

## Research Article

# Seamless Guidance System Combining GPS, BLE Beacon, and NFC Technologies

Rung-Shiang Cheng,<sup>1</sup> Wei-Jun Hong,<sup>2</sup> Jheng-Syun Wang,<sup>1</sup> and Kawuu W. Lin<sup>2</sup>

<sup>1</sup>Department of Computer and Communication, Kun Shan University, No. 195, Kunda Road, Yongkang District, Tainan City 710, Taiwan

<sup>2</sup>Department of Computer Science and Information Engineering, National Kaohsiung University of Applied Sciences, No. 415, Jiangong Road, Sanmin District, Kaohsiung City 807, Taiwan

Correspondence should be addressed to Rung-Shiang Cheng; [rscheng@mail.ksu.edu.tw](mailto:rscheng@mail.ksu.edu.tw)

Received 2 October 2015; Accepted 7 December 2015

Academic Editor: Jong-Hyouk Lee

Copyright © 2016 Rung-Shiang Cheng 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.

Users rely increasingly on Location-Based Services (LBS) and automated navigation/guidance systems nowadays. However, while such services are easily implemented in outdoor environments using Global Positioning System (GPS) technology, a requirement still exists for accurate localization and guidance schemes in indoor settings. Accordingly, the present study proposes a system based on GPS, Bluetooth Low Energy (BLE) beacons, and Near Field Communication (NFC) technology. Through establishing graphic information and the design of algorithm, this study develops a guidance system for indoors and outdoors on smart phones, wishing to give user perfect smart life through this system. The proposed system is implemented on a smart phone and evaluated on a student campus environment. The experimental results confirm the ability of the proposed app to switch automatically from an outdoor mode to an indoor mode and to guide the user to requested target destination via the shortest possible route.

## 1. Introduction

According to statistics published by the International Telecommunication Union (ITU), the number of mobile devices in the world reached 6.835 billion at the end of 2013. Furthermore, the number of devices is still growing. As wireless technology continues to improve and wireless networks are ever more extensively deployed, the feasibility of developing Location-Based Services (LBS) has attracted growing interest [1]. LBS have many advantages from a user perspective, including convenience, efficiency, and fun. As a result, they are now widely applied in social networks, traffic and geographic search systems, and even public safety [2, 3]. Of the many functions offered by LBS, those of positioning localization and guidance are some of the most useful. According to previous research, adults spend around 86.9% of their time indoors, 5.5% in vehicles, and 7.6% outdoors [4]. Thus, in realizing seamless LBS applications, it is necessary

to develop localization and guidance schemes capable of working in both indoor and outdoor environments and switching between the two modes automatically as required.

Global Positioning System (GPS) technology is used widely in the navigation, tourism, measurement, and engineering fields. However, the success of GPS depends on a strong signal between the user and the navigational satellite. Thus, while GPS functions well in open outdoor environments, its performance suffers dramatically in mountainous areas or build-up urban areas. Furthermore, the signals are unable to penetrate through building structures, and hence GPS is of only limited use in indoor environments.

The literature thus contains various alternative proposals for performing indoor localization. For example, in [5, 6], the user position is estimated using a wireless communication signal, while, in [7], a visible light communication system is used. The authors in [8] performed indoor positioning using radiofrequency identification (RFID) tags. In [9], user

positioning was performed by mapping the activities of the user to the positions in the indoor environment at which these activities were known to be performed. Finally, in [10], multiple indoor positioning technologies were combined in order to improve the localization accuracy.

Although mobile devices are invaluable in daily life nowadays, their usefulness is limited by their short battery lives, which prompts the need for frequent recharging. To address this problem, many mobile devices use Bluetooth Low Energy (BLE) technology to realize wireless communication connections. BLE has many advantages as a connection technology, including a stable signal, an ease of distribution, a low cost, and widespread compatibility with existing wireless devices. Furthermore, BLE beacons have an operating life of several months using only a simple button cell battery [11]. As a result, BLE has significant potential as an enabling technology for indoor LBS applications.

Owing to indoor environment being a complicated space, GPS signals can not be used and the spreading of wireless is easily affected by the interior structure of the building; therefore, it urgently needs precise technologies and a lot of spherical auxiliary electronic devices, leading to the difficulty in indoor precise positioning. Thus, for many years, these well-known positioning technologies such as GPS or Google Map services are usually applied in public construction fundamental facilities, road, or outdoor large area; however, along with the urbanization of living environment, there are more and larger complex buildings, increasing the needs for indoor positioning. Near Field Communication (NFC) is a short-range wireless connectivity standard which enables communications to be achieved between devices simply by touching them together or bringing them into very close proximity of one another (typically, less than 10 cm). NFC has found widespread use nowadays for such applications as loyalty schemes, home healthcare, public transport payment, ticketing, mobile workforce management, and so on. With the ability it provides to infer the user position with an ultra-high degree of precision, NFC also has significant potential for indoor localization purposes.

Thus, to enhance the application of positioning system and enable smart phones to become key equipment of augmented humanity which can effectively enhance the convenience, this study is based on short-distance wireless communication technology, combining with NFC, BLE beacon, and GPS technology to develop a positioning guidance system which is suitable for indoor buildings. This study combines GPS, BLE, and NFC technologies to realize a seamless indoor-outdoor user localization and guidance app suitable for implementation on a smart mobile device. In the proposed scheme, user localization is performed using conventional GPS technology in the outdoor environment. However, when the user enters an indoor space, the app switches automatically to an indoor mode, and user positioning is performed by means of BLE beacons and NFC tags. The positioning information obtained via the various technologies is combined with map information (outdoor and indoor) to realize a guidance system capable of leading the user to the requested target destination via the shortest available route.

The design of the system includes “storage data design,” “positioning method design,” “algorithm design,” and “graphic design.” In between, for indoor map establishment, this study designs a method which can assist in establishing positioning point storage function through wireless signal exploration (indoor space area as unit positioning base). Outdoor map uses GPS as positioning base to obtain the location information, using route map to display the guidance results. When users are in outdoors, this app will automatically adopt “outdoor mode” to obtain GPS positioning to provide users with guidance information.

