problem (3) and (4), we design a greedy vehicle selection algorithm. We will select vehicles iteratively according to the increased sensing area. Given the selected vehicle set N_s, for each vehicle $i \in N - N_s$, we will calculate the new area by

$$\mathcal{M}\left(S_{N_s,i}\right) = \int_{S_R} I_{(s_i - s_i \cap S(N_s))}(x, y)dxdy, \tag{8}$$

where $I_s(x, y)$ equals 1 when $(x, y) \in s$, otherwise equals 0. And $S(N_s) = \bigcup_{i\in N_s} s_i$. We start from an empty selected vehicle set, and calculate the utilities of each vehicle by (8), add the vehicle with highest utility into the selected vehicle set and repeat the process, until the coverage ratio reaches the threshold, which is defined as

$$\rho = \frac{\mathcal{M}\left(S_{N_s,i}\right)}{\mathcal{M}(S_R)}. \tag{9}$$

The pseudo-code of the selection process is presented in Algorithm 1.

If we selected all candidate vehicles, the number of integrals will be $O(n^2)$. If all of the computations are processed by the head vehicle, the computation delay will be intolerable. However, in each loop in Algorithm 1, the integrals of each utility of the unselected vehicles are independent of each other, which makes it straightforward to parallel the calculation process by the task offloading technology. For example, in the i-th iteration, there are $n-i$ vehicles not been selected. The head vehicle can choose the first $n-i$ vehicles ordered by the number of available computation resources.

Algorithm 1 Greedy Vehicle Selection Algorithm

Input: s_i, S_R, ρ^*.
Output: Optimum selection strategy a^*.

Initialisation: $N_s = \emptyset$, $\rho = 0$
while $\rho < \rho*$ and $N_s \neq N$ **do**
  $n_s = -1, u = -\infty$
  for All vehicles i in $N - N_s$ **do**
    Calculate utility u_i with (8)
    if $u_i > u$ **then**
      $n_s = i, u = u_i$
    end if
  end for
  $N_s = N_s \bigcup \{n_s\}$
  Calculate ρ according to updated N_s and (9)
end while
$a_i^* = 1$ if $i \in N_s$, otherwise $a_i^* = 0$
return a^*

Then offload each utility calculation task to each vehicle, and obtain the result. Meanwhile, the messages transmitted between the head vehicle and other vehicles only contain the selected vehicles and the utility value, which is short enough to ignore the communication delay. In this manner, the number of integrals in the proposed algorithm will decrease to $O(n)$, which makes the computation fast enough for the real-time application.

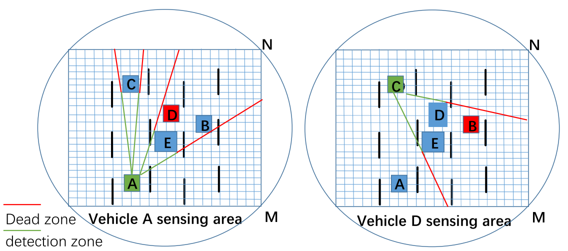

FIGURE 2. OGF map of vehicle A and C.

C. ACCURACY ENHANCEMENT

As mentioned before, accumulative sensing task is also considered by us to improve the sensing accuracy. OGF easily maps environments as occupancy states. Autonomous vehicles can be assisted with object tracking, localization, and route prediction. These grid maps are measured by sensors data. The occupancy grid map of vehicle A and C is shown in FIGURE 2. The accumulation of the OGF map of a single vehicle could enhance the cooperative sensing accuracy. The occupancy grid filtering map of V_k is defined as $C^t{}_{i_{m\times n}}$, is shown as below:

$$C^t{}_{k_{m\times n}} = \begin{bmatrix} c^t{}_{k_{1,1}} & c^t{}_{k_{1,2}} & \cdots & c^t{}_{k_{1,n}} \\ c^t{}_{k_{2,1}} & c^t{}_{k_{2,2}} & \cdots & c^t{}_{k_{2,n}} \\ \vdots & & \ddots & \\ c^t{}_{k_{m,1}} & c^t{}_{k_{m,2}} & \cdots & c^t{}_{k_{m,n}} \end{bmatrix}$$

where, $c_k{}^t{}_{i,j}$ is the particle projected on the gird $g_{i,j}$ from the on board sensors of vehicle k at time t. A particle in this paper refers to 3D data measured from sensors, i.e., a particle in a LIDAR sensor is the reflectance of a beam, and a particle in a depth map extracted from stereo vision data refers to a depth pixel. We suppose each autonomous vehicles in the platoon covered by a head vehicle can upload its OGF map in a synchrony time stamp. The head vehicle accumulates occupancy grid filtering map of superpixel-based clustering particles as $C^t{}_{m\times n}$. $C^t{}_{m\times n}$ is shown as follows:

