

Research Article

An Indoor and Outdoor Positioning Using a Hybrid of Support Vector Machine and Deep Neural Network Algorithms

Abebe Belay Adege ,¹ Hsin-Piao Lin ,² Getaneh Berie Tarekegn,¹ Yirga Yayeh Munaye,¹ and Lei Yen 

¹Department of Electrical Engineering and Computer Science, National Taipei University of Technology, Taipei, Taiwan

²Department of Electronic Engineering, National Taipei University of Technology, Taipei, Taiwan

Correspondence should be addressed to Abebe Belay Adege; abbblybelay@gmail.com

Received 9 July 2018; Accepted 25 October 2018; Published 16 December 2018

Guest Editor: Almudena Rivadeneyra

Copyright © 2018 Abebe Belay Adege et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Indoor and outdoor positioning lets to offer universal location services in industry and academia. Wi-Fi and Global Positioning System (GPS) are the promising technologies for indoor and outdoor positioning, respectively. However, Wi-Fi-based positioning is less accurate due to the vigorous changes of environments and shadowing effects. GPS-based positioning is also characterized by much cost, highly susceptible to the physical layouts of equipment, power-hungry, and sensitive to occlusion. In this paper, we propose a hybrid of support vector machine (SVM) and deep neural network (DNN) to develop scalable and accurate positioning in Wi-Fi-based indoor and outdoor environments. In the positioning processes, we primarily construct real datasets from indoor and outdoor Wi-Fi-based environments. Secondly, we apply linear discriminant analysis (LDA) to construct a projected vector that uses to reduce features without affecting information contents. Thirdly, we construct a model for positioning through the integration of SVM and DNN. Fourthly, we use online datasets from unknown locations and check the missed radio signal strength (RSS) values using the feed-forward neural network (FFNN) algorithm to fill the missed values. Fifthly, we project the online data through an LDA-based projected vector. Finally, we test the positioning accuracies and scalabilities of a model created from a hybrid of SVM and DNN. The whole processes are implemented using Python 3.6 programming language in the TensorFlow framework. The proposed method provides accurate and scalable positioning services in different scenarios. The results also show that our proposed approach can provide scalable positioning, and 100% of the estimation accuracies are with errors less than 1 m and 1.9 m for indoor and outdoor positioning, respectively.

1. Introduction

In the Internet of Things (IoT) era, indoor and outdoor positioning plays various roles to find the location of people, mobile devices, and equipment. Because of the popularities of social networks and the widespread usage of mobile devices, demands for location-based services (LBS) are increased in both indoor and outdoor environments [1, 2]. Positioning can be considered as a key technology to IoT, since it uses to provide situation-aware services in various applicable areas [3, 4]. Human daily life is also becoming highly integrated with the IoT as the Internet attracts much attention with respect to the outlook of future life

and rapidly increases communication networks. Additionally, rapid technological growths cause to increase positioning services. In [5], positioning is one of the primary services in the IoT era.

Indoor and outdoor positioning can be applied in university, airport, military system, security system, Metro Service-Route Map (MRT), farming, and forest areas. Moreover, we can apply in diverse services, such as emergency management, track mobile users, context awareness, advertisement, environmental monitoring, military surveillance, and medical care applications [6]. According to [7, 8], positioning serves to address problems such as traffic loads in 5G networks. Therefore, indoor and outdoor positioning

requires enormous attention to be effectively implemented in industry and academia, since precise positioning is used as an enabling factor for the future IoT [4, 9].

There are two main positioning schemes: range-based scheme and range-free scheme [10, 11]. In the range-based scheme, the distances and/or angles are used for positioning. This scheme includes received signal strength (RSS), time of arrival (ToA), time difference of arrival (TDoA), and angle of arrival (AoA) as a means of positioning. However, the range-free scheme performs positioning using the connections between different devices or the pattern matching between different devices. It includes the centroid algorithm (CA), the distance vector hop (DV-hop) algorithm, the multidimensional scaling map (MDS-MAP) algorithm, and convex programming approaches. It is highly depending upon regular sensor deployment area, which is very difficult in irregular and hierarchical environments. According to [7, 10], range-based positioning is more accurate, simpler, and lower-cost than the range-free approach.

Wireless local area network (WLAN), infrared (IR), radio-frequency identification (RFID), ultrasound, and Bluetooth technologies are the main indoor positioning approaches. However, Wi-Fi is common in IoT eras, because we can mount the Wi-Fi connections in various locations without any new infrastructures and additional cost. Besides, data collection using Wi-Fi devices is cheaper, for example, using access points (APs) [12].

As discussed in [11], GPS and the standalone cellular system are the most promising and accurate positioning technologies for outdoor positioning. According to [13], an outdoor positioning can be done by using the ubiquitous mobile network Base Stations (BS). However, the accuracy in outdoor positioning is highly depending on the number of surrounding BS [14]. The highly irregular and nonstatic environments easily affect the positioning accuracies in an outdoor environment, since even small changes lead to significant changes in the corresponding scans. GPS-based positioning has limitations to outdoor navigation and experience severe signal loss in urban areas [15]. Outdoor positioning using fixed sensors or GPS-based sensors is very expensive [8]. Hence, both indoor positioning and outdoor positioning require special attention to be accurate and have cheaper application.

There are two types of Wi-Fi-based positioning technologies [16]: time and space attributes of received signal- (TSARS-) based technology and received-signal strength- (RSS-) based positioning technology. The former approach includes TDoA, AoA, and ToA. This type of approach has many difficulties when measuring the radiofrequency (RF) signal in a complex environment, so that it affects positioning accuracies. Contrarily, RSS-based positioning technology uses RSS to find the users' position. RSS-based positioning technology has advantages of lower cost, fewer operational complexity, lower power consumption, and easiness in mapping into different areas [12, 16]. Positioning using RSS in wireless environments helps to provide location services without additional sensor costs. In wireless positioning techniques, we can possibly develop a system through error minimization in cheaper cost and lower computational

time. Generally, Wi-Fi-based positioning can be functional anywhere wireless services are available.

As pointed in [16], there are three kinds of RSS-based positioning: trilateration, approximate or similarity perception, and scene analysis. The trilateration approach requires at least three APs to convert received signal into spatial distance. However, it is very difficult to convert signal into distance in complicated and hierarchical environments. The approximation-perception method is relatively simple although it has lower positioning accuracy. Scene analysis, also known as fingerprint matching, does not require the locations of the APs. In this approach, the algorithms can obtain the precise position of Wi-Fi users. The scene analysis is characterized by low cost, high precision, and low-energy consumption, so that this method gets much attention [7, 8, 12, 16].

According to [17], Wi-Fi-ranging and Wi-Fi-fingerprinting are the two main methods to deliver Wi-Fi-based positioning. The former approach is implemented by measuring the distance towards the node. This type of approach is impractical inside complex, dynamic, and hierarchical environments due to numerous signal obstructions. Conversely, in Wi-Fi fingerprinting methods, positioning is operated by comparing the current data through the pre-recorded datasets. This approach requires larger amounts of stored datasets. The wireless signal variations are not also considered, which affect positioning accuracies.

Previous researchers have proposed different machine learning methods, such as radial basis function network and particle filter [3], support vector regression (SVR) [9], artificial neural network (ANN) [18], K-nearest neighbors (KNN) [19, 20], feed-forward neural network (FFNN) [21], weighted KNN (WKNN) [22], and KNN and ANN backpropagation (ANNBP) [23] to maintain positioning accuracies. In [3], radial basis function network and particle filter algorithms did the positioning and tracking. The authors used RFID and IR as a means of data sources using sensors. This type of approach requires higher cost, since it needs additional hardware and requires aggressive data collection processes. In [9], SVM and ANN were used to locate in an indoor environment. This work focused to locate APs in an indoor environment, and outdoor positioning was not considered. An ANN algorithm and ToA and AoA data sources were used for indoor positioning [18]. This approach has operational complexity, and it is difficult to locate accurately because of line-of-sight (LoS) problems. In [20], KNN was used to estimate the current position of a mobile user using historical data. This approach is very difficult to get accurate positioning values, since RSS is highly time-dependent. The KNN and Kalman filter algorithms were applied for indoor positioning [19]. This work required extra hardware for data collection.

