

iBeacon-based indoor positioning system: from theory to practical deployment

Thai-Mai Dinh Thi, Ngoc-Son Duong

Faculty of Electronics and Telecommunications, VNU University of Engineering and Technology, Vietnam National University, Hanoi, Vietnam

Article Info

Article history:

Received Sep 9, 2022

Revised Jan 11, 2023

Accepted Feb 4, 2023

Keywords:

Bluetooth low energy iBeacon

Fingerprinting

Indoor localization

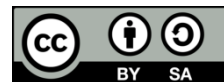
Indoor positioning system

K-nearest-neighbor

ABSTRACT

Developing an indoor positioning system became essential when global positioning system signals could not work well in indoor environments. Mobile positioning can be accomplished via many radio frequency technology such as Bluetooth low energy (BLE), wireless fidelity (Wi-Fi), ultra-wideband (UWB), and so on. With the pressing need for indoor positioning systems, we, in this work, present a deployment scheme for smartphone using Bluetooth iBeacons. Three main parts, hardware deployment, software deployment, and positioning accuracy assessment, are discussed carefully to find the optimal solution for a complete indoor positioning system. Our application and experimental results show that proposed solution is feasible and indoor positioning system is completely attainable.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Thai-Mai Dinh Thi

Faculty of Electronics and Telecommunications, VNU University of Engineering and Technology

G2 Building, 144 Xuan Thuy Str., Cau Giay Dist., Hanoi, Vietnam

Email: dtmai@vnu.edu.vn

1. INTRODUCTION

In the forthcoming decade, indoor positioning will be an essential part of smart cities, where users and service providers can gain many advantages from the position information. It will be applied in various aspects such as robot navigation, indoor location tracking, emergency rescuing, business intelligence and analytics, experience improvement, health care, and other indoor location-based services. There are many technology were proposed to turn indoor positioning into reality such as wireless fidelity (Wi-Fi) [1], ultra-wide band (UWB) [2], radio frequency identification (RFID) [3], and so on. However, due to their high cost or less reliability, such technologies seem unsuitable for indoor localization. Therefore, Bluetooth low energy (BLE) have emerged as a new candidate to compensate for the old ones. With the release of the Bluetooth 4.0 standard in 2009, Bluetooth-based indoor positioning system (IPS) gained popularity. Since then, the standard's requirements have made it possible to use the technology for localization in more sophisticated ways by establishing a brand-new class of gadgets called "Bluetooth beacons". The development of iBeacon is an example. The new devices running on iBeacon protocol have the possibility to broadcast at predetermined intervals, which improves the hardware and the immunity to interference while greatly increasing the energy efficiency of the system compared to those that used the previous standards. Spachos and Plataniotis [4], the group of authors introduces an indoor localization system that helps to enhance the user experience in a museum. This study focuses on estimating the distance from the exhibits to the user's location, collecting useful analytics regarding each visit, and providing a recommendation to the users. The experiment results show that the BLE beacon is a promising solution for a smart interactive museum. Liu *et al.* [5] introduce a smart home control that includes five parts: web server, home gateway, smart terminal, smartphone app, and BLE beacons. The results show that the proposed system provides a good solution

with appropriate accuracy, execution time, extensibility, and endurance. Chan *et al.* [6], for the first time, use iBeacon to support non-verbal students in special education. iBeacon in this study is utilized to display contextually related picture cards for the classroom via a unique identifier of the nearest detected beacon. Molina *et al.* [7] use Wi-Fi and BLE beacon for positioning in the airport. Khruahong *et al.* [8] posed a situation of needing to find lost property in an indoor environment using iBeacon. Chen *et al.* [9] designed a tracking and monitoring system for the child. The designed system used mobile iBeacon nodes for crowdsourced finding and static iBeacon nodes for indoor positioning. Chiu *et al.* [10] iBeacons were used to aid Wi-Fi indoor positioning. In this hybrid setup, Wi-Fi access points may create distinct large zones, and iBeacons can precisely pinpoint the user's location within an enclosed space. The findings demonstrate that the hybrid system developed may reach a certain degree of accuracy and may be a viable localization option for a mobile application. Lin *et al.* [11] introduce a iBeacon-based indoor positioning solution for patient. Their deployment strategy can be divided into three phases: preparatory, measure, and location identification. The first phase involves iBeacon calibration and device deployment. Meanwhile, the second and third phases involve collecting data and then indicating the user's location using a previous database. In addition, some systems that use BLE beacons include [12], [13]. Thus, it can be seen that iBeacon has been used in many studies before. However, there is no research on how to build a practical iBeacon-based IPS. Ye *et al.* [14] is one of the few studies that identify the influencing factors when building an iBeacon-based IPS. However, this study is still very rudimentary. Hence, this work will provide a comprehensive insight into this problem.