$$C^t{}_{m\times n} = \sum_{k=1}^{S} C^t{}_i \tag{10}$$

We use SVM algorithm to classify whether the grids of OGF map are occupied by a real vehicle. The SVM algorithm is a fast and reliable method for dichotomy of whether the grid is occupied by a vehicle.The grid state of a vehicle can be depicted as follow:

$$\begin{cases} \max\limits_{w,b} \frac{2}{||w||}; \\ s.t. g_{i,j}(wC_{i,j} + b) \geq 1; i = 1, 2, \ldots, n \end{cases} \tag{11}$$

The purpose is to find the optimal verdict field (optimal w and b) to fuse the OGF maps into a high accurate projection of vehicle states. Then we train our SVM algorithm by the real traffic data to get optimal. Finally, we can get a full render of all vehicles' states $G_{m\times n}$ within an occupancy grid map fused by the head vehicle group with SVM algorithm. The definition of the final output $G_{m\times n}$ follows:

$$G_{m\times n} = \begin{bmatrix} g_{1,1} & g_{1,2} & \cdots & g_{1,n} \\ g_{2,1} & g_{2,2} & \cdots & g_{2,n} \\ \vdots & & \ddots & \\ g_{m,1} & g_{m,2} & \cdots & g_{m,n} \end{bmatrix}$$

where $g_{i,j} = 0$ means that $g_{i,j}$ is not occupied by a vehicle and $g_{i,j} = 1$ means that $g_{i,j}$ is occupied by a vehicle.

D. TRAINING OF SVM ALGORITHM

In order to simulate the situations of multi-vehicle sensing, we extract the snapshot (frame) of the real traffic data NGSIM I-80 and US 101. In order to execute the process of multi-vehicle sensing, we should locate each single autonomous vehicle. We extract the Global Positioning System (GPS) data from the NGSIM. It is well-known that the GPS data is inaccurate. Luckily, our cooperative sensing approach can take advantage of the high precision of the sensors to revise the GPS error [33]. We assume that each autonomous vehicle equips with the same 360° LIDAR and stereo vision sensor in a single snapshot. According to the linear propagation characteristic of the on-board sensors, we define $c_{i_{i,j}} = 1+\delta$ is the occupancy grid filtering map of superpixel-based clustering particles of V_i on a unite single grid. The $\delta \in [-0.5, 0.5]$ is additive system inherent error. δ follows a normal distribution with a mean of 0 and a variance of p. In our study we set $p = 0.01$. The size of a single grid is set as 0.2 meters. Therefore, each frame is divided into a occupancy map with the granularity of 0.2 meters. Further, to model our OGF faster, we make each vehicle has the same size which is 2 meters wide and 5 meters long. The GPS data of NGSIM is to locate the vehicles roughly into OGF map. We assume that the GPS data are the geometric center of a vehicle, hence we can get a OGF map based on GPS. This map of single vehicle at time t can be depict as ${R^t}_{i_{m\times n}}$ which has the same structure as ${C^t}_{i_{m\times n}}$:

$$
{R^t}_{k_{m\times n}} = \begin{bmatrix} {r^t}_{k_{1,1}} & {r^t}_{r_{1,2}} & \cdots & {r^t}_{k_{1,n}} \\ {r^t}_{k_{2,1}} & {r^t}_{r_{2,2}} & \cdots & {r^t}_{k_{2,n}} \\ \vdots & & \ddots & \\ {r^t}_{k_{m,1}} & {r^t}_{r_{m,2}} & \cdots & {r^t}_{k_{m,n}} \end{bmatrix}
$$

where ${r^t}_{k_{m,n}}$ is the GPS data extract from each frame projected to grid (m, n). If there is no GPS data projected into the grid, we define ${r^t}_{k_{m,n}} = 0$. Otherwise, ${r^t}_{k_{m,n}} = 1$ for a 5 meters wide and 10 meters long range due to the low accuracy of GPS data and the geometric center of this range is extract from the GPS data which is transferred into the relative position of the occupancy grid map. Similarly, the head vehicle accumulates the OGF maps of GPS as ${R^t}_{m\times n}$. Finally, we set the input as $I^t = {R^t}_{m\times n} + {C^t}_{m\times n}$. The SVM algorithm is effectively revised the cumulative perceptual results. We extract 10-s (100) single frame to train the SVM algorithm, the head vehicle is appointed as the vehicle which is closet to the camera in each frame. The definition of head vehicle is as follows:

$$
\begin{cases} V_{head} = V_i \\ i = \min\limits_i(\sqrt{{X^i}_{local}{}^2 + {Y^i}_{local}{}^2}) \end{cases} \tag{12}
$$