In [24], the combination of indoor and outdoor positioning was done in athletic training and gymnasium areas. The authors used GPS with microelectromechanical system (IMU-MEMS) technology for outdoor positioning and ultra-wideband (UWB) with IMU-MEMS technology for indoor positioning to provide high precision positioning services. In [15], single body-mounted camera and computer

vision techniques were applied for outdoor positioning. This approach is too expensive because of requiring extra hardware and much storages. Authors in [25] compared GPS, global system for mobile communication (GSM), WLAN, and Bluetooth to apply to indoor and outdoor positioning, and the result showed that GPS and WLAN offer the most harmonizing accuracy. However, this work required extra terminals to integrate GPS and WLAN devices.

Positioning accuracy, in complex environments, using shallow-learning approaches is usually unsatisfactory in larger datasets, because the algorithms learn poorly. More specifically, traditional approaches have poor learning capabilities in complex and dynamic wireless environments, since there are many Wi-Fi-signal changing factors, such as multipath fading, attenuation of objects, or non-line of sights.

In this paper, we propose to use the integration of FFNN, linear discriminate analysis (LDA), SVM, and deep neural network (DNN) algorithms for scalable and accurate positioning, as proposed in our previous work [26]. We use LDA for dimension reduction, as well as hybrid of SVM and DNN as a means to locate the actual position of smartphone users. Additionally, we apply the FFNN algorithm for automatic bursting the missed RSS values before LDA is applied in both training and testing phases. If there will be more than one feature having missed RSS values, then the proposed system fills RSS values in each feature iteratively. The use of IoT helps to enhance accuracy, minimize computational time, and yield benefits of larger numbers of datasets. To find the target of smartphone users, we use a hybrid of SVM and DNN algorithms. The performances of our proposed method are evaluated in different scenarios, which helps to evaluate the scalability and the accuracy of our system performances.

The main contribution of this paper is to provide universal positioning services as it focuses on indoor and outdoor positioning, to deliver accurate positioning as we apply IoT and minimize positioning time complexities. Additionally, the proposed approach provides scalable positioning whenever trained APs are unable to offer Wi-Fi signal. To the best of our knowledge, this article is the first to present the integration of IoT, such as FFNN, LDA, SVM, and DNN algorithms to provide scalable and accurate positioning in indoor and outdoor environments.

The rest of this paper is organized as follows. In Section 2, we discuss related works. In Section 3, we discuss data collection approaches. Section 4 presents the details of the proposed technique. The results and discussions are presented in Section 5. Finally, conclusions of the work are given in Section 6.

2. Related Works

Indoor positioning and outdoor positioning using traditional machine-learning techniques in a Wi-Fi-based dynamic environment are not accurate and robust because the machine is unable to adapt to signal oscillations, noises, and radio signal fluctuations. DNN has been proposed as a new strategy due to the fact that it helps to handle traditional learning problems. The discussion in [12] shows that DNN

is used for easy adaptation of data variations, management of the dins of Wi-Fi signal, and device and time obsessions of wireless signal because it has advanced learning capability from complex and larger datasets. According to [17, 27, 28] discussions, the DNN is used to reduce positioning workloads, to improve the accuracy of Wi-Fi-based positioning, and to provide efficient positioning services, because DNN has deeper learning capabilities and efficient prediction performances.

In [29], the four-layered DNN was proposed for indoor and outdoor positioning. In this work, the stacked denoising autoencoder (SDA) and Hidden-Markov model (HMM) were used to minimize features without affecting information contents and to smooth the original locations, respectively. The performance of the work was evaluated using root mean square errors (RMSEs) in various testing sets. In this work, the scalability issue was not addressed for indoor and outdoor positioning. Additionally, system performances for indoor and outdoor positioning were evaluated at different numbers of layers and neurons, which makes it difficult to reach concise points about system performances. Kim et al. [12] used DNN for indoor positioning via radio signal values. In this work, system performance is evaluated by comparing epoch sizes only. However, other machine learning factors such as gradients and batch sizes were not considered. In [17], the deep learning algorithm was applied for building and floor-level classification using publicly offered data. In this work, SDA and deep learning were used for dimensional reduction and classification-based positioning, respectively. In [30, 31], deep learning was applied for indoor positioning. In these papers, researchers used channel state information (CSI) as means of data sources. However, this approach is very difficult to validate the system performances since the actual target is not clearly known because of data natures.

The FFNN-based algorithm was applied for indoor positioning using datasets collected from multiple buildings and floors experimentally and attained 99.82% and 91.27% accuracies, respectively [32]. This work did not focus the outdoor scenarios. In [33], classification-based positioning was applied using RSS values. In this work, the largest RSS values were selected to train the positioning algorithm. However, this approach is very problematic to evaluate when there are similar RSS distributions. In [34], the radial basis function (RBF) was applied to locate the mobile users in an indoor environment only. Since AoA was used as a means of data collection approach, the proposed technique has LoS problem. In [35], multilayer perceptron (MLP) was applied for indoor positioning. A limited number of APs were used in a building. Additionally, the scalability and robustness of the proposed method were not considered. In [36], MLP and KNN were used as a range-free positioning approach to select the better algorithm and found that MLP has better performances in different sample sizes.

Petr et al. [13] used UWB to indoor positioning and real-time kinematic global navigation satellite system (GNSS) through several BS to outdoor positioning. This work used TDoA to collect required datasets. In [37], indoor positioning and outdoor positioning were operated using two sensors, and system performance was assessed