The rest of this paper is organized as follows. In section 2, we briefly introduce iBeacon protocol and methods of positioning using received signal strength (RSS) from iBeacon. We present a comprehensive guideline for IPS providers and developers who want to deploy a practical IPS in section 3, including hardware deployment in subsection 3.1 and software deployment in subsection 3.2. Section 4 introduces an indoor application and evaluates the accuracy of the proposed system. Ultimately, section 5 concludes this work.

2. INDOOR POSITIONING WITH iBeacon AND RSS MEASUREMENT

2.1. iBeacon protocol

Apple unveiled the Bluetooth communication protocol known as iBeacon in 2013. When a device emitting iBeacon is nearby (commonly those devices are called beacons), the beacon allows smartphones, tablets, and other devices to know its (relative) location. As its name implies, an iBeacon device acts as a lighthouse. Instead of emitting light to provide navigation for ships, iBeacon devices broadcasts BLE signals to let smartphones know their location. Its signal contains four pieces of information which are:

- Unique universal identifier (UUID): general information, refers to the name of organization that is using it.
- Major: segment of UUID, refers to the beacon's general location. For example, this beacon is located in the building #1.
- Minor: smaller subsegment, refers to beacon's detailed location. For example, this beacon is located in floor 5.
- Tx-Power: refers to RSS.

2.2. Indoor positioning with RSS measurement from BLE iBeacon

By the method employed for localization, systems that use RSS from BLE iBeacons can be divided into two groups: i) techniques based on distance and ii) techniques based on fingerprints (FP). In distance-based techniques, one should gain distance information from the well-known propagation model (1),

$$\text{RSS}(d_i) = \text{RSS}(d_0) - 10\eta \log\left(\frac{d_i}{d_0}\right) + \sigma \quad (1)$$

where $\text{RSS}(d_0)$ and $\text{RSS}(d_i)$ are the received signal strength at the reference distance $d_0 = 1$ m and d_i , respectively. η is path loss exponent and σ is Gaussian noise with zero mean. The distance between target and the i -th iBeacon node can be expressed as (2).

$$d_i = d_0 10^{\frac{\text{RSS}(d_0) - \text{RSS}(d_i)}{10\eta}} \quad (2)$$

After having distance from at least 3 beacons, we can estimate the target's position through lateral-based or least square method. This approach has the benefit of being straightforward and requiring less effort for site surveying and maintaining. However, it offers a low level of precision. Meanwhile, fingerprinting approach is a two-part method that consists of an offline phase and an online phase. Site surveying, which involves sampling the region of interest's fingerprints at predetermined reference points (RPs), is carried out during the offline phase. Smartphone sensors normally are used to obtain fingerprints. A database is used to

store the sampled fingerprints and the coordinates that go with them. After that, the data is utilized to train by machine learning algorithm to discover the optimal function for mapping the sampled fingerprints to their accurate coordinates. Following that, during the online phase, the learned function is utilized to infer a user's coordinates from the measured fingerprints at the user's position. This method is widely used because its stability and high precision. However, its drawback is time-consuming.

3. PROPOSED SYSTEM IMPLEMENTATION OF iBeacon AS INDOOR POSITIONING SYSTEM

3.1. Hardware deployment

3.1.1. Deployment of beacon ID

We can understand this relationship by (3).

$$UUID \subset Major \subseteq Minor \quad (3)$$

In practical applications, the signal needs to be read and converted into relevant information by smartphone applications. Hierarchically, the UUID value can be used to differentiate an organization's iBeacon from others, the major value is used for specification of any iBeacon within a group, and the minor value is for identifying a particular iBeacon [15]. An iBeacon implementation example is provided by Apple in [16]. In this article, the author uses the UUID to differentiate the product of a company from others, this parameter is shared by all locations. Major is used for distinguishing products in a specific area from others. And minor is used to differentiate the product type in the exact location. Considering a large-scale case, i.e., many companies and organizations/corporations want to implement an indoor positioning system using iBeacon, the deployment in terms of identification value should be: i) use the UUID to differentiate buildings managed by an organization/corporation, ii) use the major number to distinguish floors in a building, and iii) using minor numbers to distinguish rooms/areas in a floor.

For resource-saving and efficient management, the major assigned to the floors from the building to the building should be the same, while the minor is optional. Moreover, the homogeneous selection of the major number allows positioning both in 2-dimensional space and 3-dimensional space by mapping from major to the floors' numerical order. Suppose we need to deploy several iBeacon for two buildings of company A, namely CT1 and CT2. CT1 and CT2 consist of 2 and 3 floors, respectively. Each floor in a building need three iBeacons. Then, the implementation strategy is given by Table 1.