Finally, we train our SVM algorithm in the following. The end-to-end output is $O_{m\times n}$ which is an accurate OGF map of our proposed approach. And because of the light data size of the OGF map and low computational complexity of the SVM algorithm, the computation task is processed in the head vehicle due to its powerful computation ability. In addition, we sample the geometric center $[X_i, Y_i]$ of the V_i in the OGF map. The $[X_i, Y_i]$ can be set as the input of our Li-GRU neural network algorithm.

V. LANE CHANGE PREDICTION

The neural network is a promising method for trajectory prediction to enhance driving safety. In order to satisfy the high prediction accuracy of autonomous driving, we propose a Li-GRU neural network for lane change detection. Furthermore, the training of the neural network is a high computational task and it could take a long time. Based on our VFC architecture, the training task can be offloaded to the intelligent vehicles which enable the distributed learning to reduce the training time significantly.

A. LIGHT GRU ALGORITHM FOR LANE CHANGE PREDICTION

Deep Neural Network (DNN) algorithm is an effective method for vehicle trajectory prediction, especially the LSTM and GRU algorithms which perform well for the temporal and spacial characteristics. At the same time, the accuracy and the computational complexity is very important for DNN algorithm. Although it is proved that GRU algorithm is a light-weight method for prediction compared with LSTM algorithm, which lowers the computational complexity, there still has considerable room for the improvement of accuracy and lower complexity. The Li-GRU algorithm is a modified GRU algorithm which removes the reset gate, replacing the hyperbolic tangent function with the ReLU activation, and applying batch normalization has better performance. Li-GRU algorithm is shown in the following model:

$$
\begin{aligned} z_t &= \sigma(BN(W_z x_t) + U_z h_{t-1} + b_z), && (13) \\ \widetilde{h}_t &= ReLU(BN(W_h x_t) + U_h h_{t-1} + b_h), && (14) \\ h_t &= z_t \odot h_{t-1} + (1 - z_t) \odot \widetilde{h}_t. && (15) \end{aligned}
$$

The batch normalization $BN(\cdot)$ is defined as follows:

$$
BN(a) = \gamma \odot \frac{a - mu_b}{\sqrt{\sigma_b^2 + \epsilon}} + \beta \tag{16}
$$

where μ_b and σ_b are the mini-batch mean and variance.ϵ is added for numerical stability. The variables γ and β are trainable scaling and shifting parameters, introduced to restore the network capacity. The presence of β makes the biases b_h and b_z redundant are omitted in Eq. (13) and (14).

Lane changing manoeuvres plays an essential role in traffic flows and autonomous vehicles theory. Therefore, we discuss the lane changing manoeuvres for our deep learning empowered cooperative sensing. We consider that the target vehicle which intends to change lane as V_{tar} and the vehicles in front and rear of the target vehicle in the current lane and the neighbour lanes are defined as surrounding vehicles. $V_1, V_2, \ldots, V_i$ are the surrounding vehicles. The trajectory of theses vehicles are ${x^t}_{tar}$ and ${x^t}_1$, ${x^t}_2$,…, ${x^t}_s$. The end-to-end input and output are defined as $x^t = [{x^t}_{tar}, {x^t}_1, {x^t}_2, \ldots, {x^t}_s]$ and the output $\mathbf{y}_s^t = [0, 1]$. The label of lane-keeping is 0 (Positive), and the label of lane changing behaviour is 1(Negative). The details of x^t are as follow:

$$
\begin{cases} \mathbf{x}_{tar}^t = [\Delta X_{tar}^{(t-t_h+\Delta t)}, \Delta Y_{tar}^{(t-t_h+\Delta t)}, \ldots, \Delta X_{tar}^t, \Delta Y_{tar}^t] \\ \mathbf{x}_i^t = [\Delta X_i^{(t-t_h+\Delta t)}, \Delta Y_i^{(t-t_h+\Delta t)}, \ldots, \Delta X_i^t, \Delta Y_i^t] \end{cases} \tag{17}
$$

$$
\mathbf{y}_s^t = [0, 1] \tag{18}
$$

t_h is the historical time horizon and t_p is the prediction time horizon. $[\Delta X_{tar}^t, \Delta Y_{tar}^t]$ and $[\Delta X_i^t, \Delta Y_i^t]$ are the position displacements of V_{tar} and V_i from the time $t - \Delta t$ to t.