The remainder of this paper is organized as follows. Section 2 reviews the GPS, BLE, and NFC technologies used for localization purposes in this study and introduces the path-finding algorithm used to realize the indoor guidance system. Section 3 describes the system framework and implementation. Section 4 presents and discusses the experimental evaluation results. Finally, Section 5 provides some brief concluding remarks.

## 2. Background Knowledge

This section commences by describing the GPS, BLE beacon, and NFC technologies used in the present study to develop the proposed positioning and guidance system. The shortest-path algorithm used to accomplish indoor guidance is then briefly introduced. Figure 1 presents a schematic illustration of the respective communication distances of the GPS, beacon, and NFC technologies.

*2.1. GPS.* GPS is a middle-distance global tracking satellite guidance system with a coverage area of more than 98% of the earth’s surface. GPS can be used by any enabled device capable of receiving its signals and has the advantage of anonymity in that the user’s position is not recorded as part of the communication process. However, GPS relies on the availability of a clear Line of Sight (LOS) between the user device and the satellite system. As a result, it provides only a limited positioning capability in indoor environments [12].

*2.2. BLE.* Bluetooth Low Energy (BLE) is a communication standard designed to enable short-range wireless devices to operate for months or even years on a single coin cell battery. When combined with beacon technology, BLE provides a highly effective method for estimating the position of the user relative to certain predefined monitoring spots. BLE operates over a distance of up to 50 m and provides the means to customize the LBS offered to the user based on their physical location. For example, certain ads can be pushed to the user device as the user approaches a particular sales counter in a store. Similarly, the user may be presented with different notification messages and application events as he or she moves across the boundary separating one monitored area from another.

The literature contains various proposals for integrating the BLE standard with beacon technology in order to support user localization, including iBeacon [13], Gimbal [14], and AltBeacon [15]. The beacons used in such systems periodically broadcast a wireless radio signal advertising

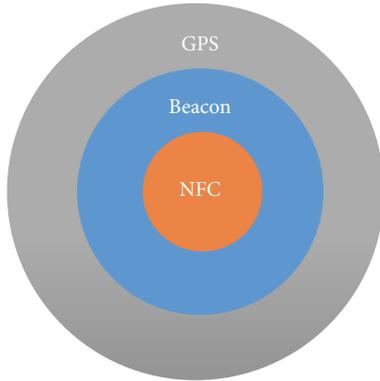


FIGURE 1: Communication distances of three positioning technologies.

their presence. As described above, BLE operates over a range of up to 50 m. Hence, in the event that the signal is detected (sighted) by a proximity-enabled user device, the user position can be inferred with an error of no more than 50 m. As a result, BLE/beacon technology provides a low-cost and energy-efficient solution for performing user localization with a medium degree of accuracy. The localization system proposed in the present study utilizes the Gimbal Series 10 beacon produced by Qualcomm (see Figure 2). The periodic message broadcast by the beacon contains many items of information, including the Factory ID, RSSI value, name, battery capacity, and temperature. The Factory ID uniquely identifies the beacon. Thus, by storing the Factor IDs and physical beacon locations in a database, the ID information contained in the message provides an efficient and reliable method for estimating the position of the user.

2.3. *NFC*. Near Field Communication (NFC) is an ultra-short distance wireless communication technology based on radiofrequency identification (RFID). NFC utilizes signal attenuation technology to enable devices to conduct noncontact point-to-point data transmissions over distances of up to approximately 10 cm (3.9 inches). NFC is currently used for such applications as automated payment, ticketing, loyalty schemes, and so forth. However, with its high bandwidth and low energy consumption [16], NFC (see Figure 3) also has significant potential for highly precise indoor positioning. As with the Gimbal beacon, each NFC chip has a unique ID number assigned to it by the manufacturer. Thus, by associating the ID with a physical location and storing this information in a database, the position of the user can be inferred with an extremely high degree of precision each time a sensing event occurs.

2.4. *Shortest-Path Algorithm*. Determining the shortest route between a start point and a target end point given the availability of multiple paths between them is a common problem in many walks of life [17]. The guidance system proposed in this study utilizes the algorithm proposed by Dijkstra [18]. Since, of the various algorithms available, it has the advantages of concise algorithm, the optimal solution can be obtained.



FIGURE 2: Gimbal beacon.

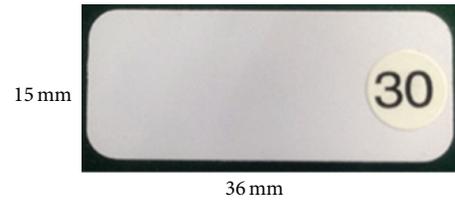


FIGURE 3: NFC tag.

### 3. System Framework and Implementation

The app proposed in this study provides the user with a seamless positioning and guidance service as he or she moves from an outdoor environment to a target destination in an indoor environment or vice versa. In other words, the app switches automatically not only from an outdoor mode to an indoor mode, but also from an indoor mode to an outdoor mode. As described in Introduction, positioning in the outdoor environment is performed using conventional GPS technology, while that in the indoor environment is performed using BLE beacon and NFC technologies. For both environments, the guidance function is achieved using map information stored in a remote server and downloaded to the user device as required. For illustration purposes, the present study considers the localization/guidance problem for the case of a student campus environment containing many buildings scattered over a wide geographic area with many floors and rooms within each building. As described in the following sections, the system framework comprises four design components, namely, (1) storage data design, (2) positioning method design, (3) shortest-path algorithm design, and (4) map structure design.

3.1. *Storage Data Design*. To simplify the data storage and management task, four different data structures are used to support the different functionalities of the system, namely, an outdoor map structure, an indoor map structure, a beacon positioning data structure, and an NFC positioning data structure. Table 1 shows the data structure used to store and maintain the outdoor map information. As shown, the structure comprises six fields, namely, id (the primary index key); name (used to store the name of the destination); address (used to store the address of the destination); latitude and longitude (used to store the geographical coordinates

TABLE 1: Data format for outdoor map.

Name	Type	Remark
id	Int(11)	Primary
name	Text	Destination name
address	Text	Address
latitude	Text	Latitude
longitude	Text	Longitude
info	Text	Brief info

TABLE 2: Data format for indoor map.

