Received December 26, 2018, accepted January 5, 2019, date of publication January 24, 2019, date of current version February 14, 2019.
Digital Object Identifier 10.1109/ACCESS.2019.2894906

CNVPS: Cooperative Neighboring Vehicle Positioning System Based on Vehicle-to-Vehicle Communication

SANGHYUCK NAM, DANHEE LEE, JAEHWAN LEE, AND SANGOH PARK
School of Computer Science and Engineering, Chung-Ang University, Seoul 06974, South Korea
Corresponding author: Sangoh Park (sopark@cau.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) through the Korean Government (MSIT) under Grant NRF-2017R1C1B5075856.

ABSTRACT One of the key components of the intelligent transport system is to provide a safe driving environment. The ability to quickly and accurately recognize the surrounding environments and neighboring vehicles is essential for the development of safety-related services to provide a safe driving environment. Currently, research in this field is being conducted based on the global positioning system (GPS), light detection and ranging, camera, and ranging sensors. However, the currently used sensors cannot recognize a wide range of vehicles because of their limited range of recognition. Moreover, GPS-based studies are highly affected by the surrounding environment because of the nature of GPS and have relatively high error rates and low rate of location information updates. The use of GPS-based location information for safety-related services can result in negative consequences. In this paper, we propose a new positioning system, called cooperative neighboring vehicle positioning system (CNVPS). The CNVPS rapidly identifies the locations of neighboring vehicles based on their information obtained through various sensors and shares this information with a wide range of neighboring vehicles over vehicle-to-vehicle communications. The CNVPS also compensates the position of a neighboring vehicle by applying the maximum likelihood estimation to the duplicated position observation of the other neighboring vehicles. The simulation results show that the CNVPS achieves approximately 370% improvement in location errors over GPS, assuming that the root-mean-squared error of GPS is 15 m. In addition, the proposed system has a location refresh cycle ten times faster than the existing GPS-based system.

INDEX TERMS Cooperative neighboring vehicle positioning system (CNVPS), intelligent systems, global positioning system.

I. INTRODUCTION

The most important factor in recent research on intelligent transportation system (ITS) is to consider the driver's safety while driving [1]. Services and applications developed to ensure driver's safety could prevent collision and prepare for dangerous situations in advance based on the location of neighboring vehicles. As most car accidents are affected by the location of neighboring vehicles, the rapid and accurate prediction of the position of neighboring vehicles is essential for the safety of the vehicles [2]. In addition to the safety-related services, it is crucial to recognize neighboring vehicles in the self-driving environment, such as the inter-vehicle distance maintenance service and low-speed auto-pilot services. With the growing importance of the recognizing neighboring vehicles, many researchers considered the use of Global Navigation Satellite Systems [1], [2], such as the Global Positioning System (GPS) [1], [2], with dedicate short-range communication (DSRC) [1], [2] to transfer vehicle's location information on the vehicular AdHoc network (VANET) to achieve recognition of neighboring vehicles [3]–[5] are actively conducted. However, GPS-based system used in majority of existing studies has a slow refresh cycle owing to its high computational demands, limitations of commercial GPS, and high error rate of location information due to unexpected signal attenuation and performance of the receiving terminals as it relies on satellite signals to receive location information. With these limitations, the GPS-based system is not suitable for use in safety-related services as it requires accurate location information. Moreover, positioning errors in GPS can lead to extremely inaccurate results [6] because of the multipath problem caused by diffraction or reflection of signals by buildings in urban environment. Further, this system suffers from discontinued location update in disconnected situations such as tunnels. GPS may be an effective method for simple services, such as turn-by-turn navigation; however, the use of GPS-based location information is inappropriate in vehicle safety application, given that the suggested requirements for vehicle-safety applications in the Vehicle Safety Communications Consortium's report [7] are below 50–100 ms. Further, although the use of ranging sensors, such as light detection and ranging (LiDAR) will allow for a faster and more accurate recognition of neighboring vehicles [8] owing to sensor characteristics, recognizing a vehicle in front of a large object is difficult. In addition, LiDAR is limited to recognizing vehicles over a limited range.

Recently, owing to the development of camera performance and Computer Vision, identification of license plates by using an automatic number-plate recognition system [9] is possible; various vehicular environment information, such as vehicle-type recognition [10] and lane recognition [11], can be obtained. However, as a single sensor system, there are limits to the distance between camera and the object to be recognized owing to the nature of the system [9].

To solve these problems, we propose a new vehicle positioning system that can quickly and accurately recognize a wide range of vehicles with information received from front camera, a ranging sensor, a GPS, and an inertial sensor. The proposed system can extend the recognition range by sharing location information of the neighboring vehicle through vehicle-to-vehicle (V2V) communication. In addition, it can cooperatively compensate for the location error by using the location information of the neighboring vehicle observed by other vehicles. In addition, this system allows the limited tracking of vehicle not equipped with sensors and communication devices in presence of fully-equipped vehicles.

The remainder of the paper is organized as follows. In Section 2, we summarize the problem of existing vehicle positioning systems. In Section 3, we describe the proposed CNVPS. In Section 4, we present the simulation results, and finally Section 5 concludes our work.

II. RELATED WORK

Previous research about ITS mainly utilizes vehicle-to-vehicle communication to disseminate information from various sensors to neighboring vehicles [3]–[5], [12]–[14]; the most representative method is to use the location information obtained from the GPS sensor mounted on the vehicle. However, the accuracy of the GPS location information that can be used in the public domain has an error of 8 m on average [15]. In a complex environment, such as a tunnel or an urban area with skyscrapers, signal attenuation and multipath problems can occur, exhibiting a high error rate in GPS location information, which subsequently cannot be used. Therefore, studies have been conducted to accurately locate neighboring vehicles by complementing the problems of the GPS. To improve the accuracy of GPS, differential GPS (DGPS) was proposed that utilizes a fixed-position GPS receiver to measure and correct the error between moving GPS receivers [16]. In addition, the application of the Data Fusion technique (Kalman Filter, Particle Filtering) to multiple sensors has been proposed to measure distances and improve the accuracy of GPS. However, these aforementioned methods do not solve problems of signal attenuation or multipath, which are limitations of GPS itself. Moreover, they require additional infrastructure and equipment, such as a ground station, and a high temporal cost of computation is required to accurately correct errors. Thus, such methods are not suitable for safety-related applications requiring low latency and cycle to obtain location information.

To overcome environmental limitations, such as signal attenuation and multipath problems, the dead reckoning method [22] was proposed to estimate vehicular positioning by using velocity and direction at the initial position. However, as GPS coordination is used as the initial position, it is impossible to overcome GPS error; a large amount of error can accumulate, while new GPS location information is delayed.

