A Fusion Method for Localization of Intelligent Vehicles in Car parks

Dinh-Van Nguyen¹, Trung-Kien Dao¹, (Member, IEEE), Eric Castelli², and Fawzi Nashashibi³, (Member, IEEE)

¹MICA Institute (HUST-Grenoble INP), Hanoi University of Science and Technology, Hanoi, Vietnam
²CNRS, Inria, Institute of Engineering, University of Grenoble Alpes, LIG, 38000 Grenoble, France
³RITS Team, INRIA Paris, Paris, France

Corresponding author: Dinh-Van Nguyen (dinh-van.nguyen@mica.edu.vn)

This work was supported by the Vietnam Ministry of Education and Training under Grant B2020-BKA-12.

ABSTRACT

With the increasing demand for urban space, more and more multistory carparks are needed as they will play a central role in the city transportation system. An autonomous navigation solution for Intelligent Vehicles in these indoor scenarios is then required. One step to solve this problem is to localize Intelligent Vehicles in these specific environments. However, the lack of GPS due to signal obstruction (multipath, non-line of sight, interference, etc.) appears to be a significant concern for any localization system, let alone indoor ones. Hence, in this paper, a wireless sensor network based approach is proposed to replace the GPS (Global Positioning System) role for indoor environments. In addition, a fusion framework for multiple sensors such as Wi-Fi access points, Inertial Measurement Unit (IMU) or LiDAR is studied. Experiments in almost one-year duration yield a stable result of mean global localization error at 0.5m.

INDEX TERMS

Autonomous vehicle, fusion, Gaussian mixture, GPS-denied environment, intelligent vehicles, localization, Wi-Fi fingerprinting.

I. INTRODUCTION

In urban area, carparks are extremely important for a fluid transportation system. Without them, there would be a huge burden on public space. As the intelligent vehicles tend to be electric vehicles, a carpark will also act as a much-needed charging station. However, they also introduce a new problem: reports in [1] and [2] suggest that the average searching time for a free slot in a carpark in French big cities as Paris or Lyon is 20 minutes and can be as high as 40 minutes for some districts. This leads to an approximately 70 million hours of searching each year or equivalently 700 million euros loss for France alone. Therefore, future uses of carparks will probably exceed their original objectives: demanding features such as electric charger, online booking of parking spaces, dynamic guidance or mobile payment etc. will turn a carpark into competitive smart environments.

Several carpark improvements are developed such as an automated carpark system [3] or smart carpark guidance and management system [4], etc. These systems either require a completely rebuild in conventional carpark design with a costly investment or ask for a various sensors and computing system to guide user from software applications. This is what motivates intelligent vehicle designers to move towards fully autonomous navigation in an indoor environment such as a carpark to eliminate the time-wasting issue and enhance effectiveness and safety of car parking. With the centre role of car park in the transport chain, solving such problem would benefit the traffic flow of the whole system.

The dream of having an intelligent vehicle navigating autonomously in different environments, has been realized step by step during the last ten years. One of those steps is the challenging task of locating a vehicle position in different environments. Often it is impossible to use Global Positioning System (GPS) in certain environments such as indoor environments. While outdoor, GPS-aided localization for intelligent vehicles has been widely studied in recent years, indoor, GPS-denied localization is yet to be fully addressed.

From both theoretical and practical perspectives, the problem of navigation for intelligent vehicles in GPS-denied environments deserves a complete solution. This paper will focus on solving a crucial part of it namely localization. Throughout this study, the term GPS-denied environment will...
be defined as an environment with poor or no GPS signal and GPS-aided environment refers to the one with good reception of GPS signal. The targeted environments are the GPS-denied environments such as indoor carpark and can also be extended to places with poor GPS signals reception such as industrial factory, university campus or outdoor carparks, etc. In-depth explanation of the targeted environment can be found in [28].

By targeting environment such as a carpark in the framework of a university campus, the average movement speed of vehicles is expected to be around 3m/s [1]. This is due to the nature of these environments conditions as well as speed regulation applied. Understanding the vehicle’s dynamics in the localization problem will help to accurately identify advantages/disadvantages of different positioning methods.

Having target environment defined, we now identify requirements for a carpark localization system:

- **Availability:** The system should be deployed on existing infrastructure of a carpark with limited requirements of changes in structures, hardware or software.
- **Scalability:** The system should be extensible and scalable in large scale.
- **Universality:** There should be no specific hardware/firmware changes other than off-shelf devices.
- **Accuracy:** The final fusion localization system in this paper is expected to be within 0.5m of mean localization error.

First, this paper introduces a novel method for GPS-denied environment localization using wireless sensors networks, more specifically a Wi-Fi fingerprinting localization system. Although Wi-Fi fingerprinting localization system is already a popular approach for indoor localization, so far it only targets pedestrian walking speed. The advantages of this method are its availability, scalability and universal characteristics where off-the-shelf hardware like Wi-Fi receivers and Wi-Fi access points are used without any modification. These sensors are also expected to be widely available nowadays in urban area. One main concern of this method is the low sampling frequency of Wi-Fi scan. In general, the time to complete a scan of Wi-Fi signals is around 1 second (i.e., 1Hz). At 1.0 to 1.6m/s of human walking speed [5], this sampling frequency is adequate to deliver real-time localization results. However, as the paper aims to target intelligent vehicles in the carpark at 3m/s, the classic approach of the Wi-Fi fingerprinting method is insufficient. Thus, an original approach using an ensemble neural network on Wi-Fi fingerprinting method is proposed.