Name	Type	Remark
id	int(11)	Primary
map_name	Text	map code
node	int(11)	node code
x	int(11)	X-axis coordinate
y	int(11)	Y-axis coordinate

of the destination); and information (used to store a brief description of the building).

Table 2 shows the data structure used to store the indoor map information. The structure contains five fields, namely, *id* (the primary index key); *map-name* (used to store the code of the indoor map); *node* (used to indicate the name of the nodes (rooms/locations) in the indoor map); and *x* and *y* (used to store the *x*- and *y*-coordinates of the nodes in the indoor map).

Table 3 shows the data structure used to store the beacon location information. As shown, the structure contains six fields, namely, *id* (the primary search index); *node* (the node identifier in the indoor map); *fid* (the unique ID assigned to the beacon by the factory); *name* (the name of the room/location in which the tag is located); *middle* (the name of the building in which the tag is located); and *far* (the name of the general campus area in which the building is located). Table 4 represents the indoor map information used to store graphic object-oriented information; clicking each point represents a position on the screen. Data list design includes *id* primary index key and *map\_name* is the symbol of the map for the usage of server inquiry. Node is the present representing node symbol and *x* and *y* are the coordinates of the center of the node.

**3.2. Positioning Method Design.** As shown in Figure 4, the localization/guidance app proposed in this study resides by default in the “outdoor mode” and uses conventional GPS technology to locate the position of the user. More specifically, the system acquires the current latitude and longitude information from the GPS and uploads this information together with the Device ID to a remote server (see Figure 5). On receiving this information, the server interrogates the coordinate information and returns the appropriate outdoor map to the user device using the JSON format shown in Algorithm 1.

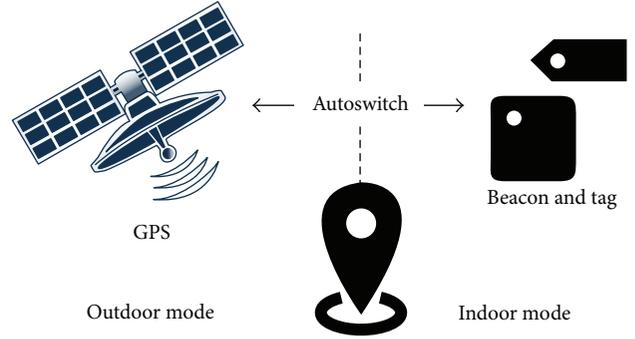


FIGURE 4: Automatic mode switching.

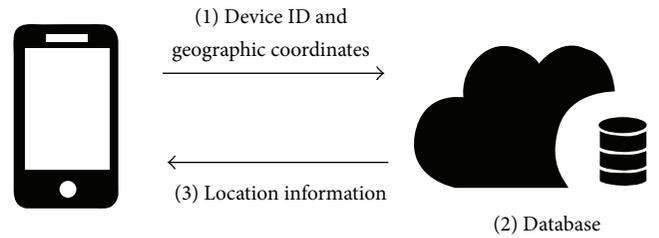


FIGURE 5: Outdoor map data flow.

TABLE 3: Beacon data format.

Name	Type	Remark
id	int(11)	Primary
node	int(11)	node
fid	text	Factory ID
name	text	Room name
middle	text	Building name
far	text	Big area name

TABLE 4: NFC data format.

Name	Type	Remark
id	int(11)	Primary
node	int(11)	node
uid	text	Tag uid
name	text	Room name
middle	text	Building name
far	text	Big area name

JSON message sent back includes the name, address, latitude and longitude, and building introduction of the destination whose format is shown in Algorithm 1.

When the user moves from the outdoor environment to an indoor environment, the app switches automatically to an “indoor mode” and launches an indoor positioning routine. If the user is within wireless range of a beacon, a sighting event occurs and the device uploads both its own ID and that of the beacon to the server. Utilizing the Factor ID as a key, the server retrieves the approximate location of the user and

```
{
  "id": 1,
  "name": "School",
  "address": "No.195, Kunda Rd.",,
  "latitude": "22.996175",
  "longitude": "120.252970",
  "info": "Kun Shan University"
}
```

ALGORITHM 1: JSON data format.

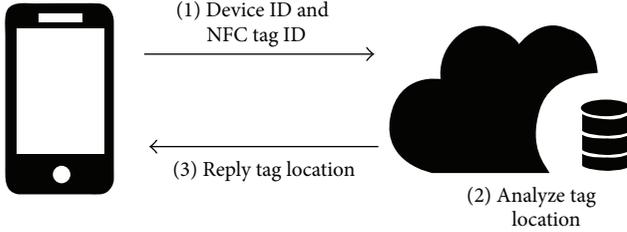


FIGURE 6: NFC positioning process.

returns this information to the user device. When the user's smart phone approaches a NFC tag, the user can send tag ID along with Device ID back to the server after the mobile reads the set NFC tag. The server will conduct index from the NFC data list and send the index results back the app to obtain the user's location and conduct indoor positioning as shown in Figure 6.

**3.3. Shortest-Path Algorithm Design.** After obtaining the user location, the app downloads the map from the server. Specifically, the app informs the server of the Device ID and the required map ID and the server searches its database for the corresponding map and returns it to the user device.

Let  $G = (V, E)$  denote an indoor map, where  $V$  is the set of nodes in the map and  $E$  is the set of connecting edges (paths). In constructing the map, the server hosts a database with four columns, namely, ID (primary index key), node ID, adjacent node ID, and distance between adjacent nodes. Taking Node 2 in Figure 8 as an example, let the nodes adjacent to Node 2 be denoted as Node 0, Node 1, Node 3, and Node 4, respectively. Furthermore, let the distances of these nodes from Node 0 be equal to 4, 2, 9, and 2, respectively. The position of Node 2 relative to its neighboring nodes can therefore be expressed in the form shown in Table 5.

The node positioning information is communicated from the server to the device using the JSON format shown in Algorithm 2. Note that the information object includes such information as the name of the map, the number of nodes in the map, and the number of adjacent roads in the map. Similarly, the algorithm object includes the adjacent nodes, the costs of the adjacent paths, and the ID of the present node.