In addition, studies have used Computer Vision to recognize a lane by comparing it with the stored lane images [19]. Further, Benslimane [2] modeled the current driving environment with inertial sensors and compared it with premade maps to recognize location; however, the accuracy of this method is low and the performance is influenced by the environment.

III. CNVPS

In this section, we describe CNVPS, a new method that can recognize the relative position of neighboring vehicles more accurately and quickly then existing methods.

CNVPS consists of three modules, i.e., network manager, cooperative positioning manager, and data manager, as shown in Fig. 1; the method is performed in three workflows, i.e., measurement, sharing, correction, as shown in Fig. 2.

A. NETWORK MANAGER

The network manager consists of a real-time broadcaster, which shares information of the neighboring vehicles of the current vehicle with all other vehicles in the communication range, and vehicle discovery, which discovers neighboring vehicles and measures their relative positions through sensors.

The real-time broadcaster module constructs and delivers a message for location-information sharing between systems installed in each vehicle through V2V communication. The message shared with each vehicle is termed basic safety message (BSM) [21], which is a standard message of SAE J2735. Additional information used in CNVPS is added to the optional part of the BSM. The message constructed by the real-time broadcaster consists of the current location

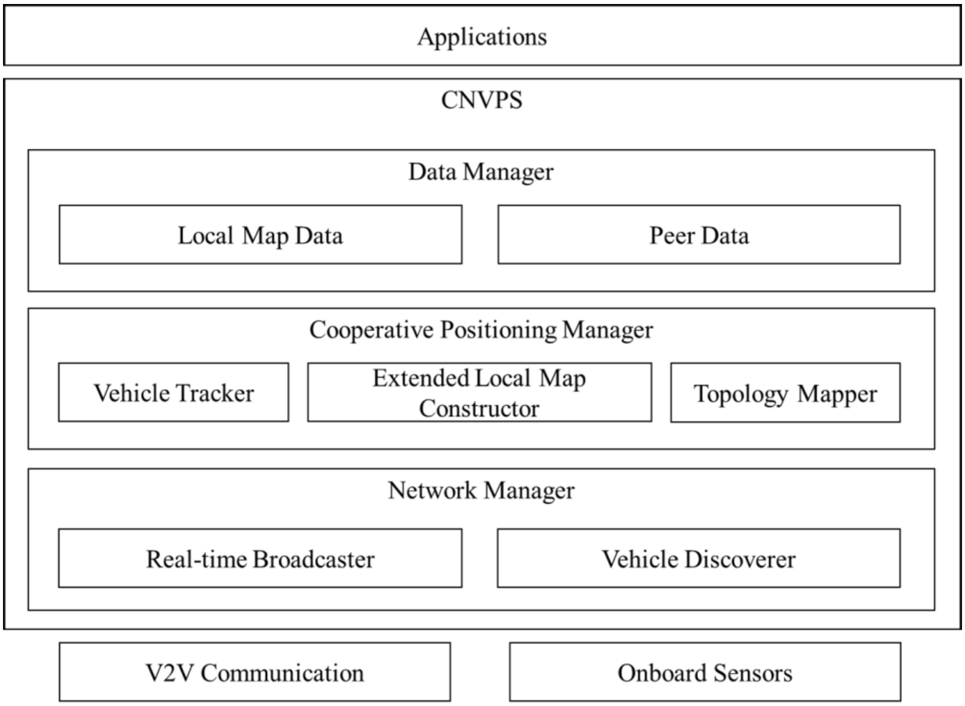

FIGURE 1. System architecture.

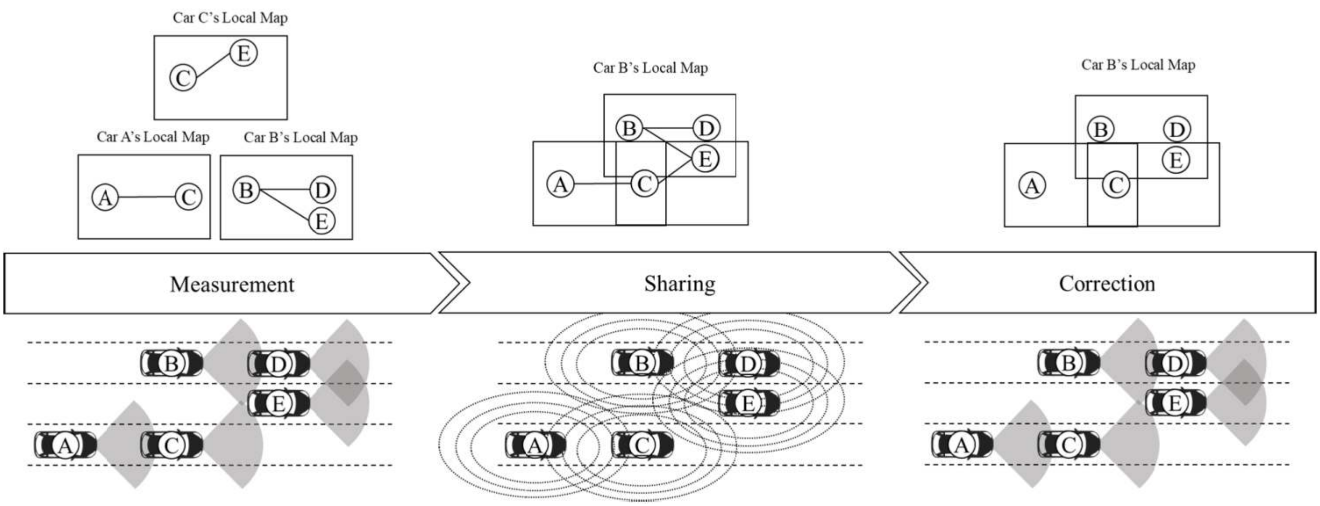

FIGURE 2. System workflow.

of the current vehicle in absolute coordinates, a local map containing relative positions of neighboring vehicles directly recognized by the vehicle, and a timestamp of the message, which is broadcasted every 100 ms (the time required by the vehicle safety application) to vehicles within the DSRC communication range. The position information and local map of the current vehicle are received from the cooperative position manager, and the position information and local map received from the neighboring vehicle are passed to the cooperative position manager.

The vehicle discoverer module collects position and kinematic information (speed vector) from onboard sensors consisting of GPS and inertial sensors. We utilizes GPS in spite of its limitations, due to it is one of the most practical way to gather absolute location information, and we use GPS in tandem with various sensors and methods to improve accuracy of GPS. The relative distance and angle of each vehicle are measured by the ranging sensor, and a unique identifier, relative distance, and angle of the vehicle are obtained through license-plate analysis. By using the obtained position information, position information of the current vehicle, kinematic information, and position information of neighboring vehicle are passed to the vehicle tracker.