Second, a complete localization solution should be a fusion of multiple techniques. This allows global as well as local levels of localization to function together. At the same time, having redundancy in the system boosts accuracy and reliability. In this paper, a flexible fusion framework for multiple localization sensors is proposed. This fusion framework will not only deal with the GPS-denied environment but could be potentially used in the GPS-aided environment and provide a smooth transition between the two kind of areas.

II. STATE OF THE ART

A. OVERVIEW OF INTELLIGENT VEHICLES LOCALIZATION IN GPS-DENIED ENVIRONMENTS

Localization is a task of determining an object’s pose (e.g. coordinate, heading angle) or the spatial relationship among objects. This is an essential task for autonomous navigation that a vehicle need to achieve [6]. Only by precisely knowing the location of itself in either a local or a global map, then action such as path planning or obstacles avoidance can be carried out. Often, this task is accomplished through a set of dedicated sensors (on vehicle sensors or environment sensors). The process of combining these sensors inputs to infer the vehicle’s position is called sensor fusion.

In this section, a quick review of localization methods for intelligent vehicles is presented. Both local and global levels of localization methods and more specifically, those that are dedicated to the GPS-denied environment, are studied. In the last few years, the research community in Intelligent Vehicles has been developing several dedicated systems for localization in GPS-denied areas in general and carparks in particular. Due to the lack of GPS signals, most of the solutions for localization in this domain fall into the local level of localization. Depending on the choice of the reference coordinate system, these works can be categorized into two classes: absolute localization (or map-based) methods and relative localization (self-centric, without a map) methods. The two classes’ recent works will be studied in the following sections.

1) ABSOLUTE LOCALIZATION

In the absolute positioning approach, it is required that a map of the environment is known beforehand by the vehicle. In this map, there are two main components: static objects which contribute to the structure of the map (e.g. roads, walls, doors, etc.), and dynamic objects which are moving obstacles in the environment (e.g. other vehicles, pedestrian, etc.). Depending on the solution, the map may contain both or just static objects.

A FastSLAM approach can be found in [7] where a Rao-Blackwellized particle filter laser-SLAM is implemented. Given that a full SLAM can be stated as in (1), given $x_t$ the vehicle’s state at time $t$, $m$ is the map built in SLAM process, $z_t$ sensors reading at $t$ and $u_t$ is control input. In this solution, the static map (i.e., map with only static objects) of the environment is included and denoted as $s$. Thus, the full SLAM problem with a static map can be formulated as in (2).

$$\begin{align*}
    p (x_{1:T}, m | z_{1:T}, u_{1:T}) \\
    p (x_{1:T}, m | s, z_{1:T}, u_{1:T})
\end{align*}$$

Note that the static map only influences the SLAM map directly. By adding static map $s$ to classical SLAM, the posterior over maps can be computed using products of all cells (given an occupancy grid representation of map is selected)
as shown in (3).

\[ p(m | z_{1:T}, u_{1:T}, x_{1:T}) = \prod_i p(mi | z_{1:T}, u_{1:T}, x_{1:T}, s_i). \] (3)

Using information from a static map, the posterior of each cell can be estimated as in (4).

\[ P = p(mi | z_{1:T}, u_{1:T}, x_{1:T}, s_i) \]

\[ = \begin{cases} 
1, & s_i \text{ occupied;} \\
\nu_p m_i | z_{1:T}, u_{1:T}, x_{1:T}, s_i, & s_i \text{ unknown.} 
\end{cases} \] (4)

Given a static map with a resolution of 0.125m each cell, a parking garage of 80 x 35m² is tested in this study. A 0.19m in position error and 2.3° in orientation error are reported. This solution, however, encountered issues with dynamic objects that obstruct the view to static map components and scalability potential since a high-resolution static map is required as it directly influences positioning error estimation.

A similar solution of laser-based FastSLAM can be found in [8]. In this solution, a static map of static (e.g., walls, doors, columns, etc.) and semi-static objects (objects that are supposed to be static for a short period of time, e.g., parked cars). This solution is tested with a 0.05m resolution map and returns a 0.33m of position error as well as 1.03° of orientation error. Despite a marked improvement in the orientation error compared to the previous study, this solution requires higher map resolution but shows a higher position error.

A dead-reckoning based map matching method for carpark is presented in [9]. The core idea of this work is to use a detailed 3D structural map of the environment and process map-matching with particle filter and collision detection. The collision detection in this context is defined as the intersection/overlapping of a particle (corresponding to a vehicle’s possible position) with the 3D structure of the garage, thus effectively results in the weight of each particle. The resulting localization is limited to 1.5m of accuracy partially due to inaccurate 3D mapping and dead-reckoning sensors’ noise.

Research presented in [10] demonstrates a fully autonomous vehicle in carpark. The vehicle is equipped with four monocular fisheye cameras, two stereo cameras and stock ultrasonic sensors. The method has an offline 3D mapping phase using cameras sensors for the entire environment and an online visual localization using grid map and map matching using both thresholds of the distance in image-space and the descriptor distance. In addition, a semantic mapping of the environment is required. This includes: a road-map graph which gives details about positions of lanes, way direction and intersections; the location of parking spaces; the speed profile allowed in the carpark. The system has been tested and has been able to successfully park automatically a vehicle within 0.1m of lateral and 0.15m of longitudinal localization accuracy.

2) RELATIVE LOCALIZATION

In contrast to absolute localization, relative localization does not require an extensive map of the environment. The approach aims to estimate the vehicle position relative to its surrounding local objects such as other vehicles, lane marking, etc.