Finally, the training process of our Li-GRU neural network is allocated to the intelligent vehicles in our autonomous platoon to perform as distributed learning. The train time can reduce significantly by our VFC architecture.

B. TRAINING OF LI-GRU ALGORITHM

As mentioned before, we only consider the lane changing situations. There are two types of lane changing: mandatory lane

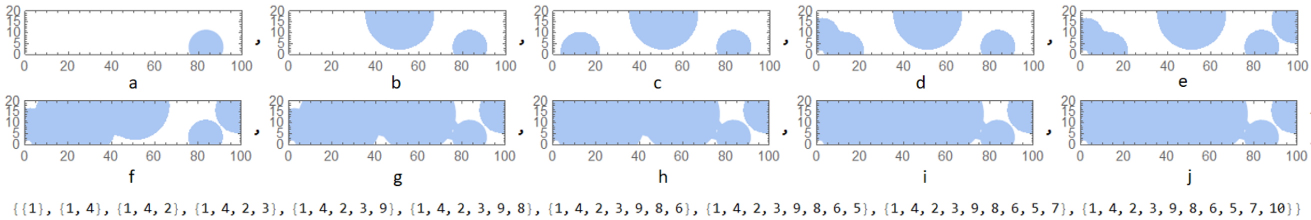

FIGURE 3. Greedy Vehicle selection process.

TABLE 1. Different Li-GRU Algorithm Structure.

Structure	Hidden Layer		
	1	2	3
1	20	0	0
2	50	0	0
3	200	0	0
4	20	20	0
5	50	50	0
6	10	10	10
7	40	40	40

changing (MLC) discretionary lance changing (DLC) [34]. MLC means that the driver is forced to change from the current lane, due to the on-ramp or off-ramp situations. A DLC happens when a driver is not satisfied with the situation of the current lane and intends to change to a neighbour lane. Compared with MLC, DLC is more likely to cause dangerous. Hence, we make the data extracting to filter the irrelevant information:

- We extract the data from the middle lane(lane 2, 3, 4, 5) which is concerned about DLC.
- Excluding abnormal lane changing behaviours which include multiple lane changing and non-neighbour lane changing.
- In order to contain the historical experience of vehicle-to-vehicle effect, we extract the data of 5-S interval (50 frames) before the lane changing happens.
- The frame we extracted must be continuous to avoid sensor occlusion.

We extract 453 DLC vehicles from the NGSIM I-80 data and US 101 data to train our Li-GRU neural network algorithm. We set 300 DLC vehicles for target vehicle as to the train sets and extract 5-s (50 frames) historic data of each target vehicle and the rest 153 as the validation sets. Therefore, the $\Delta t = 0.1$ and $t_h = 5$. To avoid the general machine learning problem, over-fitting should be paid special attention during the training process. The previous researches had proved that a medium sample size and a simpler NN structure can get satisfactory results [35]. Accordingly, the 300 DLC target vehicles are randomly sampled from the 453 DLC dataset. 20 vehicles will be selected as the cross-validation set.

In order to find the optimal structure of our Li-GRU algorithm model, we test different structures based on our cross-validation samples. TABLE 1 shows the different structures of the Li-GRU neural network algorithm. The results show that the Li-GRU algorithm model obtains the best assessment performance with Structure 4 (two hidden layers that contain 20 neurons, respectively) in TABLE 1. Therefore, we adopt Structure 4 Li-GRU algorithm for its best performance.

Moreover, based on our VFC architecture, we offloading the train set to the intelligent vehicles in our autonomous platoon. Therefore, the deep neural network can be trained in parallel processing [36].

VI. PERFORMANCE EVALUATION

In this section, we provide the simulation results of our proposed VFC architecture for cooperative sensing of autonomous platoon driving. First, we evaluate the cooperative sensing coverage. Then, the results of the fusion accuracy of our SVM algorithm with accumulation OGF map are shown. Finally, we reveal the performance of our proposed Li-GRU algorithm for lane changing prediction. For comparison, we also present the results of several popular neural network algorithms for lane changing.

A. SIMULATION RESULTS OF COVERAGE ENHANCEMENT

To simulate our proposed VFC architecture for autonomous driving, we consider the communication coverage of the head vehicle is 100 meters. Therefore, we focus on a four-lane road that's 100 meters long and 20 meters wide. The intelligent vehicles in this communication coverage of the head vehicle can connect to the head vehicle. We set the sensing coverage of a single vehicle s_i range randomly from 5 meters to 20 meters due to the effective precision of onboard sensors [37], [38]. In the real world, the distribution of vehicles varies. To simulate a general situation, we set the N vehicles are located randomly in this area. The greedy vehicle selection process is shown in FIGURE 3 from a to j. We set 10 vehicles in our autonomous platoon. Besides, the vehicle id is shown at the bottom of FIGURE 3.