Table 1. Deployment of UUID, major and minor of iBeacon on 2 buildings

Level (Floor)	Beacon number	1st building (CT1)	2nd building (CT2)
1	1	UUID = ffffffff-1234-aaaa-a1b2c3d4e5f6 Major = 10; Minor = 10	UUID = ffffffff-5678-aaaa-a1b2c3d4e5f6 Major = 10; Minor = 10
1	2	UUID = ffffffff-1234-aaaa-a1b2c3d4e5f6 Major = 10; Minor = 20	UUID = ffffffff-5678-aaaa-a1b2c3d4e5f6 Major = 10; Minor = 20
1	3	UUID = ffffffff-1234-aaaa-a1b2c3d4e5f6 Major = 10; Minor = 30	UUID = ffffffff-5678-aaaa-a1b2c3d4e5f6 Major = 10; Minor = 30
2	4	UUID = ffffffff-1234-aaaa-a1b2c3d4e5f6 Major = 20; Minor = 10	UUID = ffffffff-5678-aaaa-a1b2c3d4e5f6 Major = 20; Minor = 10
2	5	UUID = ffffffff-1234-aaaa-a1b2c3d4e5f6 Major = 20; Minor = 20	UUID = ffffffff-5678-aaaa-a1b2c3d4e5f6 Major = 20; Minor = 20
2	6	UUID = ffffffff-1234-aaaa-a1b2c3d4e5f6 Major = 20; Minor = 30	UUID = ffffffff-5678-aaaa-a1b2c3d4e5f6 Major = 20; Minor = 30
3	7	None	UUID = ffffffff-5678-aaaa-a1b2c3d4e5f6 Major = 30; Minor = 10
3	8	None	UUID = ffffffff-5678-aaaa-a1b2c3d4e5f6 Major = 30; Minor = 20
3	9	None	UUID = ffffffff-5678-aaaa-a1b2c3d4e5f6 Major = 30; Minor = 30

3.1.2. iBeacon hardware placement

The study in [17] introduces a novel iBeacon placement strategy called crystal placement (CiP). By placing the iBeacons into an equilateral triangle, CiP allows the mobile device to hear the maximum number of beacons in the environment, thereby increasing positioning accuracy. We would agree with them that desired height should be greater than 175 cm. But we would argue that CiP is only optimal for the weighted centroid localization algorithm and not for others. Another research [18] also recommended placing beacons in the shape of equilateral triangles with an optimal distance between two adjacent beacons is $3\sqrt{3}$ meter instead of

50 m as [17] suggest. This strategy ensures the number of beacons that need to be deployed is the smallest and also provides the highest accuracy for the fusion of multilateration and pedestrian dead reckoning (PDR). In general, geometry-based methods recommend placing beacons in the form of an equilateral triangle [18], [19]. With FP, it is not necessary to set beacons to equilateral triangles. An environment-dependent beacon placement strategy would be better than CiP. Basically, indoor environments are divided into two categories: corridors and open spaces. Among them, for perceived objects such as humans, the effort to locate and navigate the object in the aisles is less than in the open space. Therefore, the number of beacons needed to deploy and the positioning algorithm for these two areas may be different for optimizing the system's cost.

3.2. Software deployment

3.2.1. Ontology and functionality

In this section, we will cover the implementation of the application on both Android and iOS platforms. Since we want to build an application with convenience like Google Maps, we use Google Map Software Development Kit (SDK) as a reference for both Android [20] and iOS [21] operating systems. The Google Map SDK provides us with many resources for building navigation applications. Any positioning system, regardless of the environment being applied: Outdoor or Indoor, should contain two main functions: positioning and navigation. In which positioning task involve finding and showing on display the current user's location. Meanwhile, the navigation part engages the task of finding the shortest path from the current location to a destination being approached. Applications on Android or iOS should work with modules given by Figure 1.

The ontology should include tables as listed in Figure 2. As mentioned above, buildings have a primary ID which is the UUID of each iBeacon. In a building system with many floors, each floor is assigned an ID that is the Major number of iBeacon. Similarly, rooms on a floor will be assigned with the minor number, which is a subset of the Major. With the analysis of the iBeacon packet on the user's device, it is possible to determine the user's location at the corresponding building, floor, and room. Next, we need a dependence table to specify the relationship between nodes (or reference point or location where offline RSSs are collected). The start-id, and target-id fields are the start/current and destination locations in the algorithm, respectively. The distance field is the non-negative physical distance between nodes. The location table contains the connected nodes and FP data.