A free parking slot searching strategy using four fisheye cameras are shown in [11]. Although the method focuses on detecting free parking lots, it also highlights the technique of localizing vehicles relatively to other objects. Using pre-learned parking lot markings information such as standard width, parking lot structure as well as a database of training images for occupancy parking lots, the algorithm successfully locates the vehicle position relative to parking lots within 0.25m of accuracy.

Inter-vehicular network between GPS-aided and GPS-denied area vehicles is studied in [12]. This study introduces a novel way of communicating with a relative location between GPS-aided and GPS-denied area vehicles. By exchanging not only location but also Road Infrastructure Objects (RIOs), the network of vehicles has successfully improved localization of each compared to standard GPS accuracy. While this approach does not require a detailed map of the environment, it does require that the precise locations of RIOs are known. The estimated accuracy of this method is roughly 2.5m given a dense distribution of RIOs and other vehicles.

B. CARPARK LOCALIZATION USING WIRELESS SENSORS NETWORK

Recently, there are several attempts to use the concept of Wi-Fi fingerprinting in determining position of a vehicle. Some approaches take advantages of users’ smartphones to assist and guide driver to a parking lot. Other approaches directly aim for intelligent vehicles with sensors mounted on vehicles. Depending on the choice of sensors (smartphone or mounted sensors), the accuracy of the localization system is likely to be affected. In this section, several notable studies are examined.

An early attempt can be found in [13] where the authors use handheld devices to determine vehicles’ position. A classical Wi-Fi fingerprinting module is implemented on user’s device to determine its position. A dead-reckoning module is built using inputs from smartphones’ sensors such as accelerometer, compass, etc. The two modules are then combined to estimate the final position. The reported RMSE is around 4m and 95% of the errors are under 6m. The whole system is reported to function at 1Hz. Thus, the vehicle speed is not expected to be high. Since the authors only target giving location-based services such as: guidance to points of interest, location-based information or car finding, etc., this accuracy is relatively sufficient.

An approach in [14] presents a combination of fingerprinting for both Wi-Fi and Bluetooth Low Energy (BLE) devices. The study divides the environment into grid-based map and build a radio map for each intersection. A 4800m² carpark is used as a test bed. A total of 60 BLE access points and 10 Wi-Fi access points are used. The BLE sensors are operated at 10Hz while the Wi-Fi scan is performed at a rate of 1Hz. Three databases of fingerprint for Wi-Fi, BLE...
and a combination both Wi-Fi and BLE are built to evaluate the accuracy of each method. With the vehicle speed in this research is around 10km/h (or 2.8m/s), the estimated mean error for localization is around 4.4m.

A recent study in [15] also exploits a possibility of using Wi-Fi fingerprinting localization for a smart vehicle in a university campus area. The study is conducted on the campus of the Universidad Carlos III de Madrid with 30 points chosen in the environment to be training positions. Unlike other standard Wi-Fi fingerprinting approaches, this study treats the environment as a grid-based map with a resolution of 15cm. With this grid-based map, an attempt to localize the vehicle in an arbitrary position (which most of the time is different from training position) is made. A Support Vector Regression algorithm (SVR) is implemented to handle the continuous solution space. The experiments are carried out with 175 recorded Wi-Fi access points, an intelligent vehicle operated at 4.86km/h speed moving through a path of 145m. Their results show an average localization error of 6.18m in the best case.

III. Wi-Fi Ensemble Fingerprinting

The fingerprinting localization method is a popular method for WSNs localization given its simplicity, easy to deploy and scalable characteristics. The main idea of the method is: given a network of radio transmitters, at any position in the environment covered by the network, the combination of strengths and transmitters ID (most commonly transmitter’s MAC address) which is unique for any location in the environment. For instance, if a receiver scans for RSSI from a total of 5 transmitters in the environment, each with a different MAC address, consequently, the unique feature vector of this location \((x, y)\) will be written as:

\[
\{(MAC_1, RSSI_1), ..., (MAC_5, RSSI_5)\} \rightarrow (x, y) .
\]

By building a database of multiple locations’ feature vectors in the environment, a localization process later can simply be carried out with a classification algorithm of real-time RSSIs scan. Each location with feature vectors registered in the database will be called a fingerprint.

In the online phase where localization estimation is carried out, the vehicle will move inside the environment while scanning for RSSIs from surrounding APs. A likelihood function based on data from offline phase is defined as in (6). Where \(X_t\) the input vector of RSSIs scan at time \(t\) and \(c_t\) the likelihood score of \(X_t\) to be scanned at \(FP_t\) in the environment with regard to the fingerprint database. In general, the fingerprint with the highest likelihood score will be chosen as the estimated location:

\[
h (X_t) = c_t .
\]

There are two major issues for Wi-Fi fingerprinting localization when it comes to vehicles tracking. Firstly, the time to complete a scan of Wi-Fi signals in an environment depends on multiple factors but it is generally around 1 second. This means the Wi-Fi receiver performs at 1Hz of sampling frequency. For the human average walking speed of around 1.35m/s [16], this low sampling frequency is neglected in the tracking problem. However, a vehicle in a carpark moves at 3-3.3m/s [1]. With 1Hz of sampling frequency, the Wi-Fi fingerprinting localization system is only capable of giving a measurement every 3m. Since localizing a vehicle is a demanding task in terms of precision, this low sampling rate is inadequate. Secondly, the Wi-Fi signal, as any other radio signal, is influenced by multiple factors in the environment such as other radio signals, heavy metal obstacles, etc. this results in a high variance of RSSIs which in turn lowers the precision of the likelihood function. Also, in case of vehicles’ movement, due to the higher speed of movement, this issue becomes much more significant in comparison to the pedestrians’ movement.