The information provided in the JSON message in Algorithm 2 provides the app with the relevant indoor shortest-path map. However, the Dijkstra shortest-path algorithm requires the input information to be presented in the

```
{
  "information": {
    "name": "Example",
    "node": 5,
    "road": 14
  },
  "algorithm": [
    {
      "neighbor": [
        0,
        1,
        3,
        4
      ],
      "cost": [
        4,
        2,
        9,
        2
      ],
      "this": 2
    }
  ]
}
```

ALGORITHM 2: JSON format for Node 2.

TABLE 5: Node positioning information.

Name	Type	Storage results
node	int(11)	2
neighbor_A	text	0
distance_A	int(11)	4
neighbor_B	text	1
distance_B	int(11)	2
neighbor_C	text	3
distance_C	int(11)	9
neighbor_D	text	4
distance_D	int(11)	2

form of a matrix. Therefore, in implementing the indoor guidance function, the JSON map information must first be converted into a matrix form. For example, the illustrative layout in Figure 8 contains five nodes and should therefore be converted to a  $5 \times 5$  matrix of the form shown in Table 6.

For a large map comprising multiple nodes, creating the map and maintaining it over time as additional nodes are added or existing nodes are removed represent a huge task if performed manually. For example, for a map with  $n$  nodes, maintaining the corresponding adjacent matrix given a change in any one of the nodes incurs a time complexity of  $O(n^2)$ . Accordingly, in the app proposed in this study, the matrix construction task is performed automatically using the function shown in (1), in which  $i$  is the number of the current node,  $v$  is the number of the adjacent node, and  $e$  is the cost of the path between them. When presented with the node map (constructed manually by the system developer),

TABLE 6: Dijkstra adjacent matrix.

Node	0	1	2	3	4
0	0	8	4	INF	INF
1	8	0	2	7	INF
2	4	2	0	9	2
3	INF	7	9	0	2
4	INF	INF	2	6	0

the matrix construction algorithm takes the current node and adjacent node information as an input and uses (1) to automatically generate the corresponding  $n \times n$  adjacent matrix with a time complexity of  $O(n)$ :

$$\text{graph}[i].\text{adjacentEdge}(v, e). \quad (1)$$

For example, taking Node 2 in Figure 8 for illustration purposes once again, the relation between Node 2 and its adjacent nodes has the form shown in (2). Taking the  $i$ ,  $v$  and  $e$  information given in (2), the algorithm automatically constructs the matrix shown in Figure 9:

$$\begin{aligned} &\text{graph}[2].\text{adjacentEdge}(0, 4), \\ &\text{graph}[2].\text{adjacentEdge}(1, 2), \\ &\text{graph}[2].\text{adjacentEdge}(3, 9), \\ &\text{graph}[2].\text{adjacentEdge}(4, 2). \end{aligned} \quad (2)$$

Taking Figure 8 as example, the app reads the above adjacent relation and automatically converts to the matrix adopted by the algorithm as shown in Figure 9. It inputs the converted matrix into Dijkstra's algorithm to estimate the shortest path by Dijkstra's algorithm.

Dijkstra's shortest-path algorithm calculates the shortest path from one point in a node set to any other node in the same set. Assume that Node 2 in Figure 8 is chosen as the start point and Node 3 is chosen as the destination. Three feasible paths exist between the two nodes, namely, a direct path from Node 2 to Node 3 and two two-node paths via Nodes 1 and 4, respectively. The three paths have total distances of 9, 9, and 8, respectively. Thus, in accordance with Dijkstra's algorithm, path 2-4-3 is chosen as the shortest path (see Figure 10).

**3.4. Map Structure Design.** In general, the success of any app is determined to a large extent by the appearance and intuitiveness of its graphical user interface (GUI). For a guidance system such as that proposed in the present study, a pictorial map with too much detailed information will serve simply to confuse the user. Consequently, in the proposed app, the indoor and outdoor maps are presented in the form of metro-like maps, in which the key locations (e.g., buildings, offices, classrooms, and toilets) are represented as nodes and the distances between them are indicated by numerals placed alongside the corresponding paths (see Figure 11).

As discussed above, having determined the user's current location, the app requests the appropriate map from the server and then stores the received map in the device (see Figure 7). Notably, the map is downloaded in its entirety

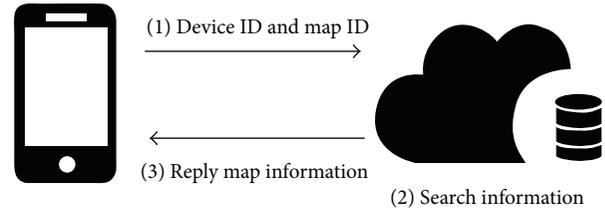


FIGURE 7: Map algorithm download process.

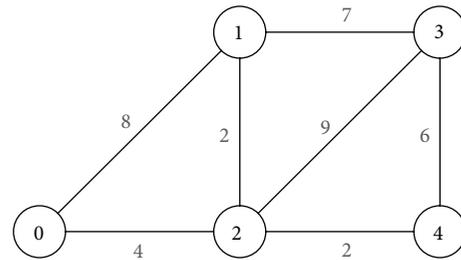


FIGURE 8: Node layout scheme.

Dijkstra					
SHOW MATRIX					
ADJACENT MATRIX					
0	8	4	INF	INF	
8	0	2	7	INF	
4	2	0	9	2	
INF	7	9	0	6	
INF	INF	2	6	0	

FIGURE 9: Automatic conversion of adjacent node relations to matrix form.

Dijkstra	
Source	2
Destination	3
Start	
2 -> 4 -> 3	Distance: 8

FIGURE 10: Output of Dijkstra shortest-path algorithm.