Vehicle identification using number plate recognition was considered due to the following reasons. Although using the ranging sensors to recognize nearby vehicles and sharing this information to extend the recognition range can accurately recognize relative position, the GPS error complicates the identification of vehicles, thus complicating the integration of position information of the neighboring vehicles sent by the neighboring vehicles of the current vehicle. To solve this problem, existing studies have chosen to match the GPS position received from a neighboring vehicle with the vehicle position recognized by the ranging sensor. However, the accuracy of the GPS and environmental limitations can cause difficulty in identification.

In case of the driving scenario shown in Fig. 3 (a), vehicle A should recognize vehicular positioning by using information sent from other vehicles, as shown in Fig. 3 (b). However, when using GPS, vehicle A may confuse vehicles B and D owing to low accuracy in recognizing the location information, as shown in Fig. 3 (c).

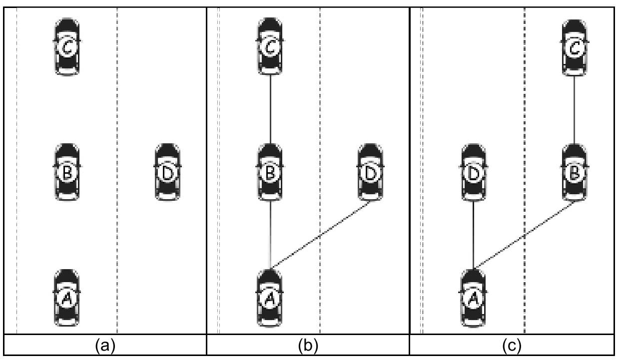

FIGURE 3. Vehicle topology estimation: (a) vehicle position; (b) correct topology; and (c) possible incorrect topology.

In this study, to solve this problem, we identified the physical vehicle by recognizing the license plate, which is the unique identifier, by using camera and integrating the relative position information measured using the ranging sensor, thereby preventing possible vehicle confusion.

B. COOPERATIVE POSITIONING MANAGER

The cooperative positioning manager constructs a local map consisting of the location information on neighboring vehicles based on the information received from the vehicle discovery module of the network manager, and expands and compensates the local map by using the local maps received from the neighboring vehicles. In addition, the cooperative positioning manager serves to compensate the position of the current vehicle by using the position information observed by neighboring vehicles. This manager consists of three submodules: vehicle tracker, extended local-map constructor, and topology mapper.

The vehicle tracker configures the latest location information of the neighboring vehicles received from the onboard sensors to a local map. In the sharing phase, the position information of the current vehicle, comprising GPS information and local map, are transmitted through the real-time broadcaster to neighboring vehicles, notifying them about the presence of the current vehicle. The local map consists of tuples that include ObserverID, indicating the vehicle that performed the observation; ObservedID, indicating the vehicle being observed; relative position, which is the relative position of the neighboring vehicle based on the observer vehicle; Sensor, indicating the type of the sensor that performed the observation; Sensor error, indicating the error of the sensor; and Timestamp, indicating the time at which the observation was last performed.

For example, Table 1 shows the local map generated by Vehicle A in Fig. 3.

TABLE 1. Local map of vehicle A.

ObserverID	ObservedID	Relative Position	Sensor	Sensor error	Timestamp
1(A)	2(B)	10.3, 13.2	GPS	10	1523969831
1(A)	2(B)	11.1, 10.1	Camera	1.5	1523969835
1(A)	4(D)	23, 43.2	GPS	8	1523969841

The period for broadcasting a message containing a local map (C_m) is 100 ms, the period for updating GPS (C_g) is 1000 ms, and the period for updating the kinematic information of the current vehicle from the sensor (C_s) is less than 100 ms; thus, $C_s < C_m < C_g$. Therefore, the vehicle tracker uses the kinematic information from the vehicle discoverer to update the location of the vehicle by using the dead reckoning method [29], until a message containing the next local map is received or the updated GPS information is received to provide the latest location information. Thus, in safety-related applications, the vehicle tracker can help make more precise decisions.

Moreover, when the vehicle tracker of the proposed system receives the local map from the neighboring vehicles, it updates the location information of the current vehicle according to Algorithm 1, and the variables used are as shown in Table 2.

Line 2 shows the *ObservedID,* which is the observation target vehicle. Further, *RelativePosition* is the relative position of the observation target included in *IncomingLocalMap,* which is the received local map. In Line 3, the algorithm checks whether there exists an *ObservedID* that matches the current vehicle's ID (***cvid***). The presence of the matching information indicates that the neighboring vehicle directly observes the current vehicle through the sensors, and the relative position of current vehicle $\boldsymbol{SRP_{nvid\_cvid}}$ observed by neighboring vehicle (***nvid***) is translated to its absolute position (based on observing vehicle) according to Line 5.

Algorithm 1 Current Position Compensation of Vehicle Tracker

```
1  Procedure CurrentPositionCompensation
     (IncomingLocalMap)
2    For each (ObserverID, ObservedID, RelativePosition)
         ∈ IncomingLocalMap
3      If ObservedID is cvid
4        nvid = ObserverID
5        P_nvid_cvid = P_nvid + SRP_nvid_cvid
6        CP_gps = PositionCompensation(P_gps, P_nvid_cvid)
7        If CE^2_gps < E^2_gps
8          P_gps = CP_gps
9        End if
10     End if
11   End for
12 End Procedure
```

TABLE 2. Definition of parameters.

Symbol	Definition
cvid	Current vehicle's ID
nvid	Neighboring vehicle's ID
P_x	Absolute position of x
P_{x_y}	Translated absolute position of x based on y
SRP_{cvid_x}	Set of relative positions of x relative to current vehicle
w	Weight for compensating the error according to time
V_x	Velocity of x
CP_{gps}	Current vehicle's absolute position of GPS compensated by MLE
CE^2_{gps}	Current vehicle's error of GPS compensated by MLE
E^2_{gps}	Current vehicle's error of GPS
C_s	Period for updating the kinematic information of the current vehicle from the sensor, <100 ms
C_m	Period for broadcasting a message, 100 ms
C_g	Period for updating GPS, 1000 ms
T_c	Current timestamp
T_o	Observed timestamp

In Line 6, the new absolute position of current vehicle $\boldsymbol{CP_{gps}}$ is compensated with the absolute position of current vehicle $\boldsymbol{P_{gps}}$. At this time, the maximum likelihood estimation (MLE) method [24], [28] is used to compensate the absolute position.