To address these issues, changes in both offline and online phases are proposed. In the offline phase, a hybrid learning database is implemented to overcome the movement speed issue. Furthermore, an ensemble neural network [17] for the online phase likelihood function is deployed to solve the high variance signals problem.

A. Hybrid Database Offline Phase

In the classical approach of Wi-Fi fingerprinting localization, the mapping \(\{X_i, \rho_i\}\) in the offline phase is perceived as a representative feature vector \(X_i\) of the fingerprint \(\rho_i\). The underlying assumption for this mapping is at a fixed location, its feature vector is unique. However, as this database is used for evaluating the online phase likelihood function, there is an issue with the online input data.

Consider a scenario as in Fig. 1, the fixed location fingerprint is defined in the offline phase and illustrated as the circle in the figure. With 1Hz of sampling frequency, a Wi-Fi scan is initiated at a certain position and terminated at a different one. The distance between two locations of scan initiation and termination is called a scan range. Depending on the movement speed of the target, the scan range can also vary.
With an average of 1.35 m/s for human walking speed, the scan range for a human walking case is also around 1.35 m. Thus, a signal vector \(X_t\) in the online phase is the feature vector of a path but not that of a fixed position. By reducing this path to a fingerprint location in the likelihood function, an error of localization is introduced. When it comes to the targeted scenario in this paper, a vehicle in a carpark moves at a 3.5 m/s average speed which results in an approximately 3.3 m of scan range. This is a significant distance for the vehicle navigation task. Hence, a new hybrid database of offline phase database is proposed: instead of only collecting feature vector data at a fixed position (i.e., a fingerprint), the vehicle will move around the fingerprint location and collect the Wi-Fi signals. This create a new mapping \(\{X_t, \rho'_i\}\) where \(\rho'_i\) is expressed as:

\[
\rho'_i \in (\rho - \Delta \rho, \rho + \Delta \rho).
\tag{7}
\]

This equation models the new fingerprint as a circle which takes location of \(\rho\) as the center and \(\Delta \rho\) as the radius. Having this new fingerprint definition, a mix of fixed location scans and moving vehicle scans is collected to constitute the new feature vectors of the area. Scans with vehicles moving are called dynamic scans. Others with vehicles at fixed place are called static scans. Note that an assumption of symmetrical distribution of error in both x-axis and y-axis is made. In the best-case scenario for the new fingerprint concept, the center of the scan range will be exactly at the location of \(\rho\). Therefore, the radius \(\Delta \rho\) is determined as:

\[
\Delta \rho = \frac{\bar{v}_{\text{vehicle}}}{2}.
\tag{8}
\]

Finally, a normalization of collected RSSIs in each scan is performed to reduce the numerical impact of signal strength feature. A detected access point will have RSSI in range [0,1] after normalization while an undetected access point (i.e. access point which is inside the environment but not detected at a certain fingerprint) is scored as −1. This further emphasizes the weight of detected AP compares to undetected one. The formal representation of a RSSI normalization process is shown in (9):

\[
x_i = \begin{cases} 
-1, & \text{AP}_i \text{ undetected;} \\
1 - \frac{(-1) \times \text{RSSI}}{100}, & \text{AP}_i \text{ detected.}
\end{cases}
\tag{9}
\]

### B. Wi-Fi Ensemble Neural Network

In the online phase, a likelihood function \(h\) is required for evaluating the real-time input RSSIs vector. Previous works in this phase often make use of classical statistical model such as KNN (K nearest neighbours), Random Forest, SVM (Support Vector Machine) etc. There are also approaches from the deep learning method such as [18], [19]. However, the high variance of the collected RSSIs is still one of the main reasons for the low accuracy of Wi-Fi fingerprinting localization. Hence, an ensemble neural network is proposed in this study to address the issue.

The idea of using multiple learning models to enhance performance of a single one is proposed in [20], [21]. Under certain conditions, the combination of diverse, uncorrelated but accurate estimators should have better performance than one alone [17]. A necessary and sufficient condition for an ensemble estimator to be more accurate than any of its individual is each individual estimator should be accurate and diverse [22]. An accurate estimator is the one that has an accuracy rate better than a random guessing on new input value. Two estimators are diverse if they make different errors on new data. Thus, in order to construct an ensemble of neural network, the study proposes to employ a bagging technique (Bootstrap Aggregating) [21]. The formal explanation of applying this technique for ensemble neural network is following.

Consider a classification problem with a pair \((x, y_j)\) where \(x\) a feature vector and \(y_j\) denotes a response of a corresponding class, \(y_j \in \{1, 2, \ldots, m\}\). The target function is \(P(y = j | x)\). Given a training database with the number of samples \(n\) and the number of classes \(m\), an approximation function \(h\) is formed in (10):

\[
g(\cdot) = h((x_1, y_1), (x_2, y_2), \ldots, (x_n, y_m)) \tag{10}
\]

Multiple bootstrap databases are formed by randomly sampling with replacement \(K\) times from the original data. Each database has equal size of \(n\) samples as:

\[
(x_{1'}, y_{1'}), (x_{2'}, y_{2'}), \ldots, (x_{m'}, y_{m'}) \tag{11}
\]

Compute \(K\) bootstrap estimators using the same approximation function \(h\) for neural network classification model with each of \(K\) bootstrap database as training data:

\[
\hat{g}(\cdot) = h((x_{1'}, y_{1'}), (x_{2'}, y_{2'}), \ldots, (x_{m'}, y_{m'})) \tag{12}
\]

Aggregating all \(K\) estimators, we have:

\[
\hat{g}_{\text{bagging}}(\cdot) = \frac{1}{K} \sum_{i=1}^{K} \hat{g}_i(\cdot) \tag{13}
\]

As \(K\) goes to infinity, result in (14) is closer to the expected value:

\[
\hat{g}_{\text{bagging}}(\cdot) = E[\hat{g}_i(\cdot)] \tag{14}
\]

In practice, with a large finite \(K\), the bagging estimator should come close to the expected global optimal.