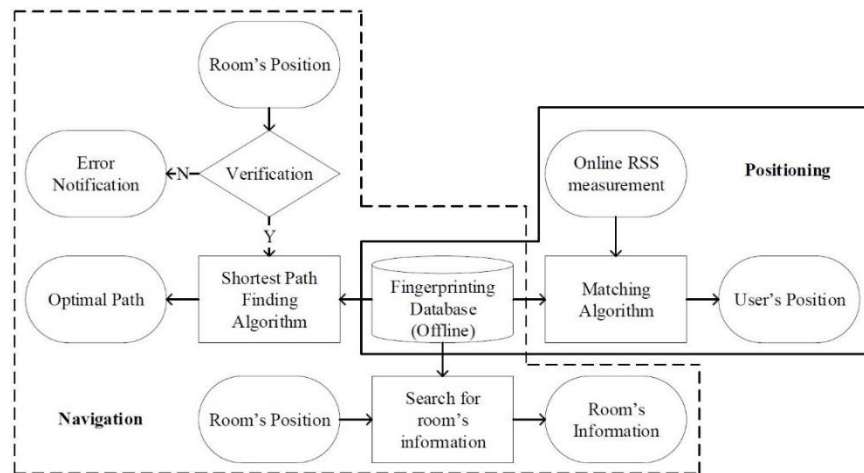


Figure 1. Indoor space ontology architecture

3.2.2. Positioning

There are two main options for positioning algorithm: i) range-based algorithm like trilateration or least square and ii) range-free algorithm like FP. It seems fair to say that FP is the method that has many advantages. Firstly, FP deals with the heterogeneity of indoor environments better than range-based algorithms. Secondly, FP works well when the position of the beacons is linear. The main idea of FP is to use big data to describe physical positions using RSSI in the interested indoor area. The fingerprint's first step is to build a so-called a radio map, which collects RSS data for all RP and stores it in a database. The user's position, in the second step, is then estimated by comparing on-the-fly RSS and pre-collected radio map using a matching algorithm. Figure 3 shows a visual representation of how fingerprinting works.

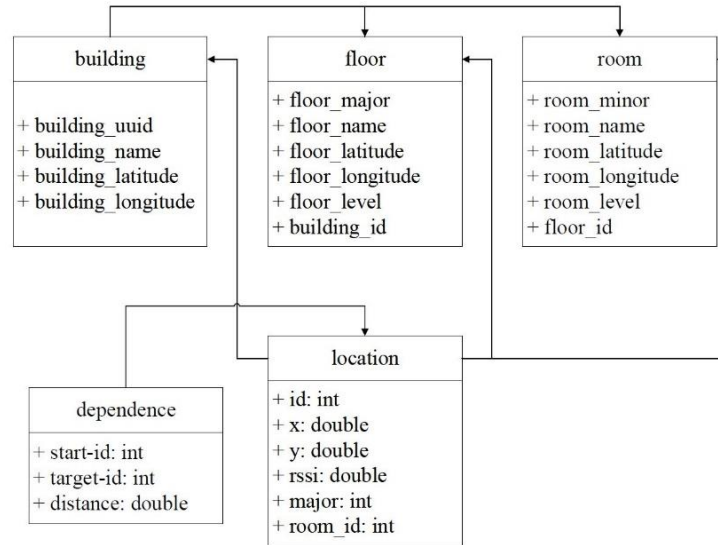


Figure 2. Indoor space ontology architecture

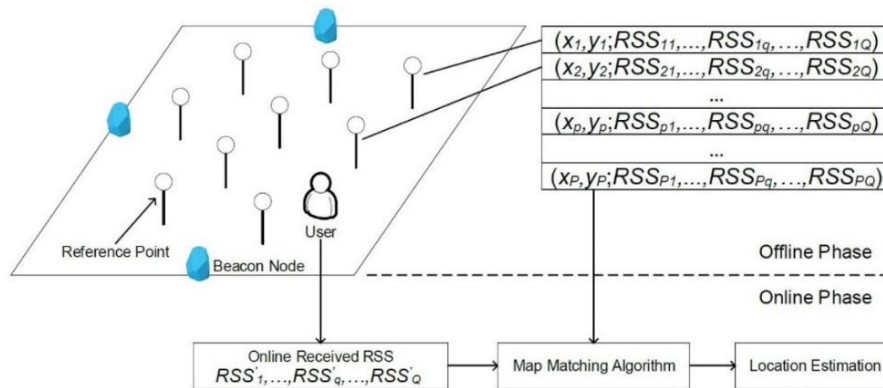


Figure 3. Fingerprinting procedure

Matching algorithms may be deterministic or probabilistic. However, deterministic is more widely used. In deterministic form, one or several RP are found that best match with recorded fingerprints. The matching mathematically tries to solve the (4).

$$\hat{x} = \underset{x_j}{\operatorname{argmin}} \sum_{i=1}^n (r_i - \rho_i(x_j))^2 \tag{4}$$

in which, r_i is RSS form i^{th} beacon in online positioning phase and $\rho_i(x_j)$ are RSS of i^{th} beacon at j^{th} RP. Algorithms of this type include K-nearest neighbors (KNN), weighted KNN (WKNN). Although we have better methods like neural network (NN) [22] or support vector machine (SVM) [22] but when compared to these methods KNN provide adequate positioning accuracy with lower computational cost [23]. Hence, KNN is chosen as matching algorithm in this study. Assume that an interested area is deployed with M beacons and can be divided into L locations. Each grid point has a physical coordinate (x_1, y_1) and includes N fingerprint points. Offline fingerprint vector can be expressed by (5).

$$\{(x_l, y_l) | \text{RSS}_{1n}^{(l)}, \dots, \text{RSS}_{mn}^{(l)}, \dots, \text{RSS}_{Mn}^{(l)}\}, \quad m \in \{1 \rightarrow M\}, n \in \{1 \rightarrow N\}, l \in \{1 \rightarrow L\} \tag{5}$$

In the online phase, when on-the-fly RSS measurement of an unknown position is available, the KNN will compare it with the pre-collected database to predict the user's location. The on-the-fly RSS vector has the form (6).

$$\{RSS'_{1n}, \dots, RSS'_{mn}, \dots, RSS'_{Mn}\}, \quad n \in [1, N] \quad (6)$$

However, KNN only use Euclidean distance that cannot accurately reflect the difference of signal propagation. As we can show in [18] that the larger the RSS value the smaller the uncertainty, so we add a weight equals to invert of current RSS to have a better estimation.