MLE is a technique that can be useful in situations where the likelihood function is used to estimate the most probable parameter values based on observed samples, and generally expressed as (1), given variable $X = (x_1, x_2, \ldots, x_n)$, p is the probability density function of X, and $\mathcal{L}(\theta)$ is the likelihood of the occurrence of θ

$$\mathcal{L}(\theta) = p(x|\theta) \tag{1}$$

The proposed system aims to find position $\boldsymbol{CP_{gps}}$ that can maximize the likelihood, as shown in (2), by modifying the MLE method to estimate the accurate position of the vehicle with duplicated observations.

$$\boldsymbol{CP_{gps}} = argmax\ \boldsymbol{P_{nvid\_cvid}}\,\mathcal{L}\left(\boldsymbol{P_{nvid\_cvid}}\right) \tag{2}$$

In this case, the likelihood function can be expressed in the conditional probability because relative position observed by each vehicles are independent; thus, $\mathcal{L}\left(\boldsymbol{CP_{gps}}\right)$ can be expressed as

$$\mathcal{L}\left(\boldsymbol{CP_{gps}}\right) = \boldsymbol{p}\left(\boldsymbol{P_{1\_cvid}} \mid \boldsymbol{CP_{gps}}\right)\boldsymbol{p}\left(\boldsymbol{P_{2\_cvid}} \mid \boldsymbol{CP_{gps}}\right)\ldots \boldsymbol{p}(\boldsymbol{P_{n\_cvid}}|\boldsymbol{CP_{gps}}), \tag{3}$$

where $\boldsymbol{n}$ is the number of elements of $\boldsymbol{P_{nvid\_cvid}}$.

The probability density function of GPS approximates a normal distribution ($\mathbf{N}\left(\boldsymbol{\mu}, \boldsymbol{E^2}\right)$) with an average of $\boldsymbol{\mu}$ and an error of $\boldsymbol{E^2}$. Therefore, when $\boldsymbol{p}\left(\boldsymbol{P_{nvid\_cvid}} \mid \boldsymbol{CP_{gps}}\right) = \mathbf{N}(\boldsymbol{\mu},\boldsymbol{E^2})$, $\mathcal{L}\left(\boldsymbol{CP_{gps}}\right)$ can be expressed as (4) and the natural logarithm is taken on both sides to simplify the calculation to obtain (5).

$$\mathcal{L}\left(\boldsymbol{CP_{gps}}\right) = \prod_{i=1}^{n} e^{-\frac{(P_{i\_cvid}-CP_{gps})^2}{2E_i^2}} \tag{4}$$

$$\boldsymbol{ln}\left(\mathcal{L}\left(\boldsymbol{CP_{gps}}\right)\right) = -\frac{1}{2}\left(\sum_{i=1}^{n} \boldsymbol{E_i^{-2}}\right) \times \left(\boldsymbol{CP_{gps}} - \frac{\sum_{i=1}^{n} \boldsymbol{SRP_{cvid\_nvid_i}} \cdot \boldsymbol{E_i^{-2}}}{\sum_{i=1}^{n} \boldsymbol{E_i^{-2}}}\right) + \boldsymbol{Constant}, \tag{5}$$

where $\boldsymbol{n}$ is the number of elements of $\boldsymbol{P_{nvid\_cvid}}$.

Thus, the value of $\boldsymbol{CP_{gps}}$ which maximizes $\mathcal{L}\left(\boldsymbol{CP_{gps}}\right)$ is obtained as

$$\boldsymbol{CP_{gps}} = \frac{\sum_{i=1}^{n} \boldsymbol{P_{i\_cvid}} \cdot \boldsymbol{E_i^{-2}}}{\sum_{i=1}^{n} \boldsymbol{E_i^{-2}}}. \tag{6}$$

In addition, as proposed system $\boldsymbol{C_s} < \boldsymbol{C_m}$ is always true, the observation time of the relative position information received from neighboring vehicles can be changed. As the vehicles are still moving, the error of the observed relative position at the time of receiving and calculating the message increases. Therefore, instead of calculating the weight of each information only by the error of the sensor, a new weight $\boldsymbol{w}$ is defined to calculate the error according to the time, and the corrected relative position $\boldsymbol{CP_{gps}}$ is calculated. Weight $\boldsymbol{w}$ for calculating the error according to the time can be calculated through (7). Moreover, when the velocity of the observed vehicle is $\boldsymbol{V_{nvid}}$, absolute position $\boldsymbol{CP_{gps}}$ can be ultimately calculated as (8), and position error $\boldsymbol{CE^2_{gps}}$ is expressed by (9).

$$\boldsymbol{w_i} = \frac{\mathbf{1}}{\sqrt{\boldsymbol{E_i^2} + \left(\frac{\boldsymbol{T_c}-\boldsymbol{T_o}}{\boldsymbol{C_m}}\right)\boldsymbol{V_{nvid}}}} \tag{7}$$

$$\boldsymbol{CP_{gps}} = \frac{\sum_{i=1}^{n}\left(\boldsymbol{P_{i\_cvid}} \cdot \boldsymbol{w_i^2}\right) + \boldsymbol{P_{gps}} \cdot \boldsymbol{E_{gps}^{-2}}}{\sum_{i=1}^{n} \boldsymbol{w_i^2}} \tag{8}$$

$$\boldsymbol{CE^2_{gps}} = \sum_{i=1}^{n} \boldsymbol{w_i^2} + \boldsymbol{E_{gps}^{-2}} \tag{9}$$

Finally, in Line 7, the corrected absolute position error (CE^2_{gps}) is replaced with GPS position error of current vehicle (E^2_{gps}) if it is lower than E^2_{gps}.

The extended local map (ELM) constructor integrates the local map of the current vehicle constructed in the vehicle tracker with the local map received from the neighboring vehicles in the sharing phase. As the ELM contains local map information created by the vehicle tracker as well as the local information received from neighboring vehicles, it can include redundant relative position values for a specific vehicle.

Fig. 2 shows the relative position values for vehicle E observed by vehicles B and C.

In addition, if all vehicles can be connected, all neighboring vehicles can be configured to the ELM. In Fig. 2, vehicle B cannot be directly connected to vehicles A and C; however, as a common vehicle E exists, all neighboring vehicles can be connected together. However, if no vehicle exists for the connection [Fig. 4 (a)], and when a nonequipped vehicle is in front of vehicle A [Fig. 4 (b)], vehicle B cannot know the relative position of vehicle A positioned outside the range of the sensors [23].