The Wi-Fi fingerprinting localization problem is now viewed as a standard supervised classification problem where a newly scan vector \(x\) of RSSIs in the online phase will be fed into each bootstrap estimator as in (15). The returned result of each classifier is a list of fingerprints IDs \(FP_i\) and its confidence \(\hat{c}_i\). Note that \(l\) is the number of fingerprints in the entire environment.

\[
\hat{g}(x) = \{FP, \hat{c}_i | i = 1, 2, \ldots, l\} \tag{15}
\]

The final aggregated ensemble results are:

\[
\hat{g}_{\text{bagging}}(x) = \{FP_{i}, c_i | i = 1, 2, \ldots, l\} \tag{16}
\]
where the bagging confidence is calculated as:

\[ c_i = \frac{\sum_{j=1}^{K} \tilde{c}_{ij}}{\sum_{j=1}^{I} \sum_{i=1}^{K} \tilde{c}_{ij}}. \] (17)

It is also important to mention the structure of the neural network. For this solution, we employed a fully connected architecture with a single hidden layer neural network. In a particular environment, the number of distinct MAC addresses collected in training data is considered as the number of inputs \((n\ln)\). The number of outputs \((n\text{Out})\) for this neural network is the number of chosen fingerprints in the environment (number of labeled positions in training phase). A two-third rule is used to determine the number of neurons in the hidden layer. This result in the number of hidden neurons can be calculated as in (18). Then a softmax activation function is employed for the classification task of the network. It is because the softmax is known for good multi-class classification. Finally, the cost function chosen for this task is the cross-entropy error. This cost function is suitable for multi-class classification with the posterior-probability vector as the final outcome. In addition, it is intentional to loosely fine-tune each neural network in this paper to avoid overfitting or data-sensitive side effect.

\[ n\text{Hidden} = \frac{2}{3} (n\ln + n\text{Out}) \] (18)

**IV. GAUSSIAN MIXTURE PARTICLE FILTER**

In localization problem, there are several algorithms dedicated to sensors fusion such as Kalman filtering, grid-based Markov localization, Particle filtering or Neural Network etc. However, since the problem of tracking and localization is often a non-linear problem with a huge solution space, machine learning approaches as well as classic Kalman filter do not always work. Statistical approaches suffer from noisy and biased data. A review of several non-linear filtering approaches can be found in [23]. Among them, particle filter appears to be a stand-out solution for nonlinear low dimensional problems. Thus, the particle filter, a non-linear filtering approach is chosen to be the fusion solution.

The particle filter has four steps: initialization, prediction, correction and selection & resampling. Instead of assuming the linearity of states as well as zero-mean Gaussian distribution of dynamic noise, the particle filter tries to predict the solution space with a propagation model on a discrete set of states known as particles. A correction phase is applied each time based on observation to weight each of the particles and approximate the assumed distribution. In this way, the particle filter can deal with non-linear and non-Gaussian process.

In classic particle filter, particles are initialized randomly all over the solution space. This ensures that the filter could recover from bad observations. However, at the same time, the randomly initialization increases the time for the particle filter to converge. For the online particle filter, this initialization step needs to be optimized. Instead of randomly picking particles in the entire solution space in the initialization step, particles are now generated based on the first observation obtained from the Ensemble Neural Network (ENN) for Wi-Fi fingerprinting localization. As presented in previous section, the ENN will return a list of fingerprints and their corresponding confidence scores \([FP_i, c_i|i = 1 : m]\). At this step, an aggregated sum of top \(k\) highest confidence fingerprints is calculated and assigned as the expected position as in (19).

\[ \mu = \sum_{i=1}^{k} FP_i \tilde{c}_i. \] (19)

Particles are then drawn from a Gaussian distribution \(N(\mu, \sigma_{Wi-Fi})\) with \(\sigma_{Wi-Fi}\) is estimated from the Wi-Fi fingerprinting localization experiments. Using this initialization process, the particle filter will converge quicker due to the reduction of the solution space and still maintain its correctness.

For a localization problem, the prediction step of particle filter usually embraces a motion model to move the particle cloud. Subject to the problem requirement, different motion models \(M\) are chosen to propagate particle \(x_i^t\) to \(x_i^{t+1}\). At this stage, there is no change in weight of each particle. More often, the vehicle’s velocity \(v\) and heading angle \(\theta\) are required for this step as in (20).

\[ x_i^{t+1} = M(x_i^t, v, \theta). \] (20)

In this paper, a simple constant speed motion model with assumed Gaussian noise is chosen.

\[ \left( \begin{array}{c} \tilde{x}_i^t \\ \tilde{\theta}_i^t \end{array} \right) = \frac{1}{\Delta t} \left( \begin{array}{ccc} v_i^t + \frac{v_i^{t-1} + v_i^t}{2} \cos(\theta) \\ v_i^t + \frac{v_i^{t-1} + v_i^t}{2} \sin(\theta) \end{array} \right), \] (21)

where the velocity, heading angle and noises are calculated as:

\[ \left( \begin{array}{c} \tilde{v}_i^t \\ \tilde{\theta}_i^t \end{array} \right) \sim N\left( \left( \begin{array}{c} v_i^t \\ \theta_i^t \end{array} \right), \left( \begin{array}{cc} \sigma_v^t & 0 \\ 0 & \sigma_\theta \end{array} \right) \right). \] (22)