3.2.3. Coordinating

One tricky problem when using Google Map SDK is that this kit only supports programmers developing outdoor navigation applications. When applying for indoor positioning, we first have to correctly attach the indoor floor's plan to a geographic coordinate to be positioned. Then, the floor's plan must be rotated by an angle equal to the building's orientation. Finally, a step is taken to convert the Cartesian (local) coordinate system with the optional origin to the geographic coordinate system, i.e., latitude and longitude. For example, in our case, latitude, γ , is calculated from (x, y) in the Cartesian system (7),

$$\gamma = \alpha \times \beta \times (x \cos(\theta) - y \sin(\theta)) \quad (7)$$

where α , β and θ stand for the origin of the indoor floor's plan latitude, the scale depends on building's size, and the building's orientation, respectively.

3.2.4. Navigation

In order to add a path-finding function to the system, we use a graph-based approach. In this approach, we consider the indoor area's map as a graph. The graph contains a number of reference nodes, which are numbered from 1 to 16, as described in Figure 4(a). Each node represents a location on the map. Nodes are connected to each other, depending on whether there is an obstacle between two nodes. Each node has the following properties in the location table in Figure 2. The link (connection) between two nodes is also assigned a cost, which is used to compute the best path from one node to another. This cost is the Euclidean distance of two connected nodes. Then, having a graph of nodes, depend on type of operation system, we use GKGraph [24] (A-star-based) in GameplayKit framework for iOS or GraphBuilder [25] (Dijkstra-based) class in Hipster library for Android. The framework/library provides a function to find the path with the lowest cost, given a starting node (the user's current location) and an ending node (the destination). For example, assume that we have 3 points A, B, C with $AB=3$, $AC=4$, $BC=5$, respectively, we use the GraphBuilder class (Android) or GKGraph class (iOS) to connect the vertices and edges. The code snippet is described in Figure 4(b).

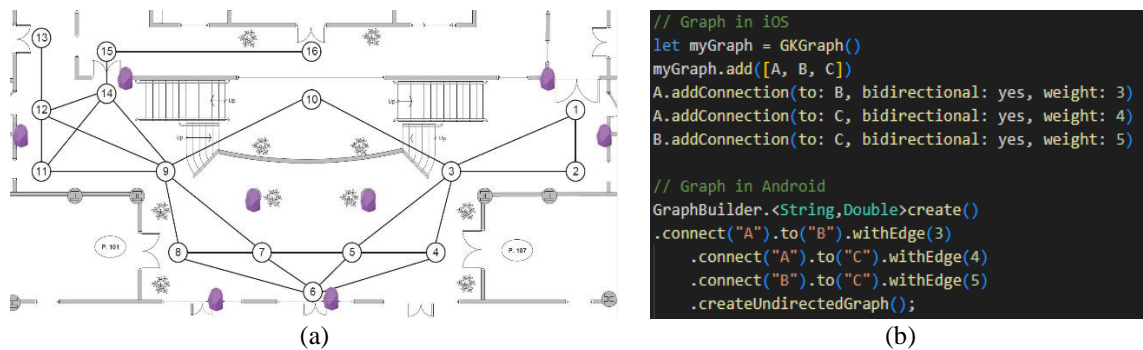


Figure 4. An example of how nodes are connected (a) under visual perspective and (b) by code

4. EVALUATION

4.1. Indoor localization application

We have built a sample application for the test. Figure 5 shows some screenshots of indoor navigation applications. The application was developed for indoor positioning and navigation purposes and applied to the G2 building, University of Engineering and Technology, Vietnam National University. With a simple interface and easy-to-use operation, the application is able to provide the location service with relative accuracy. The main functions of the application on both iOS and Android platforms include:

- Indoor map view: The application displays information about the map based on the location data of Google Map. When the user launches the application from the home screen, information about the indoor floor's plan is displayed. Since we use Google Map SDK for indoor map view purposes, we set the initial view at

- maximum zoom by default. Users can manipulate the map displayed on the screen through swiping and interactive gestures such as zoom in, zoom out, or rotating the indoor map.
- Positioning function: When signals from at least three iBeacon devices are available, the application matches the online RSSs with the radio map and delivers the user's location. This location is then marked directly on the map with a marker. When the user changes their position, the marker on the map changes according to the user's location as well.
 - Navigation function: The navigation function was designed with a search bar or picker. When the user touches the search box, it automatically displays a floating menu of all rooms. When the user enters the search box, the application suggests rooms related to the imported content. Once the user clicks on a recommended room, the application will deliver relevant information about that room, including the location and its description. The app then offers directions from the user's location to the searched room. When the user clicks on the get directions/go button, the application will display the route (red line) from the user's location to the location of the searched room. When the user changes location but is still in the process of receiving navigation information, the paths on the map automatically update according to the user's location in real-time.