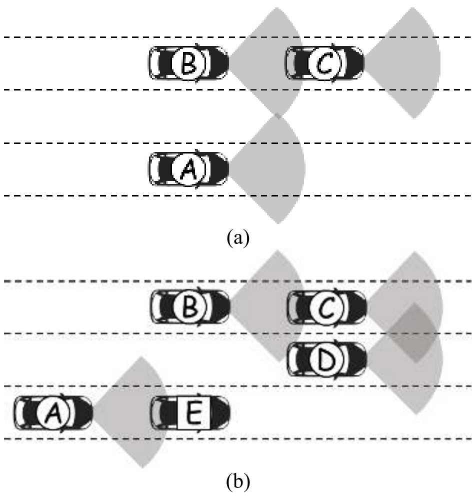

FIGURE 4. Vehicle topology with sensor range. Exemplary topology with (a) common vehicle and (b) nonequipped vehicle (Vehicle E).

That is, a vehicle that cannot be directly observed is not included in the local map, and cannot be included in the ELM as it is not connected to any common vehicle.

In this study, we solved this problem through Algorithm 2, in which Line 2 shows the addition of all local maps received to the ELM. Next, *MissingLinks* which represents vehicles that cannot be directly observed is searched for. To find MissingLinks, *CurrentVehicleID* indicating the current vehicle's ID is added to the set of directly observable vehicles, *Reachable*, in Line 11. In Lines 12–16, if there is a corresponding vehicle in the *Reachable*for all vehicle information in *ELM*, the vehicle directly observed by a specific *vehicle(vehicle.ObservedID)* is also included in *Reachable*. Therefore, the *Reachable* includes both the vehicle directly observed by current vehicle and the vehicle directly observed by those vehicles. Among the vehicles included in the *ELM*, the vehicles not included in *Reachable* become the *MissingLink*. Lines 17–21 show how *MissingLinks* are found. In Lines 4–7, to solve the *MissingLinks*, the relative position

Algorithm 2 Extended Local Map

```
1   Procedure ExtendedLocalMap
2     Add LocalMap to ExtendedLocalMap
3     MissingLinks = do getMissingLinks()
4     For each vehicle in MissingLinks
5       NewLink = do constructRPWithGPS(vehicle)
6       Add NewLink To ExtendedLocalMap
7     End for
8   End procedure
9
10  Procedure getMissingLinks
11    Add CurrentVehicleID To Reachable
12    For each vehicle in ExtendedLocalMap
13      If vehicle.VehicleID is in Reachable
14        Add vehicle.ObservedID to Reachable
15      End if
16    End for
17    For each vehicle in ExtendedLocalMap
18      If vehicle.VehicleID not in Reachable
19        Add vehicle to MissingLink
20      End if
21    End for
22    return MissingLink
23  End Procedure
```

of a *MissingLink* is generated using the GPS position of the current vehicle and the corresponding *MissingLink*; this new information on relative positions is added to the *ELM* so that the vehicles that cannot be directly observed are also included in the *ELM*.

Fig. 4 (b) shows that vehicle B generates the relative position information of vehicle A by using the GPS position information of the current vehicle and vehicle A so that the corresponding information is included in the ELM. As the relative position information is derived based on the GPS information, the sensor error value is set to an error value of the GPS error.

CNVPS also considers the case in which vehicles equipped with a CNVPS system (fully equipped vehicle) and without CNVPS (nonequipped vehicle) are mixed; a situation in which nonequipped vehicle E is in front of fully equipped vehicle A, as shown in Fig. 4 (b). Vehicle B does not know the existence of vehicle E but vehicle A recognizes the license plate of the vehicle in front of it, identifies the vehicle, and transmits the observation information to the neighboring vehicles so that vehicle B knows about the existence of vehicle E. The ELM is then passed to the topology mapper.

The topology mapper uses the ELM information from the ELM constructor in the correction phase to compensate for the lower error by using the relative position information of the vehicles redundantly observed by different neighboring vehicles. To compensate, the topology mapper uses the MLE method [24], [28] similar to that used in the vehicle-tracker module, and the variables used are listed in Table 3.

TABLE 3. Definition of parameters.

Symbol	Definition
cvid	Current vehicle's ID
nvid	Neighboring vehicle's ID
RP_{x_y}	Relative position of x relative to y
SRP_{cvid_x}	Set of relative position of x relative to current vehicle
CRP_x	Relative position of x compensated by MLE
E_x^2	Error of x's relative position
CE_x^2	Error of x's relative position compensated by MLE

In the proposed system, at the sharing phase, the relative position information received from the neighboring vehicle is composed of the ELM, which allows duplication of the observation values for the same vehicle. In the correction phase, if observation information about the same vehicle exists among the relative position information received from neighboring vehicles, the relative position information is integrated and compensated as one relative position information. First, all the relative position information in the ELM is converted into relative position information $\boldsymbol{RP_{cvid\_nvid}}$ based on the current vehicle. As $\boldsymbol{RP}$ value for a specific $\boldsymbol{nvid}$ can be more than 1, it is defined as set $\boldsymbol{SRP_{cvid\_nvid}}$.

The CNVPS modifies the MLE method to estimate the accurate position of a vehicle with a duplicate observation and aims to find position $\boldsymbol{CRP_{nvid}}$ that maximizes the likelihood, as shown in (10).

$$\boldsymbol{CRP_{nvid}} = argmax\ \boldsymbol{SRP_{cvid\_nvid}}\mathcal{L}\left(\boldsymbol{SRP_{cvid\_nvid}}\right) \quad (10)$$

By using the same method as described for the vehicle tracker, the $\boldsymbol{CRP_{nvid}}$ value that maximizes $\mathcal{L}(\boldsymbol{CRP_{nvid}})$ is calculated as

$$\boldsymbol{CRP_{nvid}} = \frac{\sum_{i=1}^{n} \boldsymbol{SRP_{cvid\_nvid_i}} \cdot \boldsymbol{E_i^{-2}}}{\sum_{i=1}^{n} \boldsymbol{E_i^{-2}}}. \quad (11)$$

Finally, the compensated position of $\boldsymbol{CRP_{nvid}}$ of the vehicle is obtained as in (12), and position error $\boldsymbol{CE_{nvid}^2}$ of each vehicle is obtained as in (13).

$$\boldsymbol{CRP_{nvid}} = \frac{\sum_{i=1}^{n} \boldsymbol{SRP_{cvid\_nvid}} \cdot \boldsymbol{w_i^2}}{\sum_{i=1}^{n} \boldsymbol{w_i^2}} \quad (12)$$

$$\boldsymbol{CE_{nvid}^2} = \sum_{i=1}^{n} \boldsymbol{w_i^2} \quad (13)$$

The received ELM can be represented as a graph, and if more than two edges exist for one vehicle, it is observed from different neighboring vehicles. The topology mapper integrates the observed relative position information with one relative position information, and the corrected local map is passed to local map data.