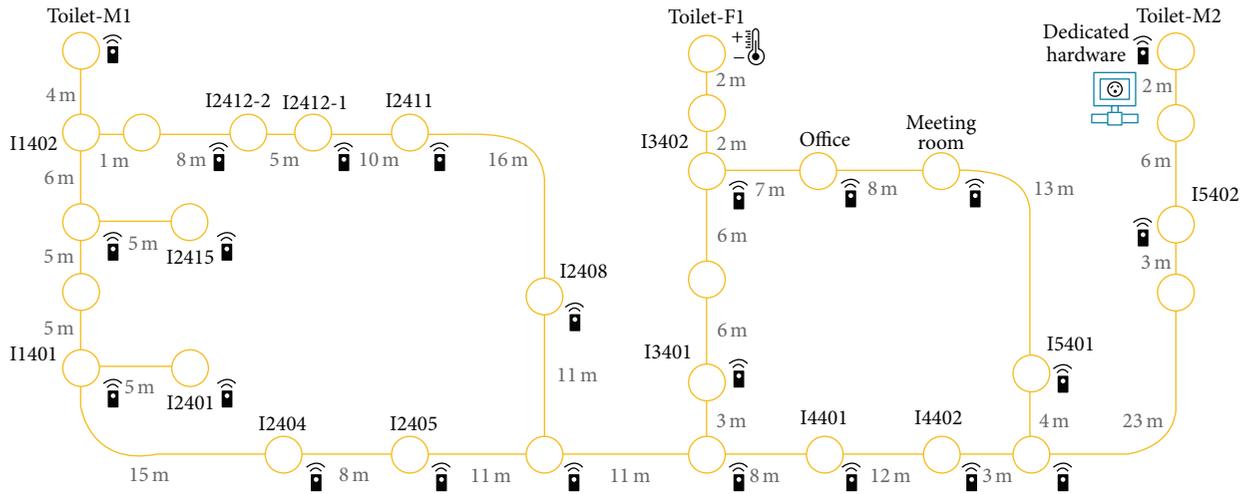


FIGURE 11: Map representation.

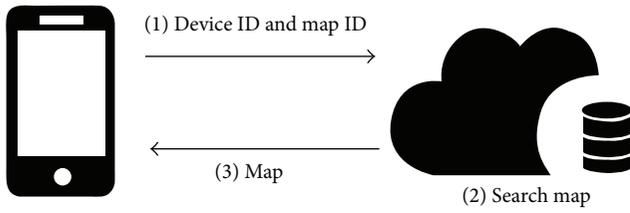


FIGURE 12: Flow chart showing map download process.

(Figure 12), and hence the need for repeated download events is avoided. Having downloaded the map, the user then selects the target destination (i.e., node) for which they require routing information. To ensure smooth node selection, the app stores an acceptable touch range error in addition to the coordinates of each node center. The corresponding JSON message exchange between the app and the server thus has the form shown in Algorithm 3.

When the user touches the screen to select a particular destination node, the app determines the intended node in accordance with (3), in which  $X_1$  and  $Y_1$  are the stored coordinates of the node center,  $X_2$  and  $Y_2$  are the coordinates of the point at which the user touches the screen, and  $r$  is the allowable touch range error:

$$|X_1 - X_2|^2 + |Y_1 - Y_2|^2 < r^2. \quad (3)$$

To support the node selection process described above, the center coordinates of each node must first be established and stored in the device. Estimating the node centers one by one and inputting them into the database are a huge task if performed manually. Accordingly, this study provides a tool for automating the node input process by enabling the map manager to simply mark the scope of each node on the map (see Figure 13). Having marked the scope, the tool estimates

```

{
  "information": {
    "name": "Example",
    "photo": {
      "width": 1503,
      "height": 947,
      "radius": 100
    }
  },
  "algorithm": [
    {
      "touch": {
        "X": 110,
        "Y": 775
      },
      "this": 0
    },
    {
      "touch": {
        "X": 750,
        "Y": 150
      },
      "this": 1
    }
  ]
}
    
```

ALGORITHM 3: JSON format for map output information.

the coverage scope between the center of the nodes and the nodes on the map and uploads the related information to the database.

Having established the user's present location and his or her intended destination, the app invokes Dijkstra's shortest-path algorithm and marks the suggested route pictorially on the node map, as shown in Figure 14.

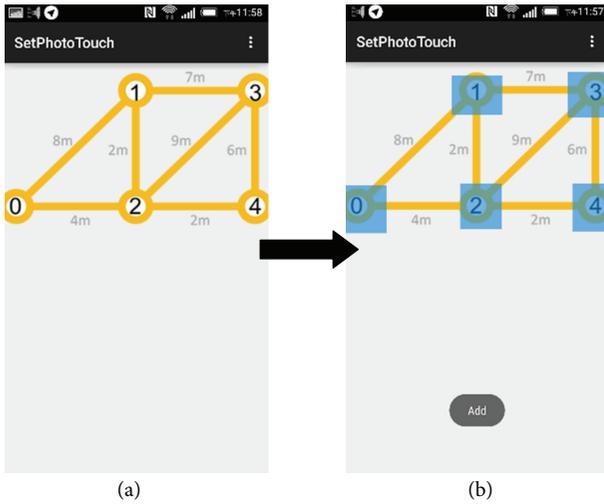


FIGURE 13: Node coordinate definition process.

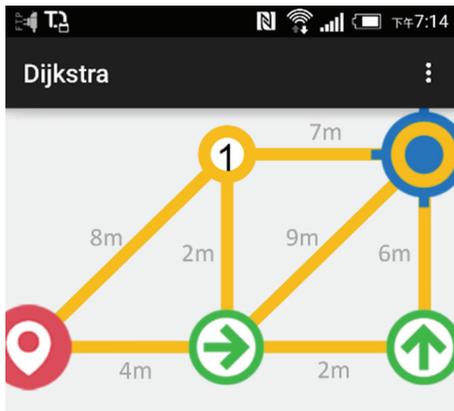


FIGURE 14: Suggested route interface.

## 4. Implementation Results

**4.1. Beacon Signal Measurement.** Localization experiments were performed using three commercial mobile phones, namely, HTC One, Sony Xperia Z1 Compact, and Samsung Galaxy S4. Each device was placed at three different distances from a beacon (i.e., 1, 2, and 4 m), and the distance measurements obtained by the device were then sampled 100 times over a 30-minute interval.