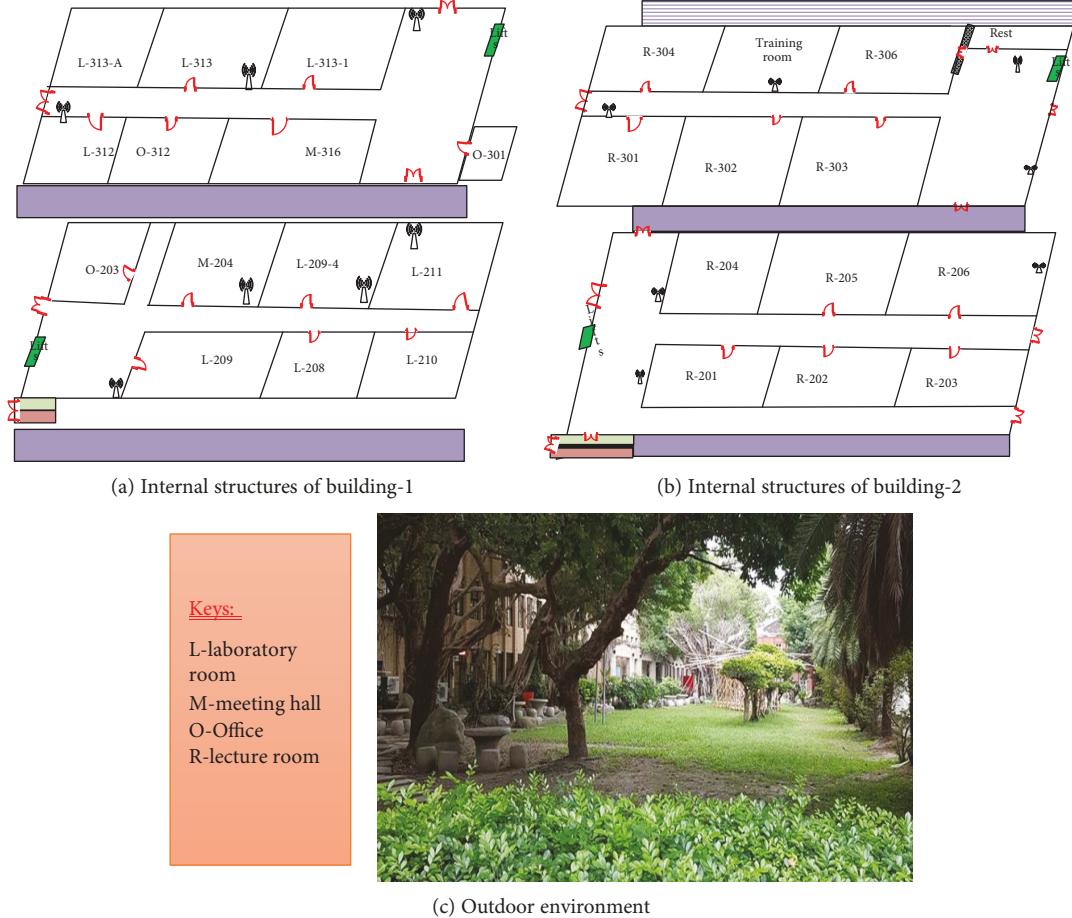


FIGURE 1: Structures of the indoor and outdoor environments.

using high-speed robot Kurt 3D equipped with the 3D laser scanner. This approach is accurate positions; however, it is not cost-effective because of using robots and sensors. In [2], a distributed-based positioning was implemented for indoor and outdoor positioning. This approach used infrared sensors and a GPS receiver deployed in working environments. The range-free approach using the DV-hop algorithm in DNN technology was implemented to locate positions [6]. This work focused on range-free positioning, which causes high computational complexity in a larger number of users. The authors in [38] combined the SVM and ANN to estimate users' position using RSS values. Scholars showed that the combination of SVM and ANN increased the accuracy rate by 5% than using SVM alone. This work focused on boundary-level localizations rather than showing a specific location of smartphone users. In [39], device-free passive localization (DfPL) by applying SVM and MLP was done. This work focused on identifying whether there are more than one person in a certain bounded location. Authors used wireless sensor networks (WSNs) with DfPL, which causes computational cost problems.

This paper is the extended part of our previous work [26], which focused on the application of LDA for feature reduction, and MLP for classification and regression in an indoor

environment. The scalability of positioning in an indoor and outdoor environment was not addressed, and they left as future works. The application of IoT was not also addressed. In [29], the DNN was applied to increase the estimation accuracy and reduce generalization errors on a dynamic indoor environment. This was done using only six APs and fewer datasets in a building. The IoT was not used to enhance positioning performances. The aim of our work is to apply IoT for scalable and accurate positioning in indoor and outdoor environments. To achieve our goal, we use FFNN, LDA, SVM, and DNN iteratively. As far as the authors' knowledge, this work is originally presented for the first time.

3. Working Environments and Experimental Data Collection

For this work, the datasets are collected from the real environment in National Taipei University of Technology (NTUT). The indoor environment's internal structures and physical layout of the outdoor environment are demonstrated in Figure 1. For indoor positioning, we use two buildings: complex building (building-1) and the academic building (building-2) as presented in Figures 1(a) and 1(b), respectively. In building-1, we consider workshops, halls,

and staffs' offices. In building-2, lecture rooms are considered. Additionally, the corridors of each building are considered as working places. For outdoor positioning, we consider the selected green area found in the university, as shown in Figure 1(c). The RSS and basic service set identifier (BSSID) datasets are used as data sources for the proposed positioning approach, since collecting RSS and BSSIDs from APs does not require additional infrastructure and hardware. In both indoor and outdoor working environments, datasets are collected from each reachable APs using smartphones.

For easily managing and collecting larger training datasets, we use $1\text{ m} \times 1\text{ m}$ grids in both indoor and outdoor environments. This results in making the DNN easily adapt to the data fluctuations. In [3], the performance of machine-learning techniques in different grids was compared: $1\text{ m} \times 1\text{ m}$, $1\text{ m} \times 2\text{ m}$, $1\text{ m} \times 1.5\text{ m}$, $2\text{ m} \times 2\text{ m}$, and $2\text{ m} \times 2.5\text{ m}$ grids. The $1\text{ m} \times 1\text{ m}$ grids have better performance than others do. The schemes in [40] used $1\text{ m} \times 1\text{ m}$ grids.

In this paper, we use similar grid sizes to evaluate the proposed system uniformly in both indoor and outdoor environments at a time. Moreover, it helps to develop a system that serves equally for indoor and outdoor positioning. In the indoor and outdoor environments, we collect data through an experiment from a real-world environment. In [21], authors discussed that the use of numerous datasets at each grid helps to adapt to signal fluctuations easily. Hence, we collect 35 RSS values and 35 BSSIDs from each grid in 5 s intervals, periodically. The BSSIDs are used to identify each AP uniquely. It also helps to identify the missed RSS values in the testing phase. For this work, a total of 21,665 RSS values and 21,665 BSSIDs are collected from 619 grids in the indoor environment. Moreover, we collect 4200 RSS values and 4200 BSSIDs from 120 grids in the outdoor environment.

For this work, all reachable APs from unknown locations are used for data collection purposes to make the system inclusive and to train the machine deeply from various signal values. In complex and hierarchical environments, the AP locations may not be known. However, the datasets reached from each AP may be more important. Additionally, seven APs are used for data collection. During data collection, about 53 RSS values are reachable from indoor environments, and there are more than 94 APs that reached the outdoor environment. The signal that reached the outdoor environment is coming from different buildings around the working location. This implied that considering APs from unknown locations is very important in obtaining basic information.

As a general structure of the collected datasets, the RSS values and BSSIDs are the basic features, and the $x-y$ -coordinates are Wi-Fi users' locations that are recorded during data collection. The scanned datasets are recorded in the form of $[(R_k^1, \dots, R_k^p), (BID_k^1, \dots, BID_k^p), (X_i^a, Y_j^a)]$, where (R_k^1, \dots, R_k^p) indicates the RSS values from AP₁ to AP_p at record $k \leq N$, $(BID_k^1, \dots, BID_k^p)$ are the recorded BSSIDs from AP₁ to AP_p, (X_i^a, Y_j^a) are the locations where RSSs and the BSSIDs are scanned, p is the reachable APs, k indicates the number of records, and i and

j refers to X- and Y-coordinates of mobile users, respectively. In the experiment, we have seen that p ranges up to 53 and 94 for indoor and outdoor environments, respectively. However, the data collected from an outdoor environment has much signal fluctuation than data collected from an indoor environment. The record k ranges from 1 to 25,865 records. We use similar data structures of RSS and BSSID values for indoor and outdoor positioning scenarios. We use the (x, y) target values to handle the position of smartphone users.

4. Proposed System

In this section, we propose a hybrid of SVM and DNN algorithms as a positioning scheme at indoor and outdoor Wi-Fi-based environments. The discussion in [38] shows that SVM produces robust classification results. Moreover, SVM can be modeled with kernel functions like SVM-RBF kernel to handle nonlinear datasets. DNN is a deep machine-learning technique, which has better learning capability than other shallow-learning approaches [29]. Hence, we propose to combine the two technologies to find accurately the final target of the smartphone users. We have a number of reached APs in an outdoor environment than in an indoor environment. However, we only take the first 43 APs as input features to the LDA, because the signal values after the 43rd APs are more fluctuated in the outdoor environment and their values ranged below -85 dBm. Additionally, RSS values beyond 43 are -100 in an indoor environment. The missed RSS values from the first 43 APs are filled automatically using the FFNN algorithm through iterative regression. Hence, we used only the first 43 reached RSS values as input features to LDA.