In case there is an observation available for the correction step, the correction will be carried out to update particles’ weight. Here, a likelihood function is employed to be the scoring function for weight updating. For the Bootstrap particle filter version [24], the particles’ weight can be calculated directly from the likelihood function \(P(\tilde{z}|x_i^t)\) with \(\tilde{z}\) as the output vector (the estimated vehicle’s state at time \(t\)). Given that observation \((\mu, \sigma)\) is available with \(\mu\) is the expected true position and \(\sigma\) is the standard deviation of the observation, particles’ weight is calculated as:

\[ w_i^t \approx w_i^{t-1} P(\tilde{z}|x_i^t, \mu, \sigma). \] (23)

Assume each particle is a potential pose of the target, substitute \(\tilde{z} = x_i^t\) we have:

\[ P(x_i^t|x_i^t, \mu, \sigma) = N(x_i^t, \mu, \sigma). \] (24)

Finally, the equation for weight updating is:

\[ w_i^t \approx w_i^{t-1} N(x_i^t, \mu, \sigma). \] (25)
In this paper, a Gaussian mixture model is applied for the observation in the correction step. Similar to the initialization step, a top \( k \) highest confidence fingerprints will be taken into account for the correction. (24) is then written as follows:

\[
w'_i = w'_{i-1} \sum_{j=1}^{k} \frac{c^j}{\sum_{j=1}^{k} c^j} N\left(x'_i | \sigma^j_i, \mu^j_i\right). \quad (26)
\]

Here, \( \mu^j_i \) is the \( j \)th highest confidence fingerprint \( FP^j \) and \( \sigma^j_i \) is related to the measured standard deviation of the Wi-Fi fingerprint as follows:

\[
\sigma^j_i = 1 - \frac{c^j}{\sum_{j=1}^{k} c^j} \sigma_{Wi-Fi}.
\]

The Gaussian mixture model is utilized to enhance the weighting function. Since the Wi-Fi fingerprinting localization has a low sampling rate as well as accuracy, taking multiple observations are expected to help the filter to recover from bad observations.

Lastly, in the selection & resampling step, an estimation of the vehicle current location can be derived in multiple ways. This step includes two small steps: a required selection step and an optional resampling algorithm.

For the selection step, the goal is to find the final estimation for the vehicle position given a particles cloud and their weight. This could be done by picking the particle with the highest score. However, choosing only the highest score particle is potentially misleading since the entire particles cloud represents the likely distribution of the estimation around the ground-truth. Therefore, an expected value could be calculated to approximately reflect the estimation of the particle filter:

\[
\hat{x}^t = E\left[x^t\right] = \sum_{i=1}^{N} x'_i w'_i.
\]

Finally, a resampling process takes place to eliminate particles that have small weight and to concentrate more on particles with high weight. This resampling process also consisting of generating a new set of particles by resampling with replacement (bootstrap resampling). Those newly generated particles allow the particle filter to overcome the degeneracy issue [25]. The newly formed particles cloud will be fed into the prediction step for the next phase of position estimation. There are many different resampling algorithms for particle filter. However, four original approaches can be identified as: Multinomial resampling, Stratified resampling, Systematic resampling and Residual resampling. Other algorithms are built based on the idea of those algorithms. A comparison of those algorithms can be found in [26]. As there is no clear winner among these, the most common one - multinomial resampling approach is adopted for this paper. The fusion strategy is shown in Fig. 2.

**V. EXPERIMENTS AND RESULTS**

Experiments for the proposed method are carried out in an open parking space of the INRIA Rocquencourt campus. Due to difficulties in having an indoor carpark for experiments, the outdoor space is utilized. At the same time, this outdoor carpark benefit from a precise RTK-GPS that is be used as localization ground truth and allows thus a better evaluation of the system. The testing area is shown in Fig. 3. There are two vehicles in the experiments: a Cybercar designed as a prototype for intelligent vehicles and a customized Citroen C1. Both vehicles are equipped with a Wi-Fi receiver, a RTK-GPS receiver, an IMU system and two LiDAR sensors (front and back). The average movement speed across all experiments is around 2.5m/s to 3.5m/s.

**A. WI-FI FINGERPRINTING LOCALIZATION EXPERIMENTS**

A total of 25 fingerprints are chosen to cover the testing area. The average distance between two adjacent fingerprints is 6.1m. For each fingerprint, 60 static scans and 20 dynamic scans are recorded for the offline database. A total 156 access points with different MAC addresses are detected across 25 fingerprints. A Wi-Fi heatmap for the experiment area (road only) is shown in Fig.4. In this heatmap, the darker the color, the lower average Wi-Fi signal strength recorded in the area. (the indicator goes from 0 to 1 maximum as the signal strength goes from −100dBm to 0dBm correspondingly).

The number of individual neural networks is \( K = 50 \). For each network, 10 random early restarts [27] in the training phase are applied in order to avoid local optimum.
In the classification approach for Wi-Fi fingerprinting localization, the fingerprint with the highest confidence score is selected as the final result if only its confidence score is greater than $\theta$: $c_{\text{max}} \geq \theta$.

A single test run for Wi-Fi fingerprinting localization is shown in Fig. 5. In total, a 64 test runs were conducted for one year. The final result is then compared with other methods as in Table 1. With 2.25m of average error, the ensemble neural network for fingerprinting localization is superior to other methods.

### B. FUSION FRAMEWORK EXPERIMENTS AND RESULTS

First, two types of the vehicle true initial position are tested: one within fingerprint defined area and one outside of any fingerprint defined area. One single test result of the first case is illustrated Fig. 6. The $y$-axis is the Euclidian distance localization error while each time step in the $x$-axis is equal to 1/10 of a second. With a good initialization position, the Gaussian mixture model particle filter shows a good performance in term of convergence and localization accuracy. The moving path can be seen in Fig. 7.