C. DATA MANAGER

The data manager provides a means to access the information generated by the proposed system and the information contained in the BSM sent by the neighboring vehicle through the API. The vehicle-related application can access the data manager and obtain information about the neighboring vehicle including the vehicle's current position, speed, direction, acceleration, break, steering angle, and relative position information, as well as the relative position calculated by the proposed system.

The local map data provides information of the neighboring vehicles, such as license plate ID, relative position coordinates, and the last updated time, to the upper application by using an ELM containing the compensated relative position constructed in the cooperative positioning manager.

The peer data manages the information of neighboring vehicles delivered to the BSM, which is configured as shown in Table 4, and contains information that can be used in vehicle-safety applications not including the CNVPS. The information about each vehicle is managed according to the unique ID of a vehicle, and information about a vehicle can be searched for by using this unique ID, which can be found in the ELM of the local map data; this can be used by applications to obtain recent information.

TABLE 4. Structure of basic safety message.

Field	Content
msgID	Message Sequence Number
	PART 1
Pos	Position coordinates in 3D
Motion	Vehicle motion information such as velocity, heading, acceleration, etc
Control	Vehicle control information such as brake status, steering angle, etc
Basic	Basic vehicle information such as vehicle length.
	PART 2 (Optional)
safetyExt	Safety related vehicle sensor information
status	User-specified extended information

IV. PERFORMANCE EVALUATION

To compare the performance of CNVPS to existing GPS-based systems, performance evaluation was performed by simulating virtual road and communication environment. The simulation was performed using the Vehicle in Network Simulation (Veins) Framework [25].

The Veins Framework is used for simulating V2V communication-based applications, which can simulate road environment and wireless network simultaneously. The framework uses simulation of urban mobility (SUMO) [26] for simulating a road environment and OMNet++ [27] for simulating a wireless network environment. Information on the software used is shown in Table 5, and hardware used is shown in Table 6.

TABLE 5. Software used in simulation.

Software	Version
V2V Network Simulation Framework	Veins-3.0
Vehicle behavior simulator	SUMO-0.21
Discrete Event Simulator	Omnetpp-4.4.1

TABLE 6. Hardware used in simulation.

Hardware	
Processor	AMD Ryzen 7 1800X 3.59GHz
RAM	32GB DDR4 SDRAM

A. SIMULATION PARAMETERS AND SIMULATION SCENARIO

To simulate the CNVPS and GPS-based system, network and wireless simulation were performed based on 802.11p protocol, which is the most typical network protocol in V2V communication. In addition, sensors used were assumed to follow a Gaussian distribution with error of 1.5 m. The simulation scenarios were implemented and tested on three environments, i.e., expressway, intersections, and urban environment, and the scenarios are as follows.

Scenario 1 assumes the case of a highway environment, with Fig. 5 showing the scenario map, and consists of a straight line with gentle corners and six lanes. Vehicles on this road are divided into four groups, with twelve cars spawning in the random lanes for each group, driving on a group basis. The characteristics of each group are acceleration, deceleration, vehicle length, and maximum speed of the vehicle, as shown in Table 7.

FIGURE 5. Highway scenario map.

TABLE 7. Parameters for group characteristics in Highway scenario.

	Group A	Group B	Group C	Group D
Acceleration (m/s^2)	3	2	1	1
Deceleration (m/s^2)	6	6	5	5
Vehicle Length (m)	5	7.5	7.5	7.5
Maximum Speed (m/s)	30	35	30	35

Scenario 2 assumes the case of a crossroad with traffic lights. The scenario map is as shown in Fig. 6, showing a crossroad, in which four roads of two lanes intersect, and traffic lights exist on each road. Vehicles on this road are divided into four groups, with twelve cars spawning in random lanes for each group, driving on a group basis. The characteristics of each group are acceleration, deceleration, vehicle length, and maximum speed of the vehicle, as shown in Table 8.

Scenario 3 assumes the case of an urban environment. The scenario map is as shown in Fig. 7, and an actual urban road is modeled. This road modeling creates an environment similar to that of a real road, including elements such as speed limits and traffic lights. Vehicles on this road are divided into four groups, with 40 cars spawning in random lanes for each group, driving on a group basis. The characteristics of each group are acceleration, deceleration, vehicle length, and maximum speed of the vehicle, as shown in Table 9.

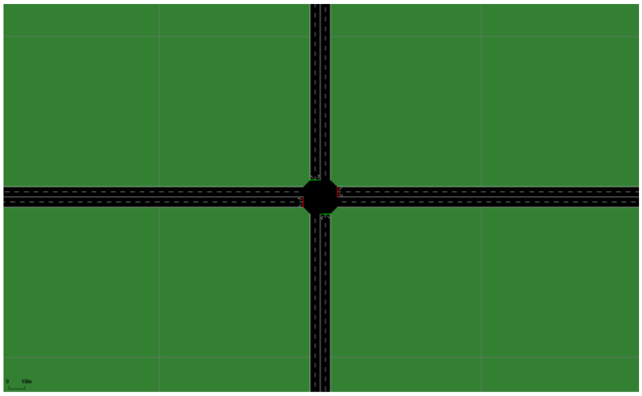

FIGURE 6. Crossroad scenario map.

TABLE 8. Parameters for group characteristics in Crossroad scenario.

	Group A	Group B	Group C	Group D
Acceleration (m/s^2)	3	2	1	1
Deceleration (m/s^2)	6	6	5	5
Vehicle Length (m)	5	7.5	7.5	7.5
Maximum Speed (m/s)	10	15	10	15

FIGURE 7. Urban scenario map.

TABLE 9. Parameters for group characteristics in Urban scenario.

	Group A	Group B	Group C	Group D
Acceleration (m/s^2)	3	2	1	1
Deceleration (m/s^2)	6	6	5	5
Vehicle Length (m)	5	7.5	7.5	7.5
Maximum Speed (m/s)	10	15	10	15

B. SIMULATION RESULTS AND PERFORMANCE ANALYSIS

To compare the CNVPS proposed in this paper with the existing GPS-based system, we compared the average relative position error obtained through CNVPS with the average

GPS-based relative position error, and set the root mean square (RMS) error of the GPS to 8 and 15 m to measure the impact of GPS's accuracy to that of CNVPS. In addition, the error and recognition rates of the vehicles were measured when the ratio of the CNVPS-equipped vehicle was 50%–100% of the total vehicles when CNVPS-equipped vehicle and unequipped vehicles were mixed.