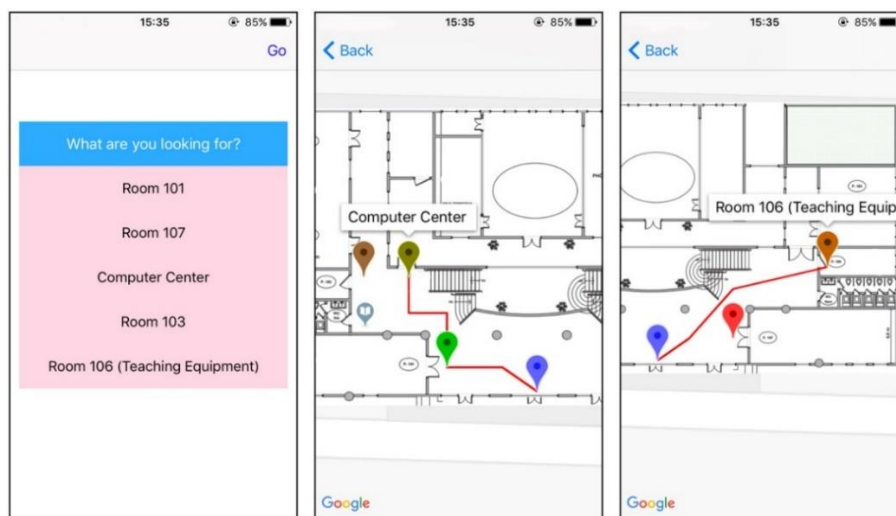


Figure 5. Screenshot of indoor positioning application in iOS platform. The app can be found at [26]. Another Android version can be found at [27]

4.2. The accuracy of our system

4.2.1. Experimental setup

To evaluate the system's performance, we integrate the system into an iOS app running on iPhone. The experiment is conducted on the 1st floor of the G2 building of the University of Engineering and Technology. The tested area is 30×11 m. Eight Estimote iBeacons are placed around the area, each placed at the height of 1.8 m. The beacons' position is shown in Figure 4(a). All beacons are set to the same configuration, with broadcasting power and advertising interval being 0 dBm and 100 milliseconds, respectively. The radio map is built by collecting real data in the lobby, as shown in Figure 4(a). This map is divided into small square area with a 1-meter edge. We follow the setup described in Figure 6 to assess the localization accuracy of the proposed system. A total of two experiments were performed. In which, the KNN and WKNN algorithms are considered in two conditions: i) only using raw RSS and ii) applying a Kalman filter on raw RSS.

4.2.2. Experiment results

Figure 7 shows the positioning errors of the proposed method without Kalman filter in Figure 7(a) and after noise removal using Kalman filter in Figure 7(b). As can be seen, WKNN does better than KNN in both cases. It is reasonable because WKNN utilizes more information than KNN, resulting in higher accuracy. In case 1, WKNN and KNN have the same performance, i.e., 1.6 m error in terms of the median. Also, in this case, WKNN, however, has better performance than KNN in terms of average error, i.e., 1.78 m compared to 1.9 m. In addition, the variance of WKNN is also lower than KNN. It is because WKNN uses a weight for each calculation from the online to the offline vector, which is equivalent to a regulatory factor of the output position.

We can see a similar trend in case 2. In this case, WKNN does better than KNN in both terms of median and average. The median and average error of WKNN and KNN is 1.18, 1.34, 1.24, and 1.39 m, respectively. WKNN and KNN in case 2 outperforms 25% and 16% WKNN and KNN in case 1, respectively. The variance in case 2 of both algorithms is also smaller than case 1. It indicates that noisy RSS can affect positioning results. Hence, it is recommended to apply filtering on raw RSS before feeding them into the matching algorithm.