Moreover, to simulate the real situations of the autonomous platoon, we set the different sizes of our platoon. Furthermore, we process 10 times of simulation for each size to eliminate the particularity of a single experiment. The efficiency of sensing coverage is shown in FIGURE 4. When $N = 10$

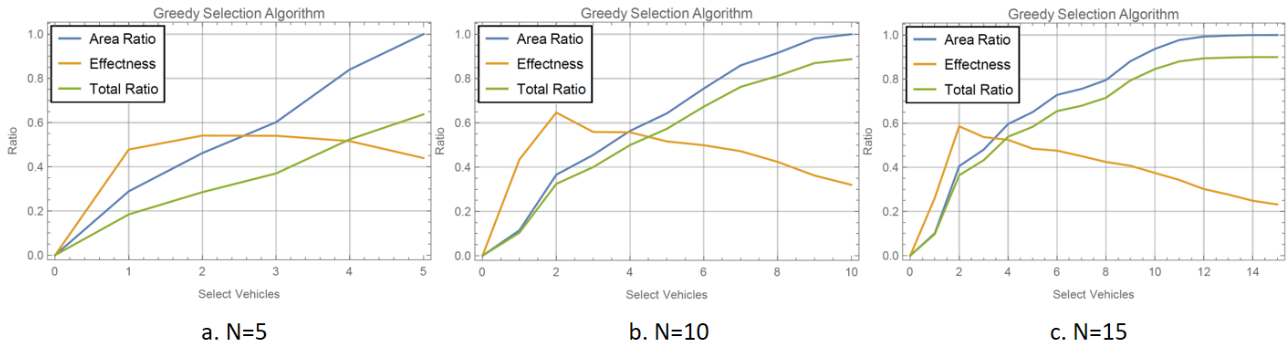

FIGURE 4. **The efficiency of cooperative sensing coverage.**

with the greedy vehicles select algorithm, the *AreaRatio* and the *TotalRatiol* show an increasing trend and converge at 9 vehicles. This is because with more vehicles are selected, the cooperative sensing coverage is enhanced and when the number of vehicles reaches a certain amount, the additional cooperative sensing coverage of vehicles is covered by the previous vehicles. The *Effectness* grows up to a peak, then gradually reduces due to there was no overlap of the sensing coverage of the first several vehicles and with more and more vehicles are selected, the overlap is growing. Besides, compared with $N = 5$ and $N = 15$ we can draw some conclusions. Although the *Effectness* performance well When $N = 5$, the *TotalRation* is much lower than $N = 10$ and $N = 15$ due to the platoon size is small. FIGURE 4 b. and FIGURE 4 c. both have a good *TotalRation* and *Effectness* when the platoon size is larger than 10 the *Effectness* could be more than 90% which means that the cooperative sensing can cover nearly all the surroundings of the autonomous driving platoon.

B. SIMULATION RESULTS OF ACCURACY ENHANCEMENT

We trained the SVM algorithm with 100 frames data. The frame ID is from 800 to 900 of the NGSIM data. Then we select frame 15 to reveal the performance. To simulate the autonomous platoon driving we make the vehicles presented in all 100 frames as the autonomous vehicle because this extraction matches the proposed principle of autonomous platoon: almost the same speed and area. The others are defined as social vehicles. The absolute position error means the absolute destination between the geometric center of the output of our SVM algorithm and the real-traffic data. We compared three different levels of fusion by choosing different numbers of fusion vehicles. FIGURE 5 shows that our proposed VFC architecture for multi-vehicle sensing fusion has a very considerable improvement for accuracy. Almost all dark blue bars are less than 1 meter which is much lower than the yellow bars. This is because the origin locating of GPS is very rough and our multi-vehicle sensing approach can highly improve the accuracy of locating. Moreover, the value of green bars is between the blue bars and yellow bars. This is since the

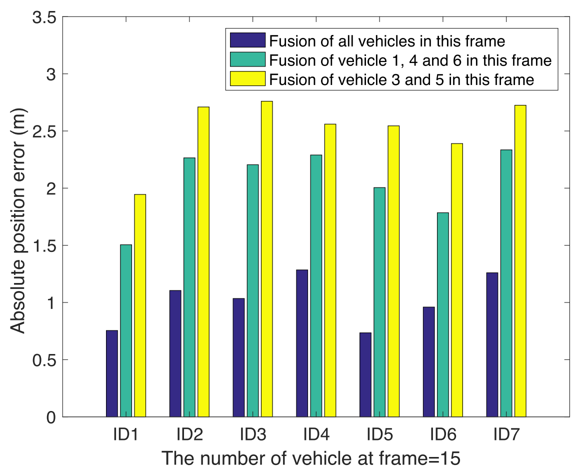

FIGURE 5. **SVM algorithm for cooperative sensing accuracy.**

sensing accuracy increases with the size of the autonomous platoon.