In [41], the missed AP values were filled using the average RSS values. Nevertheless, replacing the missed AP values with a similar value is impractical to get accurate positioning in dynamic environments. In [42], the authors used selective APs for localization. This type of approach is very difficult to implement whenever they miss the larger numbers of APs. In our proposed system, the missed RSSs are bursting using the FFNN algorithm through iterative regression. In the FFNN application, the features with nonmissed RSS values are used as input features, and a feature with missed RSS values is taken as the target of the FFNN before LDA is applied in the training process. In this case, the missed records in a feature are filled at a time, since using a larger number of target-features causes to decline the accuracy. We observed that the majority of missed APs are found in the indoor environment than in the outdoor environment. The process proceeds until the empty values are packed appropriately. Once all data are appropriate, we apply LDA for dimensional reductions, since LDA is fast, accurate, and easy for our datasets. FFNN in the testing process is applied by taking the offline data as input features and the online data with missed RSS values as the target features.

The authors in [41] point out that the datasets with high dimension have drawbacks in positioning systems because it requires higher computational time and consumes larger

storage spaces. Moreover, it causes overfitting problems. Through appropriate application of dimensionality reduction techniques, it is possible to project a set of high-dimensional vector samples into a much lower dimensionality while preserving relevant information of the data. This is because the dimensionality reduction techniques are mainly used to reduce the redundant and dependent features by transforming higher dimensional features to lower-dimensional spaces. The authors in [43] proposed LDA to extract the most discriminative features by increasing the distance between classes under the constraint of keeping within-class scatter values. LDA is a fast and less complex approach for dimension reduction, so that it helps to improve computational complexity [41].

There are two main LDA techniques depending on the projected vector preparations: class-dependent and class-independent [41]. In class-dependent, the lower-dimensional space is calculated for each class independently. The number of projected vectors depends on the numbers of classes. This type of approach is complex for a large numbers of classes. However, class-independent LDA operations are computed by using one lower-dimensional space for all classes. This approach is simple and easy in complex and nonlinear datasets. To reduce dimensions of our data vectors, we propose to use a class-independent LDA technique because of its simplicity and ease.

The collected datasets are coded in different classes based on LoS, geographical settlements, working environments, and reference points. For example, we consider a room and a corridor in the same structure and the same LoS as a class. An outdoor environment is also considered as a class. Totally, we formulate the working environments in 10 classes. After the datasets are coded in classes and after bursting the missed RSSs, we apply LDA for feature reduction. In the dimensional reduction processes, we apply LDA to find the refined projected vector, the class mean, the total mean of the datasets, the between-class matrix, and the within-class matrix. Then, any data from a similar environment can be projected with the final lower-dimensional space to find the reduced and simpler vector for further position determination. The LDA is applied as shown in Equations (1), (2), (3), (4), (5), and (6). The datasets collected from each working area are represented as D_{nxm} , where n and m are the numbers of records and features, respectively.

- (1) Assign each RSS vectors to the corresponding class C_c :

$$C_c = M_{rxN}, \quad (1)$$

where r is the record size in class C_c , and the class number c is ranged in [1, 10]. In RSS vectors, the record number in each class is always not greater than the total records.

- (2) The mean to each class is calculated as shown in Equation (2) to demonstrate the effects of features in independent classes. The computation is done from the assigned data to each class, and then, finally,

we obtained a $1 \times M$ matrix, where M is the total classes. Hence, we have a total of 1×10 mean vectors:

$$\mu_c = \frac{1}{n_c} \sum_{Di \in C_c}^n D_i, \quad (2)$$

where D_i is the original vector, $i \in c$, and μ_c is the mean value of individual class.

- (3) The global mean, μ , is calculated from all datasets:

$$\mu = \frac{1}{n} \sum_{i=1}^n D_i, \quad (3)$$

- (4) The variances of the between-class matrix are computed using the class mean and the global mean as shown in (4). This is mainly used for lowering dimensions and maximizing variances between classes:

$$S_b = \sum_{c=1}^{10} n_c (\mu_c - \mu)(\mu_c - \mu)^T. \quad (4)$$

- (5) Calculate the variances of the within-class matrix, which is conducted by minimizing the difference between the projected mean and the projected samples of each class as shown

$$S_w = \sum_{c=1}^{10} \sum_{i=1}^{n_c} (D_{i,c} - \mu_c)(D_{i,c} - \mu_c)^T, \quad (5)$$

where i th and c th are the sample and the class in matrix $D_{i,c}$, respectively.

- (6) Equation (4) and Equation (5) are combined to generate a lower-dimensional space:

$$W = S_w^{-1} S_b. \quad (6)$$

After calculating the lower-dimensional vector W , the representative features are selected as shown in Figure 2, where the larger variances are the most representative of the whole datasets. According to [44], the information contents of the nominated features should not be less than 90% of the whole information contents. Our chosen features comprehend the first top five features, which cover more than 98% of the information contents, as shown in Figure 2. This shows that the information content of the original datasets is not affected and its information content remains the same with original datasets. Once obtaining the lower-dimensional projected vector, we can transform any data from similar environments through the developed vector. The transformed vector should be the features that can denote all datasets to apply positioning algorithms.

We use the selected RSS features and five BSSIDs to both indoor and outdoor positioning. The five BSSIDs are selected

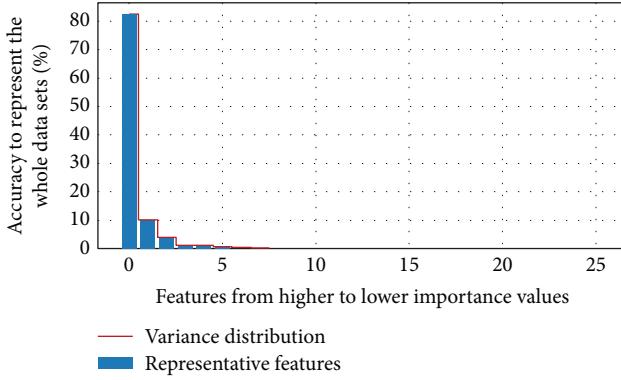


FIGURE 2: Feature distributions for LDA-based feature selections and the data variances.

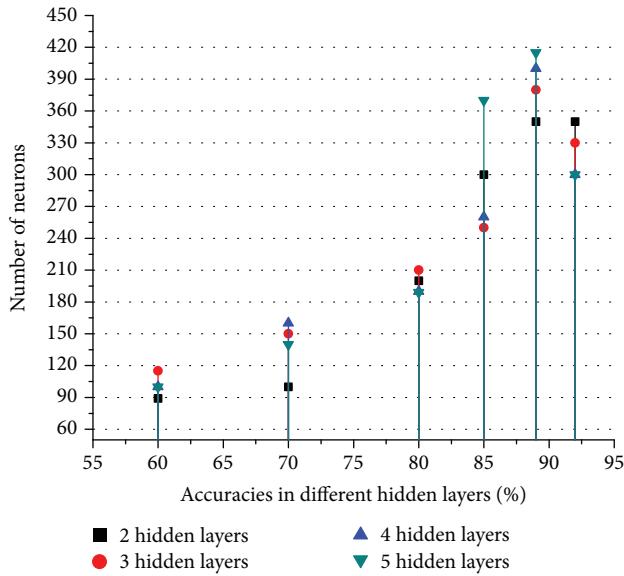


FIGURE 3: DNN performance with respect to numbers of hidden layers and numbers of neuron.

with the strongest RSS values, since AP-BSSIDs are unique and the signal strength in different locations varies for the APs. This helps to minimize the nonlinearity of RSS distributions. To evaluate the proposed approach, we select 70 locations from the indoor environment, and 8 locations from the outdoor environment. All testing locations are selected randomly from unknown locations. Our deep machine has an $I_{300_300_300_300_O_n}$ structure, where I shows the input vector, and O_n is the corresponding output. The four hidden layers of the network are selected as an appropriate layer to our datasets, as shown in Figure 3. The structure of the DNN is selected by comparing numbers of hidden layers, such as two, three, four, and five hidden layers. Moreover, the numbers of neurons and the corresponding system performances are considered.