Another case for which the vehicle starts from outside of any fingerprint defined area is examined. The localization error is shown in Fig. 8. In this case, due to the initial position is outside of the fingerprinting area, the best observation obtained from the Wi-Fi fingerprinting localization has an error of 3.2m. From this position, the particle filter takes around 30-40 timesteps (3-4 seconds) to reduce the error around 40% then slowly converge to below 1meter of localization error after 30 seconds. This happens due to after few first second, the vehicle slowly moves to the area of the fingerprint which results in better observations from Wi-Fi fingerprinting localization. Together with a high number of particles count, the particle filter can recover from the high error start.

To evaluate the number of particles required, one experiment data is run repeatedly for 100 iterations (each iteration is...
independent from each other). Both two cases: the initial true position is within the Fingerprint (FP) area and outside the Fingerprint area are studied. The mean localization errors and standard deviation after 100 iterations with different number of particles count in the particles filter is shown in Table 2. For the initial position outside of the Fingerprint area case, a statistic excluding first 4 seconds, where the localization errors are stabilized, is presented. This is to study the effect of a good starting point on the overall localization error. In general, having a good starting point (within the FP area) shows a significant improvement in terms of localization accuracy. Also, the statistic shows that the system only needs around 2000 - 4000 particles to reach an optimal localization accuracy.

Finally, in total 64 experiments were conducted with a random starting position during nearly a year. This long duration of experiment proves that the system can deliver a stable result regardless of environment conditions. The cumulative sum of localization errors of all experiments is shown in Fig. 9a. In these experiments, the particle filter has 4000 particles. The mean error is estimated at 0.859m and standard deviation is 0.2320. The figure also shows that around 90.27% of the errors is under 1.5m which is one sigma ($\sigma_{\text{wifi}} = 1.5$). If we assume a good starting position is provided (which is realistic since the initial position of a vehicle entering a car park can be approximately predicted) then a much better result will be obtained. With only 2000 particles, the mean error is 0.5885m and the standard deviation is 0.1270m. Also, in this case, 98.81% of the errors are under one sigma ($\sigma_{\text{wifi}} = 1.5$) (see Fig. 9b).

We have also tested our fusion framework with the laser-SLAM output as odometry measurements. Since the SLAM is expected to be accurate within its local frame, the extracted velocity and heading change from the SLAM is supposed to be less noisy than a physical IMU. This is then used as a correction for the physical IMU data. The fusion with laser-SLAM resulted in a slightly improved localization result as shown in Fig. 10. With 2000 particles and a good starting position, the fusion with laser-SLAM has 0.49m of mean error (compared to 0.57m of the IMU fusion).
VI. CONCLUSION
In this paper, we propose a novel absolute localization system for intelligent vehicles in GPS-denied environments. The research focuses on the carpark situation with the average speed of 3.3m/s. Even though the Wi-Fi fingerprinting localization is not a novel technique in the indoor positioning, an adaptation for higher speed of movement using ensemble neural network was not explored before our study. The proposed solution shows a mean error of 2.25m and is outperformed other fingerprinting methods.

Moreover, a fusion framework using Wi-Fi fingerprinting localization, IMU, laser-SLAM is proposed. The goal of this framework is to enhance the sampling rate (output rate) as well as the accuracy of the localization system. With IMU fusion, a 0.57m of mean error is achieved given a good starting position. The SLAM and IMU fusion shows an improvement with 0.49m of mean positioning error.

With this result, the authors hope to develop a stable indoor navigation system for autonomous vehicle as well as other indoor robots. Still, improvements can be made in term of fusion strategy as well as the scoring function.

REFERENCES
[1] S. Belloche, “On-street parking search time modelling and validation with survey-based data,” Transp. Res. Procedia, vol. 6, pp. 313–324, Jun. 2015.
[2] E. Gantelet and A. Lefauconnier, “The time looking for a parkingspace: Strategies, associated nuisances and stakes of parking management in France,” Assoc. Eur. Transp. Contributors, Tech. Rep., 2006.
[3] Skyline Inc. Innovative Automated Parking Solution|Skyline Parking|Skyline Parking. Accessed: May 20, 2018. [Online]. Available: https://skyline-parking.com/automated-parking-systems/
[4] Parking Solutions—Solutions—oneSITU. Accessed: May 20, 2018. [Online]. Available: https://www.onesitu.com/solutions/onesitu-parking/?lang=en
[5] S. Barkema, A. Behrmann, and H. Barbeau, “Evidence-based therapy for recovery of function after spinal cord injury,” Handb. Clin. Neurol., vol. 109, pp. 259–274, Jan. 2012.
[6] A. Eskandarian, “Fundamentals of driver assistance,” in Handbook of Intelligent Vehicles, vol. 1. Springer, 2012, pp. 491–535.
[7] S. Wahl, P. Schlumberger, R. Rojas, and M. Stampfle, “Localization inside a populated parking garage by using particle filters with a map of the static environment,” in Proc. IEEE Intell. Veh. Symp., Aug. 2015, pp. 95–100.
[8] B. H. Groh, M. Friedl, A. G. Linarth, and E. Angelopoulou, “Advanced real-time indoor parking localization based on semi-static objects,” in Proc. Inf. Fusion (FUSION), 2014, pp. 1–7.
[9] J. Bojja, M. Krikko-Jaakkola, J. Collin, and J. Takala, “Indoor localization methods using dead reckoning and 3D map matching,” J. Signal Process. Syst., vol. 76, no. 3, pp. 1–12, 2013.
[10] U. Schwesinger et al., “Automated valet parking and charging for e-mobility,” in Proc. IEEE Intell. Veh. Symp., Aug. 2016, pp. 157–164.
[11] S. Houben, M. Komar, A. Hohn, S. Luke, M. Neuhause, and M. Schlipsing, “On-vehicle video-based parking lot recognition with fisheye optics,” in Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC ), Oct. 2013, pp. 7–12.
[12] F. de Ponte Müller, E. M. Diaz, and I. Rashdan, “Cooperative infrastructure-based vehicle positioning,” in Proc. IEEE Veh. Technol. Conf., Sep. 2017, pp. 1–6.
[13] J. Liu, R. Chen, Y. Chen, L. Pei, and L. Chen, “iParking: An intelligent indoor location-based smartphone parking service,” Sensors J., vol. 12, no. 11, pp. 14612–14629, 2012.
[14] R. Wilfinger, “Absolute positioning of vehicles in indoor environments using wireless LAN and Bluetooth LE,” M.S. thesis, Dept. Geomatics Sci., Inst. Geodesy, Graz, Austria, Apr. 2015.
[15] N. Hernandez, A. Hussein, D. Cruzado, I. Parra, and J. M. Armingol, “Applying low cost Wi-Fi-based localization to in-campus autonomous vehicles,” in Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC), Oct. 2017, pp. 1–6.
[16] F. C. Anderson and M. G. Pandy, “Dynamic optimization of human walking,” J. Biomech. Eng., vol. 123, no. 5, p. 381, 2001.
[17] T. G. Dietterich, “Ensemble methods in machine learning,” Multiple Classifi-