The corresponding results are presented in Figures 15–17. For a device-to-beacon distance of 1 m, the distance measurements obtained by the Samsung and HTC devices deviate quite significantly from the actual distance and vary markedly over the sampling period (see Figure 15). By contrast, the measurement results obtained using the Sony device are both closer to the true value and more stable. For a device-to-beacon distance of 2 m, the measurements obtained using the HTC and Samsung devices vary significantly over

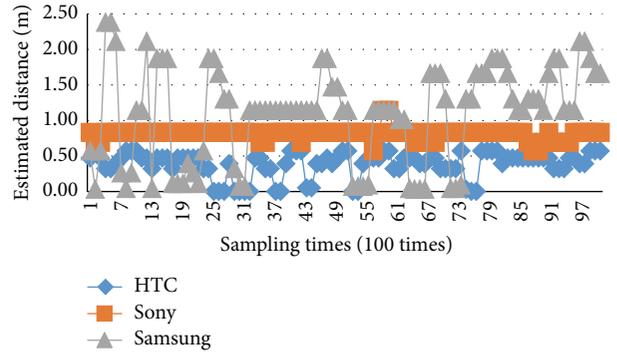


FIGURE 15: Signal measurement results at distance of 1 m from beacon.

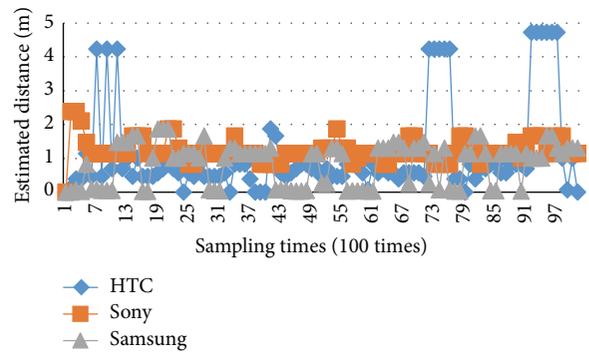


FIGURE 16: Signal measurement results at distance of 2 m from beacon.

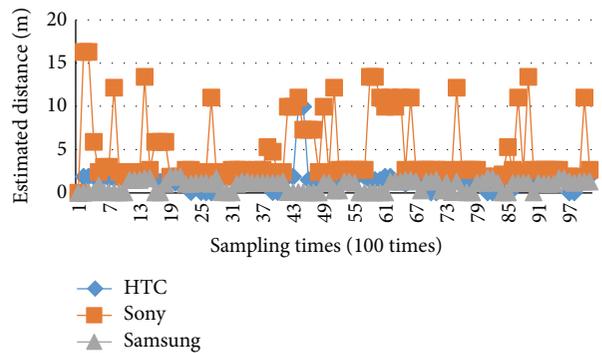


FIGURE 17: Signal measurement results at distance of 4 m from beacon.

the 30-minute interval. However, the Sony device again achieves an accurate and relatively stable measurement performance (see Figure 16). Finally, for a measurement distance of 4 m, all of the devices achieve a poor localization performance (see Figure 17).

In general, the results presented in Figures 15–17 show that none of the devices provides an absolutely precise localization performance. Hence, in implementing the proposed app, the measurement distances obtained by the devices were

quantified using the proximity value provided by the beacon, namely, a numerical value in the range of 1~3. A value of 1 was taken to indicate a close proximity of the device to the beacon, and hence the beacon location was used as a reference in estimating the current location measurement. By contrast, a value of 3 was taken to indicate a greater distance of the device from the beacon, and hence the location measurement obtained using the device was taken as an approximate value only.

**4.2. System Implementation and Function Display.** Performance evaluation trials were performed on a university campus in Taiwan, with the aim being to guide the user to a specific target in the building shown in Figure 18. As shown, 7 beacons and 31 NFC tags were placed at appropriate points throughout the experimental field, for example, on the doors of the main rooms in the building, at the entrances to staircases, and at forks or corners in the corridors. Due to their lower cost, the NFC tags greatly outnumbered the beacons (Figure 19) and were placed with an approximate spacing of 2~15 m.

When the app is launched, the system first checks the status of the communication services of the user device and issues a notification message if required (see Figure 20, e.g.). The system then performs a localization routine to establish the user position. If the app senses a GPS signal, it loads the outdoor map and marks the user location accordingly. By contrast, if the app detects a beacon or NFC signal, it loads the appropriate indoor map and again marks the user location as appropriate. Having received the map (indoor or outdoor), the user selects the destination node (e.g., a campus building or a room within the present building), and the app launches the shortest-path route discovery routine and marks the identified route on the map accordingly.

As shown in Figure 21(a), if the user is outdoors, the app automatically lists the main destinations within the closest building (Figure 21(b)) and indicates the user's location on the map (Figure 21(c)). The app then asks the user if a guidance function is required (Figure 22). If the user requests guidance, the app searches for the shortest-path to the selected destination and then plots the route on the map (Figure 23).

**4.3. Experimental Design and Results.** The performance of the proposed app was evaluated by comparing the time spent by four users in finding their way from the main campus entrance to a particular classroom within a certain building with and without the assistance of the localization/guidance system, respectively. In performing the experiments, the process of navigating to the classroom was separated into six steps, namely, (1) looking for campus map; (2) finding the target building; (3) searching for indoor floor layout; (4) reaching the floor; and (5) reaching the destination. Thus, we record the spent time in various steps as reference. As shown in Figures 24 and 25, the total time spent by each user in reaching the target was divided into four separate times, namely, the time taken in moving from the main campus gate to the first destination sign at Spot A; the time spent in

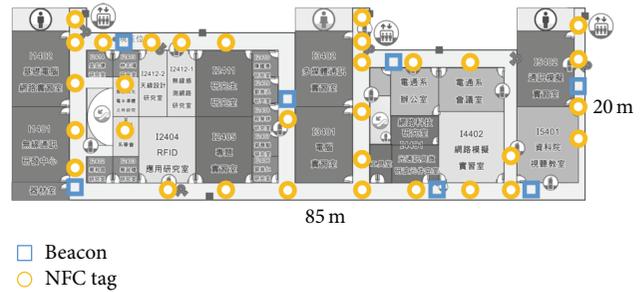


FIGURE 18: IoT building in the Department of Computing and Communication at Kun Shan University, Taiwan.