As illustrated in Figure 3, the numbers of hidden layers with respect to training accuracies and number of neurons achieve better accuracy at four hidden layers and 300 neurons. In five hidden layers, the accuracy is the same; however,

TABLE 1: Classifier algorithm performance comparisons.

Classifier algorithms	Average performances	
	Average precision	Average recall
1 KNN	0.92	0.94
2 ANN	0.91	0.96
3 SVM	0.99	0.99

we select four layers to minimize the computational complexity, since as the number of hidden layers is increased, the computational complexities, such as time complexity, will also increase. The observed variations of the local maxima are because of the data nature and environmental variations of the working environments.

Table 1 shows comparison between different classifier algorithms using the reduced RSS datasets having selected BSSIDs. In the localization process, we compare SVM-RBF with KNN and ANN algorithms to select the best classifier. The SVM-RBF has the best performance for larger classes compared with KNN and ANN. The SVM classification performance is nearly 100%, which is the best classifier compared to KNN and ANN. Therefore, we combine SVM-RBF with DNN to provide the (x, y) location with the best positioning performances.

Figure 4 shows the general structure of the proposed approach. Generally, after LDA is applied to reduce features, training and testing phases are operated. The green arrow shows the phase from preprocessing up to model construction for positioning. The blue arrow indicates the testing phase of the constructed model, providing the (x, y) location as a final target. In the training phase, we project the offline datasets to the LDA-based projected vector. Then, SVM-DNN algorithms were applied using the result data from the projected vector of the LDA technology to design an appropriate model. The proposed algorithms are carried out using Python 3.6 programming language with the TensorFlow framework because it is easy for the researchers.

In the testing phase, the online RSS values are used for position estimation from an unknown location. To identify the fail APs, we consider BSSIDs of the APs used in the offline phase to identify which is not available during online positioning. As authors presented in [45], positioning in a WLAN environment may have unexpected AP failure because of power outages, WLAN system maintenance, or temporary shutdown or permanent removal of APs. These cause positioning to be terminated due to scalability problems, which causes positioning services to be terminated for a while. Therefore, we should have a means of delivering a continuous positioning service when certain APs are unable to provide Wi-Fi services. Thus, before positioning, APs are checked by BSSIDs whether there are missed APs or not. If all APs can provide a Wi-Fi signal, localization continues using the proposed model. If there are missed APs, RSSs are filled automatically by regression using FFNN. It is necessary for reliable location estimations in case of unpredicted AP failures or malicious attacks, since it injects faults and compromises the presentation of the positioning system

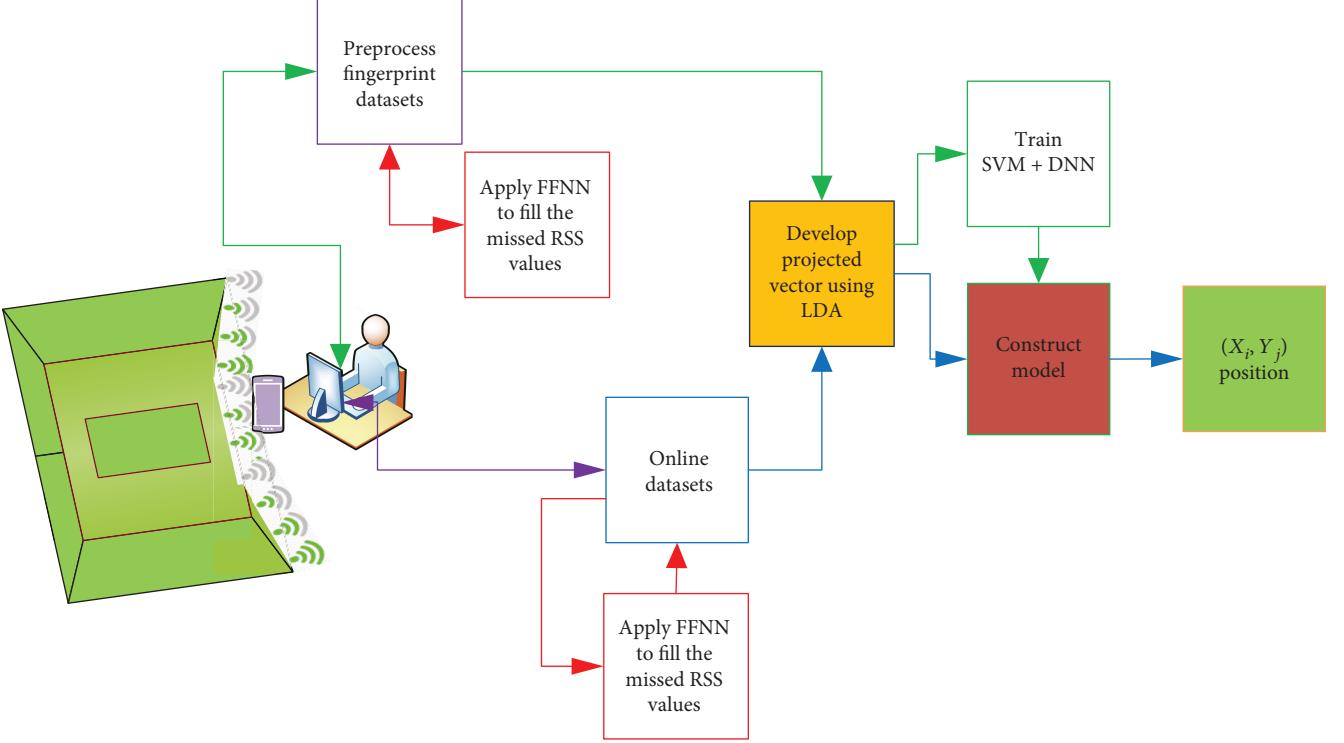


FIGURE 4: Proposed system structure.

[45]. Therefore, we use FFNN for filling the missed RSS values, since it is very fast and accurate for pattern recognitions [21]. Finally, localization is continuous through a designed model during training. Automatic filling of the missed values makes the proposed approach to provide continuous positioning services. Therefore, whenever the trained APs are unable to give Wi-Fi signal in the online stage, the system updates automatically to provide scalable positioning services. The aim of this work is to provide accurate and fault-tolerant positioning in location-data-requiring areas.

The proposed approach is evaluated mainly in terms of coordinate-based positioning of mobile users. Moreover, we use efficiency of the system performance, RMSEs, and system performance in certain ranges to evaluate the proposed approach. RMSE is computed as shown in

$$\text{RMSEs} = \sqrt{\frac{\sum_{i=1}^k (X_i^r - X_i^e)^2}{k}}, \quad (7)$$

where X_i^r is the real distance, X_i^e is the predictable distance at the i th point, and the total number of testing points is represented by k . The distance accuracy in each testing point is calculated in terms of position estimation errors, as in

$$\text{PEE}_i = \sqrt{(X_{ai} - X_{ei})^2 + (Y_{aj} - Y_{ej})^2}, \quad (8)$$

where PEE_i is the error of the proposed system at the i th location, X_{ai} is the actual distance from the X -coordinate, X_{ei} is the i th estimated distance from the X -coordinate, and Y_{aj} is

TABLE 2: Positioning accuracies of the SVM-DNN algorithm in different ranges.