D. V. Nguyen received the Ph.D. degree from Mines ParisTech and RITS Team, INRIA Paris, in 2018, under the supervision of Dr. Fawzi Nashashibi and Prof. Eric Castelli. His research interests include indoor localization, wireless sensor networks, and sensors fusion.

TRUNG-KIEN DAO (Member, IEEE) has been a Lecturer-Researcher at the Department of Per-

99738 VOLUME 8, 2020
D.-V. Nguyen et al.: Fusion Method for Localization of Intelligent Vehicles in Car Parks

ERIC CASTELLI was born in 1962. He was an Associate Professor at the Institute National Polytechnique de Grenoble (Grenoble INP), from 1990 to 2011. He defended his Ph.D. thesis in electronic system field, in 1989 and his HDR thesis, in 1999. He worked at the Institut de la Communication Parlée (ICP), from 1984 to 1997, and then he joined the CLIPS Laboratory, from January 1998 to September 2001. From 2001 to 2017, he worked in Hanoi, Vietnam, as the French Director of the MICA International Research Institute, and In Charge of the Scientific Program Management. In 2009, he became a Professor with the Hanoi University of Science and Technology, which hosts the MICA Institute and was integrated permanently to CNRS, in September 2011, as a Tenured Researcher. He currently works at the LIG Laboratory, Grenoble, France, (Pervasive INRIA Project Team). He published over 250 articles on various aspects of speech analysis, speech production, speech recognition, ubiquitous computing, human–machine interaction, and general instrumentation. He has also been involved in several national and international projects. His scientific topics include speech processing, signal processing, ambient and ubiquitous computing, human–machine interaction, advanced instrumentation, and multimedia.

FAWZI NASHASHIBI (Member, IEEE) received the master’s degree in automation, industrial engineering, and signal processing from the Laboratory for Analysis and Architecture of Systems/Centre National de la Recherche Scientifique (LAAS/CNRS), Toulouse, France, the Ph.D. degree in robotics from Toulouse University prepared in the LAAS/CNRS Laboratory, and the HDR Diploma (accreditation to research supervision) degree from the University of Pierre et Marie Curie (Paris 6), Paris, France. Since 1994, he has been a Senior Researcher and the Program Manager with the Robotics Center of the Ecole des Mines de Paris (Mines ParisTech), Paris, and since 2000, he has been a Research and Development Engineer and a Project Manager at ARMINES. Since 2010, he has been a Senior Researcher and the Program Manager of the RITS Team, INRIA, Paris-Rocquencourt, France. He was previously a Research Engineer with PROMIP, working on mobile robotics perception dedicated to space exploration, and a Technical Manager with Light Company, where he led the developments of virtual reality/augmented reality applications. He played key roles in more than 50 European and national French projects, such as Carsense, HAVE-it, PICAV, CityMobil, ARCOS, ABV, LOVE, and SPEEDCAM, some of which he has coordinated. He is also involved in many collaborations with French and international academics and industrial partners. He is the author of numerous publications and patents in the field of ITS and ADAS systems. His current research focuses on advanced urban mobility through the design and development of highly automated transportation systems, which includes highly automated unmanned guided vehicles (such as cybercars) as well automated personal vehicles. In this field, he is known as an international expert. Since 1994, he has also been a Lecturer in several universities (Mines ParisTech, Paris 8 Saint-Denis, Leonard de Vinci University, Telecom Sud Paris, INT Evry, and Ecole Centrale d’Electronique) in the fields of image and signal processing, 3-D perception, 3-D infographics, mobile robotics, and C++/JAVA programming. His main research interests include environment perception and multisensor fusion, vehicle positioning, and environment 3-D modeling with main applications in intelligent transport systems and robotics. He is a member of the IEEE ITS Society and the Robotics and Automation Society. He is an Associate Editor of several IEEE international conferences, such as ICRA, IROS, IV, ITSC, and ICARCV.