Fig. 8 shows the simulation result when the RMS error of the GPS was 15 m. In the case for every vehicle equipped with CNVPS, the relative position error is reduced to 26% of the GPS error, and no significant difference is observed when the CNVPS-equipped vehicle ratio is high for each scenario. However, as CNVPS-equipped vehicle ratio decreases, the relative position errors in scenarios 1 and 3 are greatly increased, while that of scenario 2 is not.

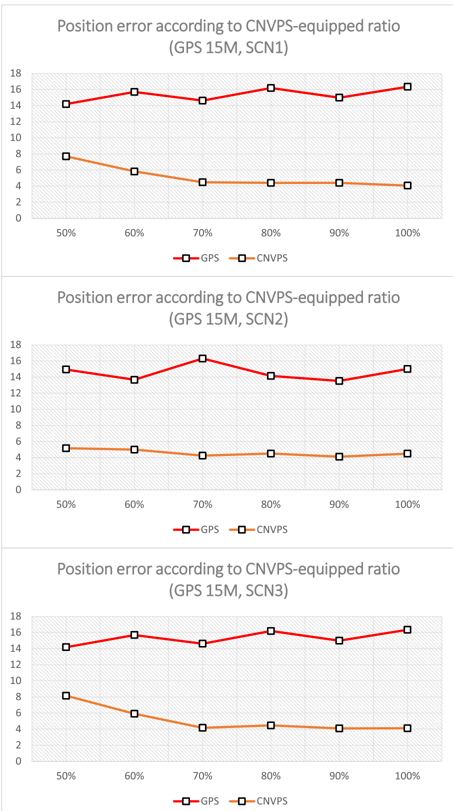

FIGURE 8. Position error according to CNVPS-equipped ratio (GPS RMS of 15 m).

In scenario 1, the vehicle is driving at a high speed and its density is relatively low. Therefore, as the ratio of CNVPS-equipped vehicles decreases, the number of vehicles that cannot be tracked increases, thus increasing the chances of error in the application of the cooperative position compensation. This in turn leads to an increased error.

Scenario 3 also shows that the error increases as the ratio of CNVPS-equipped vehicles decreases, due to frequent turnaround, low vehicle densities, and limited communication because of buildings. In contrast, scenario 2 shows no significant increase in the error even when CNVPS-equipped vehicle ratio decreases owing to high vehicle density and low vehicle speed.

Fig. 9 shows the simulation result when RMS error of GPS is 8 m. Compared to the above-mentioned results, the RMS error of GPS decreases, and the error of CNVPS decreases. Thus, the performance of CNVPS increases with the improvement of GPS performance.

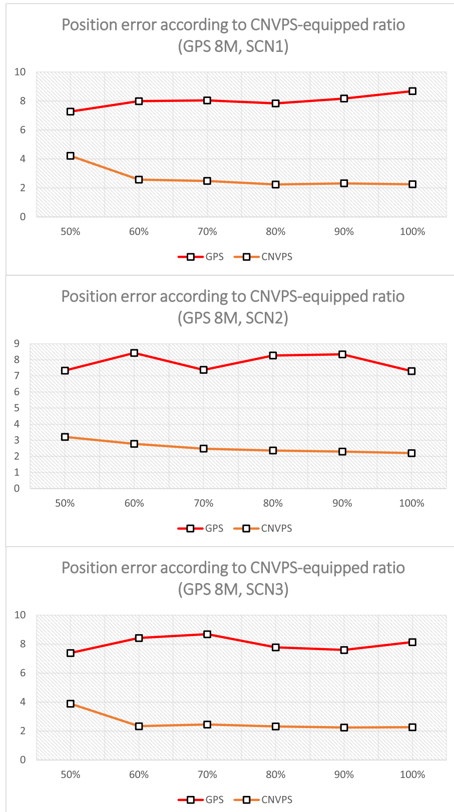

FIGURE 9. Position error according to CNVPS-equipped ratio (GPS RMS 8M).

The GPS has a direct impact on its RMS error and shows a large average error. In contrast, when every vehicle is equipped with CNVPS, it shows 27% error than the GPS-based error; this is an improvement by 370%. In addition, the error rate was confirmed to be maintained lower than that in an GPS-based system, even though the error rate increases when CNVPS-equipped-vehicle ratio decreases.

Regarding system latency, CNVPS generate location information at least every 100 ms (C_m), which is period for broadcasting local map, and it is 10 times faster than period for GPS update, which is 1000 ms (C_g).

And End-to-End latency for CNVPS, which is important as CNVPS is real-time system, is measured in two part. System overhead latency, which is latency induced by position compensation algorithms in CNVPS, and V2V communication latency, which is latency of network between cars. In simulation, we measured system overhead to be less than 1 ms in our environment, and V2V communication latency to be less than 1 ms as well. As time constraint required by vehicle safety application is 100ms, end-to-end latency for CNVPS, which is less than 2ms, it is insignificant in terms of real-time system, and as suggested algorithms used in CNVPS are in linear complexity, it can be said that it is suitable as real-time system.

V. CONCLUSION

In this paper, we proposed a CNVPS, which is a vehicle positioning system that provides low error and high refresh rate of location information of neighboring vehicles required by the critical safety-related services of ITS. To address the high error rate and low refresh rate of location information in existing GPS-based systems, CNVPS utilizes not only GPS but also cameras, a ranging sensor, and inertial sensors to recognize neighboring vehicles, cooperatively compensate via V2V communication, and expand the recognition range.

The simulation results show that the CNVPS achieves 370% accuracy than the existing GPS-based system, and 10 times faster location refresh rate because of the use of location compensation algorithm and dead reckoning. The proposed system can provide the location of a wide range of neighboring vehicles quickly and accurately; therefore, it is expected to contribute toward a safe driving environment as various safety services can be provided.

As a future work, systems with lower error margins must be researched as higher vehicle densities may result in narrow gaps between vehicles; this requires a low range of errors. Further, systems with better performance for vehicles with and without V2V communications are required.

REFERENCES

[1] G. Karagiannis *et al.*, "Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions," *IEEE Commun. Surveys Tuts.*, vol. 13, no. 4, pp. 584–616, Nov. 2011.

[2] A. Benslimane, "Localization in vehicular ad hoc networks," in *Proc. Syst. Commun.*, Aug. 2005, pp. 19–25.

[3] A. Amini, R. M. Vaghefi, J. M. de la Garza, and R. M. Buehrer, "Improving GPS-based vehicle positioning for intelligent transportation systems," in *Proc. Intell. Vehicles Symp.*, Jun. 2014, pp. 1023–1029.

[4] N. M. Drawil and O. Basir, "Intervehicle-communication-assisted localization," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 678–691, Sep. 2010.