Scenarios	Estimated errors in different ranges (m)				Average Errors (m)
	<0.50 m	<0.75 m	<0.90 m	<0.95 m	
Scenario 1 (%)	57.14%	78.58%	95.71%	100%	0.48
Scenario 2 (%)	54.29%	75.71%	97.14%	100%	0.49
Scenario 3 (%)	51.43%	77.14%	92.86%	98.57%	0.51
Scenario 4 (%)	50%	70%	90%	95.71%	0.53

the actual distance and Y_{ej} the estimated distance from the Y -coordinate.

5. Results and Discussions

In this section, the testing results using a hybrid of SVM and DNN in wireless environments are presented to show the performance of the proposed method. In the positioning processes, the proposed method is evaluated in different scenarios: positioning without any missed APs as scenario 1, missing the one trained AP as scenario 2, missing the two trained APs as scenario 3, and missing the three trained APs as scenario 4. The proposed method is evaluated in terms of accuracy and scalability.

5.1. Indoor Positioning Results and Discussions. In this subsection, the proposed method's positioning performances in four different scenarios are discussed. Table 2 illustrates the indoor positioning performances in scenario 1, scenario 2,

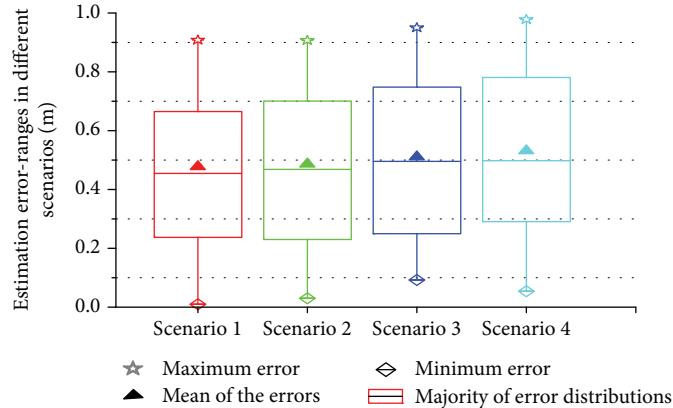


FIGURE 5: SVM-DNN positioning performance in different scenarios.

scenario 3, and scenario 4 using the SVM-DNN algorithm. A hybrid of the SVM and DNN technologies makes the positioning services continuous even when the trained APs are unable to provide a wireless signal. The performances of the proposed method are 57.14%, 54.29%, 51.43%, and 50% in scenario 1, scenario 2, scenario 3, and scenario 4 for estimation errors less than 0.50 m, respectively. The positioning errors less than 0.90 m covers 95.71%, 97.14%, 92.86%, and 90% for scenario 1, scenario 2, scenario 3, and scenario 4, respectively. All positioning errors are less than 1 m in each scenario. Majority of the errors are also less than 0.50 m in each scenario. The results indicate that indoor positioning is continuous when different trained APs are missed in the testing phase. This means that the integration of IoT and the automatic filling of missed RSS values can reduce performance degradation when APs get some troubles. To evaluate the proposed system performance, the missed APs are selected randomly from the first 43 APs. The performances of the proposed system show motivated results in each scenario. The performance is also nearly the same when one and two trained APs are missed. This indicates that the hybrid of the SVM and DNN algorithms also makes the positioning accuracy carry on robustly. This is because of the SVM kernels and the nonlinear activation function of the DNN to easily operate and understand nonlinear datasets. Additionally, the superior learning capacity of DNN and SVM makes the result to be more accurate. In each scenario, positioning accuracy has estimation errors less than 0.97 m, and the accuracies in similar ranges are nearly similar at each scenario.

The mean error difference between scenario 1 and scenario 4 is only 0.05, which is very small compared to the number of missed APs. The overall performances of the proposed system are accurate and scalable, since the algorithm makes to continuously locate the mobile users when the trained APs are missed.

Figure 5 presents the error distributions of the proposed system in four scenarios. It shows that the proposed system performances in terms of the minimum errors, mean of errors, maximum errors, and estimation error distributions in each scenario. The minimum error, mean of errors, and maximum error are 0.01 m, 0.48 m, and 0.90 m in scenario

1, respectively. In scenario 2, the minimum error is 0.03 m, the mean of errors is 0.48 m, and the maximum error is 0.90 m. Scenario 3 has 0.09 m, 0.51 m, and 0.95 m for the minimum, mean, and maximum errors, respectively. The minimum error, mean of errors, and maximum error in scenario 4 are 0.05 m, 0.53 m, and 0.97 m, respectively. The means of errors in scenario 1 and scenario 2 are similar. The majority of errors ranged from 0.23 m to 0.67 m in scenario 1, and it ranged from 0.22 m to 0.73 m in scenario 2. Hence, the proposed system performs with only a few shiftings in scenario 2 than in scenario 1. In general, the majority of error distributions in each scenario are the same. The automatic filling of the missed values using FFNN inspires positioning performances when APs are unable to give a Wi-Fi signal.

The result also indicates that the mean errors of the proposed algorithm in each scenario have no big differences due to the scalability of the proposed system. The majority of positioning error distributions are less than 0.67 m, 0.73 m, 0.74 m, and 0.79 m for scenario 1, scenario 2, scenario 3, and scenario 4, respectively. This indicates that the proposed system is fault-tolerant and adapts to signal fluctuations. The errors are changed slowly in each scenario, since the ranges of estimated errors have no big differences.

The RMSEs of the proposed system have no big differences in various scenarios, since our hybrid approach could learn deeply the signal fluctuations and can resist AP faults easily. The RMSEs of the proposed method are 0.062, 0.063, 0.068, and 0.069 in meters at scenario 1, scenario 2, scenario 3, and scenario 4, respectively. This indicates that the proposed approach provides scalable and accurate positioning in each scenario.

5.2. Outdoor Positioning Results and Discussions. In this subsection, the proposed system performances in the outdoor environment are presented. We used similar scenarios to indoor positioning. The positioning is done in an open space, where the area is divided into 1 m^2 grids. Table 3 illustrates the positioning performances in different ranges of the outdoor environment. In each scenario, positioning accuracy has estimation errors less than 1.90 m. The maximum errors in scenario 1, scenario 2, and scenario 3 are 1.43 m, 1.53 m,

TABLE 3: Positioning accuracies of the SVM-DNN algorithm in different ranges.

Accuracy at different scenarios	Error boundaries (m)	Average errors (m)	RMSE
	<0.9 m	<1 m	<1.5 m
Scenario 1 (%)	62.5%	62.5%	100%
Scenario 2 (%)	50%	62.5%	87.5%
Scenario 3 (%)	37.5%	50%	87.5%
Scenario 4 (%)	50%	50%	75%
			1.05
			0.42

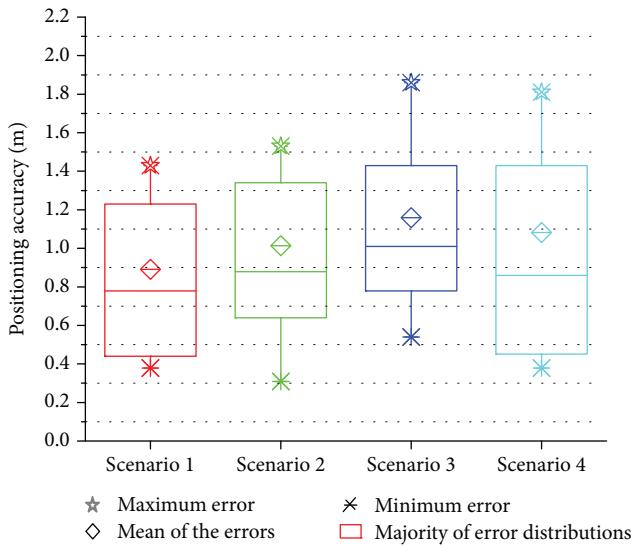


FIGURE 6: Outdoor positioning accuracies.