C. PERFORMANCE OF LI-GRU NEURAL NETWORK ALGORITHM

In this subsection, we compared the train time of the proposed model with different offloading strategies. Based on our VFC architecture, we compared two structures of the Li-GRU training which are centralized and distributed. The centralized structure means that the training is processed in the head vehicle of our autonomous platoon. Accordingly, the distributed structure means that the training is offloading to the vehicles in our platoon. As mentioned before, the training dataset size is 300. We assume that $N = 10$. Therefore, the distributed training dataset size is 30. The results are shown in TABLE 2. We can see that our proposed VFC architecture can significantly reduce the training time.

Then we compare Li-GRU neural network algorithm with several existing models in terms of accuracy. As mentioned, we use NGSIM I-80 data and US 101 data to compare the following models: LSTM, GRU and Li-GRU algorithms. The detailed algorithm procedure and model introduction can be

TABLE 2. **Training Time of Different Structure.**

Epochs	Structure	
	Centralised	Distributed
20	11s	3s
30	16s	5s
60	32s	9s

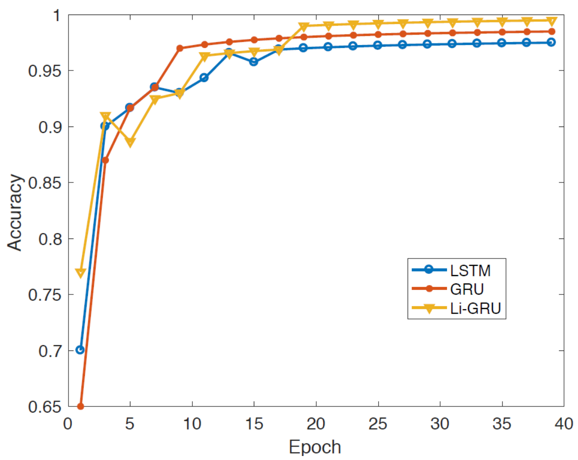

FIGURE 6. **Prediction accuracy of neural network algorithms.**

found in their references [30] and [39]. As we know, accuracy is the most important metric of the neural network algorithm. As shown in FIGURE 6, after 40 epochs Li-GRU algorithm has an accuracy of 98.9 % compared with GRU algorithm 97.7% and LSTM algorithm 96.9% which is proved the best one.

VII. CONCLUSION

In order to overcome the weakness of traditional VFC architecture, we proposed a new VFC architecture for cooperative sensing in a platoon, which can jointly utilize the sensing and the computational abilities of intelligent vehicles. Based on our VFC architecture, we adopt greedy and SVM algorithms respectively to enhance the sensing coverage and accuracy in the platoon. Besides, a distributed Li-GRU neural network is leverage for lane change detection. The real-world traffic dataset NGSIM is used to verify our model. The results show that the sensing coverage ratio can reach more than 90%. The sensing accuracy can reduce a lot. Also, the distributed Li-GRU neural network can perform a 98.9% accuracy for lane change prediction. Meanwhile, our VFC architecture enables the distributed learning to reduces the computation computing time. However, there remain some problems to be discussed. The train data we choose for our model is limited, which is not suitable for all situations. The scheduling of communication, computation, sensing, and vehicle running conditions should be in synthetic consideration. These will be the emphasis of our research in the future.

REFERENCES

[1] S. Campbell, N. O'mahony, L. Krpalcova, D. Riordan, J. Walsh, A. Murphy, and C. Ryan, "Sensor technology in autonomous vehicles: A review," in *Proc. 29th Irish Signals Syst. Conf. (ISSC)*, Jun. 2018, pp. 1–4.

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

[3] D. Jia, K. Lu, J. Wang, X. Zhang, and X. Shen, "A survey on platoon-based vehicular cyber-physical systems," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 263–284, 1st Quart., 2016.

[4] X. Chen, S. Leng, Z. Tang, K. Xiong, and G. Qiao, "A millimeter wave-based sensor data broadcasting scheme for vehicular communications," *IEEE Access*, vol. 7, pp. 149387–149397, 2019.