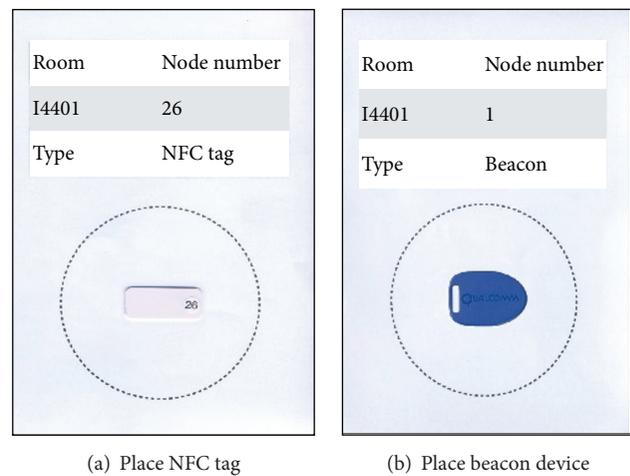


FIGURE 19: NFC tag and beacon device.

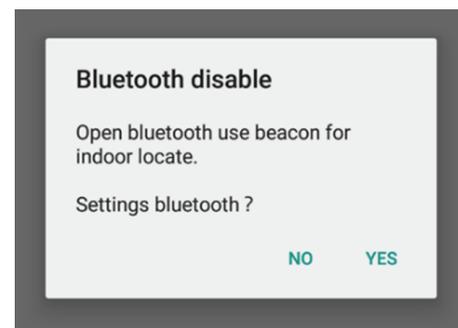
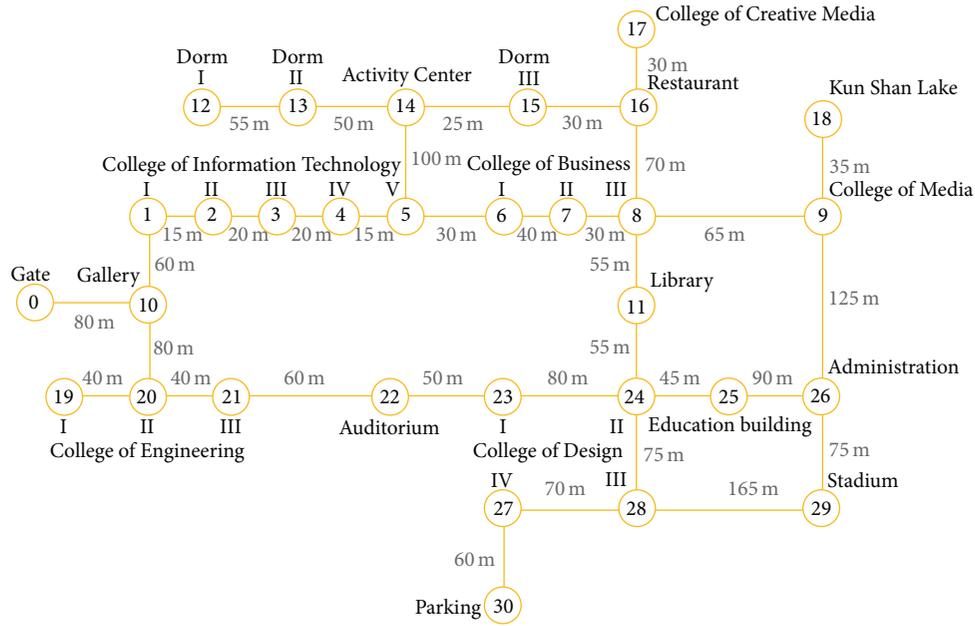


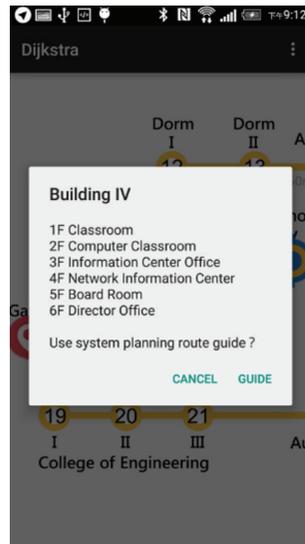
FIGURE 20: Open Bluetooth notification.

moving from Spot A to the target building (Spot B); the time spent in walking from Spot B to the destination sign located in the building at Spot C; and the time taken in moving from Spot C to the target classroom at number I4401.

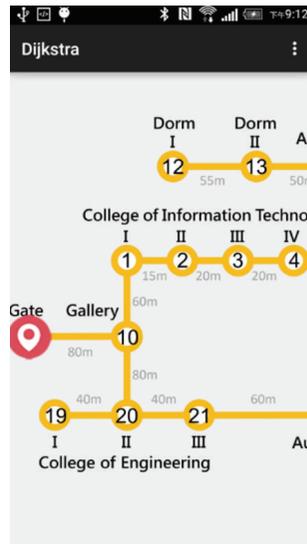
Tables 7(a) and 7(b) show the timing results obtained for the four users. Note that Users 1 and 2 performed the search



(a) Outdoor map overview



(b) Nearby building information



(c) Position of user

FIGURE 21: User located outside.

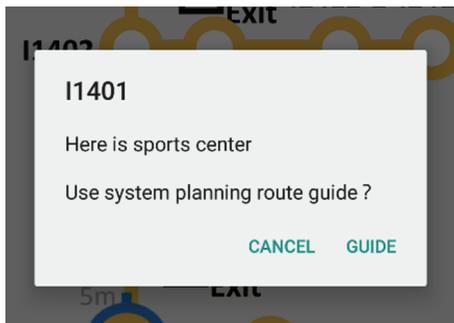


FIGURE 22: Guidance notification.

process in a nonassisted manner, while Users 3 and 4 both used the app. As shown, Users 1 and 2 completed the search process in 281.7 s and 333.9 s, respectively (i.e., an average search time of 307.8 s). Since Users 3 and 4 used the guidance app, they did not need to locate the direction signs at Spots A and C, respectively. Consequently, the total search times for the two users were just 206.48 s and 202.3 s, respectively (i.e., an average search time of 204.39 seconds). In other words, the mean time of the assisted users was 33.59% shorter than that of the two nonassisted users. Moreover, taking the search time of User 1 as a reference, the search times of Users 3 and 4 were reduced by 26.8% and 28.18%, respectively.