1.54 m, and 1.89 m, respectively. The result shows that the average errors have no big differences, since the proposed approach helps the machine to easily learn the signal fluctuations. The RMSEs also show the proposed approach providing motivating results. The observed signal from the outdoor environment is much fluctuated; however, the SVM-DNN learns easily to deliver the motivated results due to its deeper learning capacity. In each scenario, more than 50% of the estimation accuracy has less than 1 m positioning errors, and all testing errors in each scenario are less than 2 m. The mean errors in all scenarios are nearly similar. The RMSEs in different scenarios also give motivating results.

Figure 6 illustrates the positioning performances in terms of minimum error, maximum error, mean of errors, and boundary of positioning errors in each scenario. The majority of estimated errors are distributed in similar ranges, which implies that the proposed approach provides stable positioning services in the highly fluctuated Wi-Fi environments. The minimum errors are 0.38 m, 0.31 m, 0.48 m, and 0.38 m in scenario 1, scenario 2, scenario 3, and scenario 4, respectively. The maximum errors are 1.43 m, 1.53 m, 1.54 m, and 1.89 m in scenario 1, scenario 2, scenario 3, and scenario 4, respectively. The maximum errors did not always occur when more than one AP is missing.

Although there are few estimation variations in different testing points, the RMSE, the average of error values, and the

majority of estimation distributions do not show big gaps in different scenarios. The results indicate that the positioning is continuous even when the trained APs are missed. The main reasons behind the accurate performances of the proposed system are an automatic updating of the missed APs through regression and the learning capability of the proposed system. Moreover, the application of LDA for feature reduction and the appended BSSIDs having the five strongest RSS values help the proposed algorithms achieve the best positioning accuracies. The application of LDA also enables to remove noise and other irrelevant information and makes the complex datasets simpler for the operation and improve accuracies. Additionally, collecting numbers of signal values from a 1 m-by-1 m grid helps the proposed system to adapt to the environment easily for accurate positioning.

The multi-kernel of SVM and backpropagation of DNN help to provide accurate positioning, since SVM can transpose larger datasets using its multi-kernels. Furthermore, DNN can easily update itself according to data natures and can control overfitting problems. It has also good positioning performance, since it is more suitable for pattern recognition problems. The SVM can distinguish data that is not linearly separable, and then DNN can easily locate the specific position of smartphone users. DNN is fully connected so that each node in one layer connects with the following nodes with a certain weight w_{ij} , which allows DNN to learn superiorly.

The results in different ranges are motivated to apply the proposed approach for positioning services in various demanding areas. Principally, the integration of SVM and DNN makes positioning more accurate since positioning based on correctly classified values makes the system more robust [26]. DNN is more flexible with respect to accurate positioning in complex environments, and SVM helps in easily classifying hierarchical data distribution due to its multiple kernels. Although the computational time during the training stage is very high because of larger datasets and sets of integrated algorithms, the testing stage needs very small computational time and is relatively straightforward. The average computational time complexity of the proposed method using reduced datasets is 0.20 s. This means the testing time complexity is not affected due to missing the trained APs or other troubles, since the missed APs will be filled through FFNN before positioning. Our previous work [26] showed that LDA could possibly improve the computational time complexity. However, the scalability issue was not addressed.

Generally, the results show that the proposed technique can provide accurate and scalable indoor and outdoor positioning in complex, hierarchical, and dynamic environments. The results also indicate that the errors changed very calmly between consecutive testing points in both indoor and outdoor environments due to the superior robustness of SVM-DNN on uncertain, as well as complex, situations.

6. Conclusions

In this paper, the integrations of FFNN, LDA, SVM, and DNN algorithms are applied for indoor and outdoor

positioning. FFNN is used to develop a scalable system. LDA is applied to reduce the complex dimension of the scanned RSS values to lower features without affecting information contents. We use a hybrid of SVM-DNN to locate the target of smartphone users. We evaluate the proposed method in different scenarios at indoor and outdoor positioning schemes. The proposed approach provides 100% of the estimation accuracy with errors less than 0.97 m and 1.89 m during indoor and outdoor positioning, respectively. Additionally, our method implements with fewer average errors and variances. The error distributions of the majority of the results are similarly ranged in the same boundary, and error changes between consecutive testing points are slow. The computational time complexities are also small in each scenario in both indoor and outdoor positioning schemes. Thus, the integration of IoT gives the state-of-the-art performance on indoor and outdoor positioning in hierarchical, dynamic, and complex environments. The application of DNN for indoor and outdoor localization using an unmanned aerial vehicle (UAV) in urban and suburban areas is our potential future work.

Data Availability

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

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

This research is funded by the Ministry of Science and Technology (MoST) in Taiwan (Grant No.: 107-2634-F-009-006).

References

- [1] J. Liu, *Survey of Wireless Based Indoor Localization Technologies*, Department of Science & Engineering, Washington University, 2014.
- [2] O. Chakroun, M. Hild, and B. Abdulrazak, *Indoor and Outdoor Localization Framework for Pervasive Environments*, Ambient Systems, Networks and Technologies (ANT), 2010.
- [3] M. V. Moreno-Cano, M. A. Zamora-Izquierdo, J. Santa, and A. F. Skarmeta, “An indoor localization system based on artificial neural networks and particle filters applied to intelligent buildings,” *Neurocomputing*, vol. 122, pp. 116–125, 2013.
- [4] D. Macagnano, G. Destino, and G. Abreu, “Indoor positioning: a key enabling technology for IoT applications,” in *2014 IEEE World Forum on Internet of Things (WF-IoT)*, pp. 117–118, Seoul, South Korea, 2014.
- [5] K. Lin, M. Chen, J. Deng, M. M. Hassan, and G. Fortino, “Enhanced fingerprinting and trajectory prediction for IoT localization in smart buildings,” *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 3, pp. 1294–1307, 2016.
- [6] W. H. Kuo, Y. S. Chen, K. T. Cheng, and T. W. Lu, “Signal strength based indoor and outdoor localization scheme in zigbee sensor networks,” *IAENG International Journal of Computer Science*, vol. 23, no. 1, pp. 55–64, 2016.
- [7] A. Yassin, Y. Nasser, M. Awad et al., “Recent advances in indoor localization: a survey on theoretical approaches and applications,” *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 1327–1346, 2017.
- [8] A. A. Khudhair, S. Q. Jabbar, M. Qasim Sulttan, and D. Wang, “Wireless indoor localization systems and techniques: survey and comparative study,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 3, no. 2, p. 392, 2016.
- [9] Y. K. Cheng, H. J. Chou, and R. Y. Chang, “Machine-learning indoor localization with access point selection and signal strength reconstruction,” in *2006 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, pp. 1–5, Nanjing, China, 2016.
- [10] A. Paul and T. Sato, “Localization in wireless sensor networks: a survey on algorithms, measurement techniques, applications and challenges,” *Journal of Sensor and Actuator Networks*, vol. 6, no. 4, p. 24, 2017.
- [11] A. Paul and T. Sato, “Detour path angular information based range-free localization in wireless sensor network,” *Journal of Sensor and Actuator Networks*, vol. 2, no. 1, pp. 25–45, 2013.
- [12] K. S. Kim, R. Wang, Z. Zhong et al., “Large-scale location-aware services in access: hierarchical building/floor classification and location estimation using Wi-Fi fingerprinting based on deep neural networks,” *Fiber and Integrated Optics*, vol. 37, no. 5, pp. 277–289, 2018.
- [13] S. Petr, S. Martin, and K. Dominik, “An overview of indoor and outdoor positioning technologies with focus on their precision,” *Electrorevue*, vol. 18, no. 6, pp. 183–192, 2016.
- [14] T. Halonen, J. Romero, and J. Melero, Eds., *GSM, GPRS, and EDGE Performance: Evolution towards 3G/UMTS*, John Wiley & Sons, 2004.
- [15] S. Treuillet and E. Royer, “Outdoor/indoor vision-based localization for blind pedestrian navigation assistance,” *International Journal of Image and Graphics*, vol. 10, no. 4, pp. 481–496, 2010.
- [16] S. Xia, Y. Liu, G. Yuan, M. Zhu, and Z. Wang, “Indoor fingerprint positioning based on Wi-Fi: an overview,” *ISPRS International Journal of Geo-Information*, vol. 6, no. 5, p. 135, 2017.
- [17] M. Nowicki and J. Wietrzykowski, “Low-effort place recognition with WiFi fingerprints using deep learning,” in *Automation 2017. ICA 2017*, pp. 575–584, Springer, Cham, Switzerland, 2017.
- [18] C. S. Chen, “Artificial neural network for location estimation in wireless communication systems,” *Sensors*, vol. 12, no. 3, pp. 2798–2817, 2012.
- [19] M. Farhat, M. Ibrahim, and I. Nesreen, “Effects of walls and floors in indoor localization using tracking algorithm,” *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 3, 2016.
- [20] S. Ezpeleta, J. Claver, J. Pérez-Solano, and J. Martí, “Rf-based location using interpolation functions to reduce fingerprint mapping,” *Sensors*, vol. 15, no. 10, pp. 27322–27340, 2015.
- [21] A. Belay, L. Yen, S. Renu, H. P. Lin, and S. S. Jeng, “Indoor localization at 5GHz using dynamic machine learning approach (DMLA),” in *International Conference on Applied System Innovation (ICASI)*, pp. 1763–1766, Sapporo, Japan, 2017.
- [22] L. Yen, C.-H. Yan, S. Renu, A. Belay, H.-P. Lin, and Y.-S. Ye, “A modified WKNN indoor Wi-Fi localization method with