[5] J. Zhao, S. Ni, L. Yang, Z. Zhang, Y. Gong, and X. You, "Multiband cooperation for 5G HetNets: A promising network paradigm," *IEEE Veh. Technol. Mag.*, vol. 14, no. 4, pp. 85–93, Dec. 2019.

[6] Z. Zhou, C. Gao, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Social big-data-based content dissemination in Internet of vehicles," *IEEE Trans. Ind. Informat.*, vol. 14, no. 2, pp. 768–777, Feb. 2018.

[7] X. Sun, J. Zhao, X. Ma, and Q. Li, "Enhancing the user experience in vehicular edge computing networks: An adaptive resource allocation approach," *IEEE Access*, vol. 7, pp. 161074–161087, 2019.

[8] J. Feng, Z. Liu, C. Wu, and Y. Ji, "AVE: Autonomous vehicular edge computing framework with ACO-based scheduling," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 10660–10675, Dec. 2017.

[9] Z. Zhou, H. Liao, B. Gu, K. M. S. Huq, S. Mumtaz, and J. Rodriguez, "Robust mobile crowd sensing: When deep learning meets edge computing," *IEEE Netw.*, vol. 32, no. 4, pp. 54–60, Jul. 2018.

[10] H. Guo, J. Liu, and J. Zhang, "Computation offloading for multi-access mobile edge computing in ultra-dense networks," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 14–19, Aug. 2018.

[11] L. Liu, C. Chen, Q. Pei, S. Maharjan, and Y. Zhang, "Vehicular edge computing and networking: A survey," Jul. 2019, *arXiv:1908.06849*. [Online]. Available: https://arxiv.org/abs/1908.06849

[12] J. Zhao, Q. Li, Y. Gong, and K. Zhang, "Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7944–7956, Aug. 2019.

[13] X. Chen, S. Leng, K. Zhang, and K. Xiong, "A machine-learning based time constrained resource allocation scheme for vehicular fog computing," *China Commun.*, vol. 16, no. 11, pp. 29–41, Nov. 2019.

[14] Q. Li, J. Zhao, and Y. Gong, "Computation offloading and resource allocation for mobile edge computing with multiple access points," *IET Commun.*, vol. 13, no. 17, pp. 2668–2677, Oct. 2019.

[15] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen, "Vehicular fog computing: A viewpoint of vehicles as the infrastructures," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 3860–3873, Jun. 2016.

[16] A. V. Dastjerdi and R. Buyya, "Fog computing: Helping the Internet of Things realize its potential," *Computer*, vol. 49, no. 8, pp. 112–116, Aug. 2016.

[17] K. Xiong, S. Leng, J. Hu, X. Chen, and K. Yang, "Smart network slicing for vehicular fog-RANs," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3075–3085, Apr. 2019.

[18] NGSIM. *Next Generation Simulation (NGSIM) Vehicle Trajectories and Supporting Data*. Accessed: Oct. 30, 2019. [Online]. Available: https://data.transportation.gov/Automobiles/Next-Generation-Simulation-NGSIM-Vehicle-Trajector/8ect-6jqj/

[19] C. Zhu, G. Pastor, Y. Xiao, Y. Li, and A. Ylae-Jaeaeski, "Fog following me: Latency and quality balanced task allocation in vehicular fog computing," in *Proc. 15th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, Jun. 2018, pp. 1–9.

[20] G. Qiao, S. Leng, S. Maharjan, Y. Zhang, and N. Ansari, "Deep reinforcement learning for cooperative content caching in vehicular edge computing and networks," *IEEE Internet Things J.*, to be published.

[21] Y. Xiao and C. Zhu, "Vehicular fog computing: Vision and challenges," in *Proc. IEEE Int. Conf. Pervas. Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2017, pp. 6–9.

[22] L. Yang, H. Zhu, H. Wang, H. Qian, and Y. Yang, "Incentive propagation mechanism of computation offloading in fog-enabled D2D networks," in *Proc. IEEE 23rd Int. Conf. Digit. Signal Process. (DSP)*, Nov. 2018, pp. 1–4.

[23] Z. Su, Q. Xu, Y. Hui, M. Wen, and S. Guo, "A game theoretic approach to parked vehicle assisted content delivery in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 7, pp. 6461–6474, Jul. 2017.

[24] J. Zhao, X. Guan, and X. P. Li, "Power allocation based on genetic simulated annealing algorithm in cognitive radio networks," *Chin. J. Electron.*, vol. 22, no. 1, pp. 177–180, 2013.