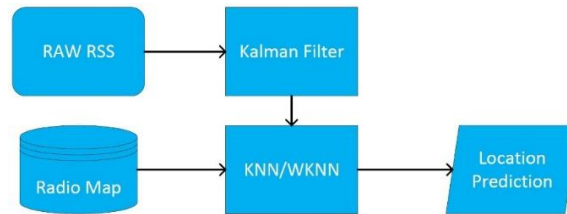


Figure 6. Experimental setup

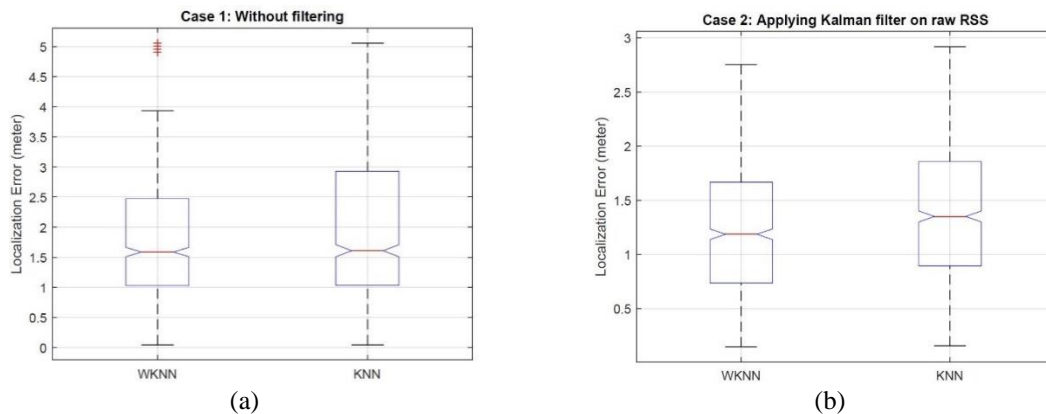


Figure 7. Localization error of WKNN and KNN (a) with filter and (b) without filter

5. CONCLUSION

This paper presents a framework that allows a service provider or programmer to build an indoor navigation system using iBeacon with available resources such as a smartphone or Google Maps SDK. The various problems are included for indoor positioning and navigation, such as hardware and software deployment for both Android and iOS. Besides, we also took into account the positioning accuracy by conducting two experiments related to the KNN algorithm as an estimator of the fingerprinting method. As the obtained results, it is recommended to use WKNN and filter on raw RSS. The procedure is partially extensible for other technologies, not only iBeacon but also Wi-Fi, RFID, UWB, or 5G system.




REFERENCES

- [1] V. Truong-Quang and T. Ho-Sy, "Maximum convergence algorithm for WiFi based indoor positioning system," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 5, pp. 4027–4036, Oct. 2021, doi: 10.11591/ijece.v11i5.pp4027-4036.
- [2] G. Schroerer, "A real-time UWB multi-channel indoor positioning system for industrial scenarios," in *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Sep. 2018, pp. 1–5, doi: 10.1109/IPIN.2018.8533792.
- [3] C.-H. Huang, L.-H. Lee, C. C. Ho, L.-L. Wu, and Z.-H. Lai, "Real-time RFID indoor positioning system based on Kalman-filter drift removal and heron-bilateration location estimation," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 3, pp. 728–739, Mar. 2015, doi: 10.1109/TIM.2014.2347691.
- [4] P. Spachos and K. N. Plataniotis, "BLE beacons for indoor positioning at an interactive IoT-based smart museum," *IEEE Systems Journal*, vol. 14, no. 3, pp. 3483–3493, Sep. 2020, doi: 10.1109/JSYST.2020.2969088.
- [5] Q. Liu, X. Yang, and L. Deng, "An iBeacon-based location system for smart home control," *Sensors*, vol. 18, no. 6, Jun. 2018, doi: 10.3390/s18061897.
- [6] R. Y.-Y. Chan, X. Bai, X. Chen, S. Jia, and X. Xu, "iBeacon and HCI in special education," in *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, May 2016, pp. 1533–1539, doi: 10.1145/2851581.2892375.
- [7] B. Molina, E. Olivares, C. E. Palau, and M. Esteve, "A multimodal fingerprint-based indoor positioning system for airports," *IEEE Access*, vol. 6, pp. 10092–10106, 2018, doi: 10.1109/ACCESS.2018.2798918.