- differential coordinates,” in *2017 International Conference on Applied System Innovation (ICASI)*, pp. 1822–1824, Sapporo, Japan, 2017.
- [23] A. B. Adege, Y. Yayeh, G. Berie, H.-p. Lin, L. Yen, and Y. R. Li, “Indoor localization using K-nearest neighbor and artificial neural network back propagation algorithms,” in *2018 27th Wireless and Optical Communication Conference (WOCC)*, pp. 1-2, Hualien, Taiwan, 2018.
 - [24] Y. Liu and Q. Chen, “Research on integration of indoor and outdoor positioning in professional athletic training,” *Proceedings*, vol. 2, no. 6, p. 295, 2018.
 - [25] L. Reyero and G. Delisle, “A pervasive indoor-outdoor positioning system,” *Journal of Networks*, vol. 3, no. 8, 2008.
 - [26] A. Adege, H.-P. Lin, G. Tarekgn, and S.-S. Jeng, “Applying deep neural network (DNN) for robust indoor localization in multi-building environment,” *Applied Sciences*, vol. 8, no. 7, p. 1062, 2018.
 - [27] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*, MIT Press, Cambridge, MA, USA, 1st edition, 2016.
 - [28] G. Félix, M. Siller, and E. N. Alvarez, “A fingerprinting indoor localization algorithm based deep learning,” in *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 1006–1011, Vienna, Austria, 2016.
 - [29] W. Zhang, K. Liu, W. Zhang, Y. Zhang, and J. Gu, “Deep neural networks for wireless localization in indoor and outdoor environments,” *Neurocomputing*, vol. 194, pp. 279–287, 2016.
 - [30] X. Wang, L. Gao, S. Mao, and S. Pandey, “DeepFi: deep learning for indoor fingerprinting using channel state information,” in *2015 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1666–1671, New Orleans, LA, USA, 2015.
 - [31] X. Wang, L. Gao, S. Mao, and S. Pandey, “CSI-based fingerprinting for indoor localization: a deep learning approach,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763–776, 2017.
 - [32] K. S. Kim, S. Lee, and K. Huang, “A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting,” *Big Data Analytics*, vol. 3, no. 1, 2018.
 - [33] A. Moreira, M. J. Nicolau, F. Meneses, and A. Costa, “Wi-Fi fingerprinting in the real world - RTLS@UM at the EvAAL competition,” in *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–10, Banff, AB, Canada, 2015.
 - [34] D. L. Tapia, J. Bajo, J. F. De Paz, R. S. Alonso, S. Rodríguez, and J. M. Corchado, “Using multi-layer perceptrons to enhance the performance of indoor rtls,” in *Proceedings of the Progress in Artificial Intelligence in 15th Portuguese Conference on Artificial Intelligence: EPIA*, pp. 31–39, Lisbon, Portugal, 2011.
 - [35] U. Ahmad, A. Gavrilov, S. Lee, and Y. K. Lee, “Modular multilayer perceptron for WLAN based localization,” in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, pp. 3465–3471, Vancouver, BC, Canada, 2006.
 - [36] M. Dakkak, B. Daachi, A. Nakib, and P. Siarry, “Multi-layer perceptron neural network and nearest neighbor approaches for indoor localization,” in *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1366–1373, San Diego, CA, USA, 2014.
 - [37] K. Lingemann, H. Surmann, A. Nuchter, and J. Hertzberg, “Indoor and outdoor localization for fast mobile robots,” in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, pp. 2185–2190, Sendai, Japan, 2004.
 - [38] Y. Tifani, B. Lee, and E. Jeong, “A patient’s indoor positioning algorithm using artificial neural network and SVM,” *Journal of Theoretical & Applied Information Technology*, vol. 95, no. 16, 2017.
 - [39] G. Deak, K. Curran, J. Condell, D. Deak, and P. Kiedrowski, “Support vector machine and probability neural networks in a device-free passive localization (DFPL) scenario,” *Image Processing & Communications*, vol. 17, no. 4, pp. 9–16, 2012.
 - [40] H. Mehmood, N. K. Tripathi, and T. Tipdecho, “Indoor positioning system using artificial neural network,” *Journal of Computer Science*, vol. 6, no. 10, pp. 1219–1225, 2010.
 - [41] P. Lopez-de-Teruel, O. Canovas, and F. J. Garcia, “Using dimensionality reduction techniques for refining passive indoor positioning systems based on radio fingerprinting,” *Sensors*, vol. 17, no. 4, 2017.
 - [42] S. Jeon, J. P. Jeong, Y. J. Suh, C. Yu, and D. Han, “Selective AP probing for indoor positioning in a large and AP-dense environment,” *Journal of Network and Computer Applications*, vol. 99, pp. 47–57, 2017.
 - [43] Z. A. Deng, Y. Xu, and L. Chen, “Localized local fisher discriminant analysis for indoor positioning in wireless local area network,” in *2013 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 4795–4799, Shanghai, China, 2013.
 - [44] E. Alpaydin, *Introduction to Machine Learning*, The MIT Press, Cambridge, Massachusetts, London, England, 2nd edition, 2014.
 - [45] C. Laoudias, M. P. Michaelides, and C. G. Panayiotou, “Fault detection and mitigation in WLAN RSS fingerprint-based positioning,” *Journal of Location Based Services*, vol. 6, no. 2, pp. 101–116, 2012.