[25] Z. Junhui, Y. Tao, G. Yi, W. Jiao, and F. Lei, "Power control algorithm of cognitive radio based on non-cooperative game theory," *China Commun.*, vol. 10, no. 11, pp. 143–154, Nov. 2013.

[26] Y. Li, D. Duan, C. Chen, X. Cheng, and L. Yang, "Occupancy grid map formation and fusion in cooperative autonomous vehicle sensing," in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Dec. 2018, pp. 204–209.

[27] R. O. Chavez-Garcia and O. Aycard, "Multiple sensor fusion and classification for moving object detection and tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 525–534, Feb. 2016.

[28] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[29] S.-I. Oh and H.-B. Kang, "Fast occupancy grid filtering using grid cell clusters from LIDAR and stereo vision sensor data," *IEEE Sensors J.*, vol. 16, no. 19, pp. 7258–7266, Oct. 2016.

[30] X. Liu, J. Liang, and B. Xu, "A deep learning method for lane changing situation assessment and decision making," *IEEE Access*, vol. 7, pp. 133749–133759, 2019.

[31] H. Iwaki, G. Srivastava, A. Kosaka, J. Park, and A. Kak, "A novel evidence accumulation framework for robust multi-camera person detection," in *Proc. 2nd ACM/IEEE Int. Conf. Distrib. Smart Cameras*, Sep. 2008, pp. 1–10.

[32] B. Guo, H. Chen, Z. Yu, X. Xie, S. Huangfu, and D. Zhang, "FlierMeet: A mobile crowdsensing system for cross-space public information reposting, tagging, and sharing," *IEEE Trans. Mobile Comput.*, vol. 14, no. 10, pp. 2020–2033, Oct. 2015.

[33] A. El Abbous and N. Samanta, "A modeling of GPS error distributions," in *Proc. Eur. Navigat. Conf. (ENC)*, May 2017, pp. 119–127.

[34] A. Talebpour, H. S. Mahmassani, and S. H. Hamdar, "Modeling lane-changing behavior in a connected environment: A game theory approach," *Transp. Res. Procedia*, vol. 7, pp. 420–440, 2015.

[35] L. Huang, H. Guo, R. Zhang, H. Wang, and J. Wu, "Correction to 'capturing drivers' lane changing behaviors on operational level by data driven methods,'" *IEEE Access*, vol. 7, p. 71423, 2019.

[36] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," 2016, *arXiv:1602.05629*. [Online]. Available: https://arxiv.org/abs/1602.05629

[37] J. Kim, D. S. Han, and B. Senouci, "Radar and vision sensor fusion for object detection in autonomous vehicle surroundings," in *Proc. 10th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2018, pp. 76–78.

[38] Y. Zeng, Y. Hu, S. Liu, J. Ye, Y. Han, X. Li, and N. Sun, "RT3D: Real-time 3-D vehicle detection in LiDAR point cloud for autonomous driving," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3434–3440, Oct. 2018.

[39] S. Dai, L. Li, and Z. Li, "Modeling vehicle interactions via modified LSTM models for trajectory prediction," *IEEE Access*, vol. 7, pp. 38287–38296, 2019.

HAO DU received the B.Sc. degree in information and communication engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2016, where he is currently pursuing the Ph.D. degree. His research interests include mobile edge computing, the Internet of Vehicles, and artificial intelligence in wireless networks.

SUPENG LENG received the Ph.D. degree from Nanyang Technological University (NTU), Singapore. He is currently a Full Professor and the Vice Dean with the School of Information and Communication Engineering, University of Electronic Science and Technology of China (UESTC). His researches focus on the Internet of Things, vehicular networks, and the next generation mobile networks. He serves as an Organizing Committee Chair and TPC member for many international conferences, as well as a reviewer for over ten international research journals.

FAN WU received the Ph.D. degree from the University of Electronic Science and Technology of China, China, in 2015. He is currently an Associate Professor with the School of Communication and Information Engineering, University of Electronic Science and Technology of China. His research interests include broadband wireless access networks, vehicular networks, and wireless sensor networks.

XIAOSHA CHEN received the B.Sc. degree in information and communication engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2016, where he is currently pursuing the Ph.D. degree. His research interests include handling routing protocols, and performance analysis and congestion control in vehicular networks.

SUN MAO is currently pursuing the Ph.D. degree with the School of Information and Communication Engineering, University of Electronic Science and Technology of China. His research interests include simultaneous wireless information and power transfer, mobile edge computing, intelligent reflecting surface, and cellular-connected UAV communications.