[5] M. Elazab, A. Noureldin, and H. S. Hassanein, "Integrated cooperative localization for connected vehicles in urban canyons," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2015, pp. 1–6.

[6] T. Kos, I. Markezic, and J. Pokrajcic, "Effects of multipath reception on GPS positioning performance," in *Proc. IEEE ELMAR.*, Sep. 2010, pp. 399–402.

[7] C. V. S. C. Consortium *et al.*, "Vehicle safety communications project: Task 3 final report: Identify intelligent vehicle safety applications enabled by DSRC," U.S. Dept. Transp., Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. DOT HS 809 859, 2005.

[8] J. Zhang, "Multi-source remote sensing data fusion: Status and trends," *Int. J. Image Data Fusion*, vol. 1, no. 1, pp. 5–24, Nov. 2010.

[9] C. Patel, D. Shah, and A. Patel, "Automatic number plate recognition system (ANPR): A survey," *Int. J. Comput. Appl.*, vol. 69, no. 9, pp. 1–13, 2013.

[10] X. Wen, L. Shao, Y. Xue, and W. Fang, "A rapid learning algorithm for vehicle classification," *Inf. Sci.*, vol. 295, pp. 395–406, Feb. 2015.

[11] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: A survey," *Mach. Vis. Appl.*, vol. 25, no. 3, pp. 727–745, 2014.

[12] S. Fujii *et al.*, "Cooperative vehicle positioning via V2V communications and onboard sensors," in *Proc. IEEE Veh. Technol. Conf. (VTC)*, Sep. 2011, pp. 1–5.

[13] A. A.-K. Jeng, C. Chen, R.-H. Jan, W.-Y. Fong, and H.-R. Tseng, "VIP: Video-assisted inter-vehicle positioning system," in *Proc. 12th Int. Conf. ITS Telecommun. (ITST)*, Nov. 2012, pp. 522–526.

[14] L. Altoaimy and I. Mahgoub, "OWL: Optimized weighted localization for vehicular ad hoc networks," in *Proc. Int. Conf. Connected Vehicles Expo (ICCVE)*, Nov. 2014, pp. 699–704.

[15] *Global Positioning System Standard Positioning Service Performance Standard, Assistant Secretary of Defense for Command, Control, Communications, and Intelligence*, U.S. Dept. Defense, Washington, DC, USA, 2001. .

[16] K. Ansari, C. Wang, L. Wang, and Y. Feng, "Vehicle-to-vehicle real-time relative positioning using 5.9 GHz DSRC media," in *Proc. IEEE 78th Veh. Technol. Conf. (VTC)*, Sep. 2013, pp. 1–7.

[17] G. Yanling, L. Lichen, G. Meng, and G. Lewen, "Urban road traffic state identification algorithm based on particle filter and fuzzy discrimination," *Int. J. Smart Home*, vol. 9, no. 8, pp. 229–238, 2015.

[18] J. Georgy, A. Noureldin, and M. Bayoumi, "Mixture particle filter for low cost INS/Odometer/GPS integration in land vehicles," in *Proc. IEEE 69th Veh. Technol. Conf. (VTC)*, Apr. 2009, pp. 1–5.

[19] S. Zabihi, S. S. Beauchemin, E. A. M. De Medeiros, and M. A. Bauer, "Lane-based vehicle localization in urban environments," in *Proc. IEEE Int. Conf. Veh. Electron. Saf. (ICVES)*, Nov. 2015, pp. 225–230.

[20] J. Gim and C. Ahn, "Ground vehicle localization based on inertial measurement signals," in *Proc. 15th Int. Conf. Control, Automat. Syst. (ICCAS)*, Oct. 2015, pp. 1775–1779.

[21] *Dedicated Short Range Communications (DSRC) Message Set Dictionary*, SAE Standard J2735_200911, DSRC Committee, 2009.

[22] L. Hu and D. Evans, "Localization for mobile sensor networks," in *Proc. ACM 10th Annu. Int. Conf. Mobile Comput. Netw.*, 2004, pp. 45–57.

[23] O. Abumansoor and A. Boukerche, "A cooperative multi-hop location verification for Non Line Of Sight (NLOS) in VANET," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2011, pp. 773–778.

[24] I. J. Myung, "Tutorial on maximum likelihood estimation," *J. Math. Psychol.*, vol. 47, no. 1, pp. 90–100, Feb. 2003.

[25] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of sumo-simulation of urban mobility," *Int. J. Adv. Syst. Meas.*, vol. 5, nos. 3–4, pp. 128–138, 2012.

[26] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO—Simulation of Urban MObility," in *Proc. 3rd Int. Conf. Adv. Syst. Simulation (SIMUL)*, Barcelona, Spain, vol. 42, 2011, pp. 1–6.

[27] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," in *Proc. 1st Int. Conf. Simulation Tools Techn. Commun., Netw. Syst. Workshops*, 2008, p. 60.

[28] J.-X. Pan and K.-T. Fang, "Maximum likelihood estimation," in *Growth Curve Models and Statistical Diagnostics*. New York, NY, USA: Springer-Verlag, 2002, pp. 77–158.

[29] T. Aono, K. Fujii, S. Hatsumoto, and T. Kamiya, "Positioning of vehicle on undulating ground using GPS and dead reckoning," in *Proc. IEEE Int. Conf. Robot. Automat.*, vol. 4, May 1998, pp. 3443–3448.

SANGHYUCK NAM received the B.S. degree in computer science and engineering, in 2017. He is currently pursuing the M.S. degree with the School of Computer Science and Engineering, Chung-Ang University.

His research interests include mobile systems, fog computing, vehicular ad-hoc networks, and machine learning.

DANHEE LEE received the B.S. and M.S. degrees from the School of Computer Science and Engineering, Chung-Ang University, in 2014, and 2017, respectively.

His research interests include artificial intelligence and vehicular ad-hoc networks.

JAEHWAN LEE received the B.S. degree in computer engineering from the School of Computer Science and Engineering, Chung-Ang University, Seoul, South Korea, in 2015, where he is currently pursuing the Ph.D. degree in software engineering.

His research interests include vehicular ad-hoc networks, mobile computing, and embedded systems.

SANGOH PARK received the B.S., M.S., and Ph.D. degrees from the School of Computer Science and Engineering, Chung-Ang University, in 2005, 2007, and 2010, respectively, where he has been serving as an Assistant Professor with the School of Computer Science and Engineering, since 2017.

He served as a Senior Researcher with the Global Science Experimental Data Hub Center, Korea Institute of Science and Technology Information, from 2012 to 2017, and as a Research Professor with the School of Computer Science and Engineering.

His research interests include embedded systems, big data systems, cyber physical systems, and Linux systems.