- [8] S. Khruahong, X. Kong, K. Sandrasegaran, and L. Liu, "Develop an indoor space ontology for finding lost properties for location-based service of smart city," in *2018 18th International Symposium on Communications and Information Technologies (ISCIT)*, Sep. 2018, pp. 54–59. doi: 10.1109/ISCIT.2018.8588014.
- [9] L.-W. Chen, T.-P. Chen, H.-M. Chen, and M.-F. Tsai, "Crowdsourced children monitoring and finding with holding up detection based on internet-of-things technologies," *IEEE Sensors Journal*, vol. 19, no. 24, pp. 12407–12417, Dec. 2019, doi: 10.1109/JSEN.2019.2938324.
- [10] C.-C. Chiu, J.-C. Hsu, and J.-S. Leu, "Implementation and analysis of hybrid wireless indoor positioning with iBeacon and Wi-Fi," in *2016 8th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, Oct. 2016, pp. 80–84. doi: 10.1109/ICUMT.2016.7765336.
- [11] X.-Y. Lin, T.-W. Ho, C.-C. Fang, Z.-S. Yen, B.-J. Yang, and F. Lai, "A mobile indoor positioning system based on iBeacon technology," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug. 2015, pp. 4970–4973. doi: 10.1109/EMBC.2015.7319507.
- [12] X. Zhang, S. Zhang, and S. Huai, "Low-power indoor positioning algorithm based on iBeacon network," *Complexity*, vol. 2021, pp. 1–10, May 2021, doi: 10.1155/2021/8475339.
- [13] M. Lewandowski, B. Placzek, M. Bernas, and P. Szymała, "Road traffic monitoring system based on mobile devices and Bluetooth low energy beacons," *Wireless Communications and Mobile Computing*, vol. 2018, pp. 1–12, Jul. 2018, doi: 10.1155/2018/3251598.
- [14] N. Ye, J. Luo, S. Zhao, and J. Yan, "Application of iBeacon in indoor positioning and navigation," *Journal of Physics: Conference Series*, vol. 1575, no. 1, Jun. 2020, doi: 10.1088/1742-6596/1575/1/012056.
- [15] F. Zafari and I. Papapanagiotou, "Enhancing iBeacon based micro-location with particle filtering," in *2015 IEEE Global Communications Conference (GLOBECOM)*, Dec. 2015, pp. 1–7. doi: 10.1109/GLOCOM.2015.7417504.
- [16] Apple, "Getting started with iBeacon." Accessed: May 22, 2022. [Online] Available: <https://developer.apple.com/ibeacon/Getting-Started-with-iBeacon.pdf>
- [17] J. Rezazadeh, R. Subramanian, K. Sandrasegaran, X. Kong, M. Moradi, and F. Khodamoradi, "Novel iBeacon placement for indoor positioning in IoT," *IEEE Sensors Journal*, vol. 18, no. 24, pp. 10240–10247, Dec. 2018, doi: 10.1109/JSEN.2018.2875037.
- [18] T.-M. T. Dinh, N.-S. Duong, and Q.-T. Nguyen, "Developing a novel real-time indoor positioning system based on BLE beacons and smartphone sensors," *IEEE Sensors Journal*, vol. 21, no. 20, pp. 23055–23068, Oct. 2021, doi: 10.1109/JSEN.2021.3106019.
- [19] Y. Chen, J. Yang, W. Trappe, and R. P. Martin, "Impact of anchor placement and anchor selection on localization accuracy," in *Handbook of Position Location*, Wiley, 2019, pp. 435–465. doi: 10.1002/9781119434610.ch13.
- [20] "Maps SDK for Android overview," *Google Developers*, 2020. <https://developers.google.com/maps/documentation/android-sdk/overview> (accessed May 22, 2022).
- [21] "Maps SDK for iOS Overview," *Google Developers*, 2020. <https://developers.google.com/maps/documentation/ios-sdk/overview> (accessed May 22, 2022).
- [22] Y. Wang, Y. Shang, W. Tao, and Y. Yu, "Target positioning algorithm based on RSS fingerprints of SVM of fuzzy Kernel clustering," *Wireless Personal Communications*, vol. 119, no. 4, pp. 2893–2911, Aug. 2021, doi: 10.1007/s11277-021-08377-4.
- [23] H. Zhang, Z. Wang, W. Xia, Y. Ni, and H. Zhao, "Weighted adaptive KNN algorithm with historical information fusion for fingerprint positioning," *IEEE Wireless Communications Letters*, vol. 11, no. 5, pp. 1002–1006, May 2022, doi: 10.1109/LWC.2022.3152610.
- [24] "GameplayKit framework," *Apple*, 2020. <https://developer.apple.com/documentation/gameplaykit/gkgraph> (accessed May 22, 2022).
- [25] "GraphBuilder library," *Intel Corporation*, 2023. <https://github.com/intel/graphbuilder> (accessed Apr 19, 2023).
- [26] N.-S. Duong and T.-M. D. Thi, "Indoor positioning application for iOS," *Software – Online*. <https://shorturl.at/aimQZ> (accessed May 22, 2022).
- [27] N.-S. Duong and T.-M. D. Thi, "Indoor positioning application for Android," *Software – Online*. <https://shorturl.at/elotI> (accessed May 22, 2022).

BIOGRAPHIES OF AUTHORS



Thai-Mai Dinh Thi    graduated from the Post and Telecommunication Institute of Technology, Vietnam, in 2006. She received the M.Sc. degree from the University of Paris Sud 11, France, in 2008, and the Ph.D. degree from the VNU University of Engineering and Technology, Hanoi, Vietnam, in 2016. She is the Deputy Head of the Department of Telecommunications Systems, Faculty of Electronics and Telecommunications, VNU University of Engineering and Technology. Her research interests include 5G mobile networks, wireless communications, and localization techniques. She serves as reviewer for IEEE Sensors Journal, IEEE Transactions on Instrumentation and Measurement, Geo-spatial Information Science and so on. She can be contacted at email: dtmai@vnu.edu.vn.



Ngoc-Son Duong    was born in Bac Giang, Vietnam. He received the B.E. and M.Sc. degrees in electronics and telecommunications from the VNU University of Engineering and Technology, Vietnam National University, Hanoi, Vietnam, in 2018 and 2020, respectively, where he is currently pursuing the Ph.D. degree. His research interests include RF-based localization and wireless communications. He serves as reviewer for several IEEE journals in the field of indoor positioning. He is IEEE student member. He can be contacted at email: sondn24@vnu.edu.vn.