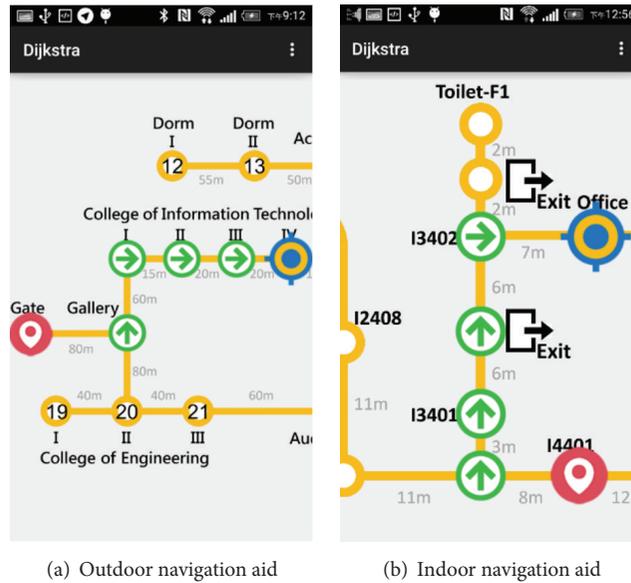


FIGURE 23: Navigation aid.

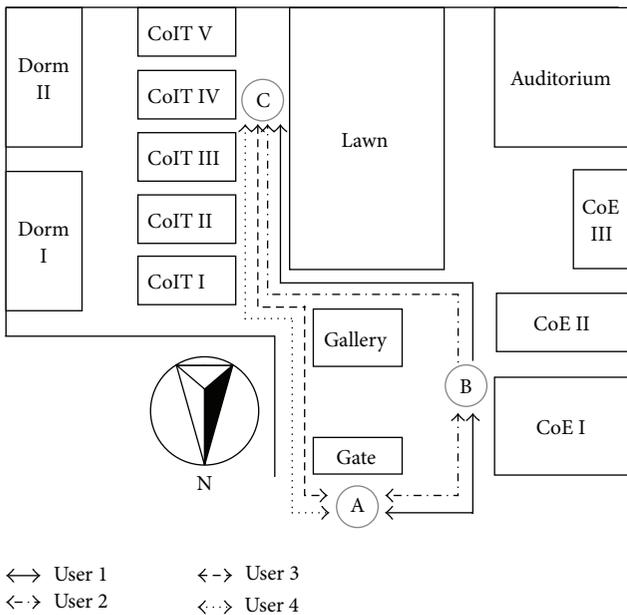


FIGURE 24: Experimental path (outdoor).

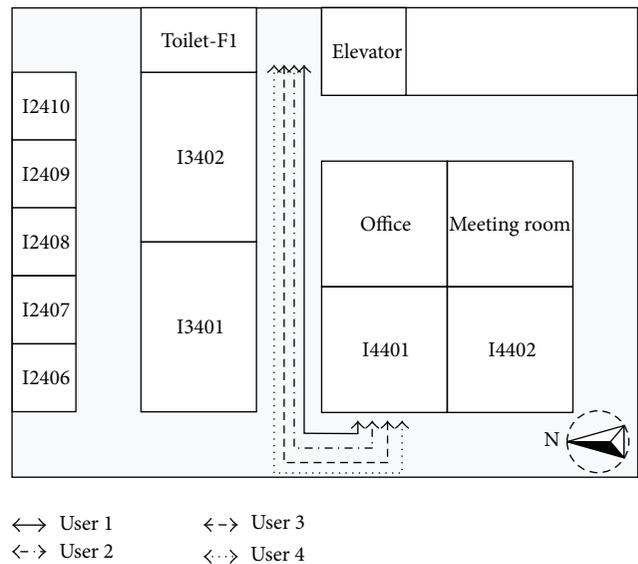


FIGURE 25: Experimental path (indoor).

**5. Conclusion**

With the emergence of Location-Based Services (LBS), the need to locate the position of the user with a high degree of accuracy has emerged as an important concern. Accordingly, this study has proposed an app based on GPS, Bluetooth beacon, and NFC technology for providing both a user localization service and an automatic guidance function. Importantly, the app functions in both outdoor and indoor environments and thus provides a seamless

localization/guidance function as the user moves from one environment to the other. The feasibility of the proposed system has been demonstrated by means of localization and guidance tests on a typical student campus building.

**Conflict of Interests**

The authors declare that there is no conflict of interests regarding the publication of this paper.

TABLE 7: Times spent by four users in completing each stage of search process.

(a) Search times of four users (unit: s)

Name	Spot A	Spot B	Spot C	Spot D
Subject 1	43.87	113.93	118.97	4.93
Subject 2	23.4	153.6	128	28
Subject 3	N/A	109.43	N/A	97.05
Subject 4	N/A	129.9	N/A	72.4

(b) Total search times of four users and average saved time (unit: s)

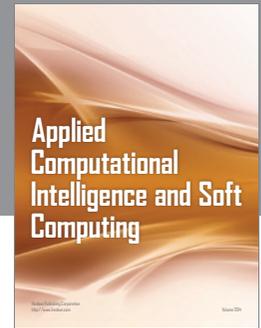
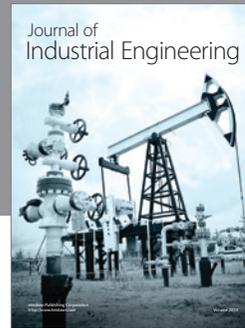
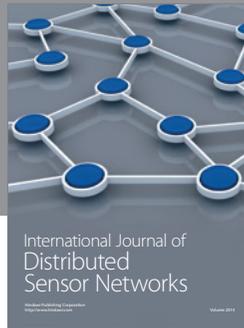
Name	Total	Percentage of time saving
Subject 1	281.7	0%
Subject 2	333.9	-18.53%
Subject 3	206.48	26.8%
Subject 4	202.3	28.18%

## Acknowledgment

The authors would like to thank the National Science Council, Taiwan, for the financial support of this study under Contract no. MOST 103-2627-E-168-001.

## References

- [1] S. Ray, R. Blanco, and A. K. Goel, "Supporting location-based services in a main-memory database," in *Proceedings of the 15th IEEE International Conference on Mobile Data Management (MDM '14)*, pp. 3–12, Brisbane, Australia, July 2014.
- [2] A. Chandra, S. Jain, and M. A. Qadeer, "GPS locator: an application for location tracking and sharing using GPS for Java enabled handhelds," in *Proceedings of the International Conference on Computational Intelligence and Communication Networks (CICN '11)*, pp. 406–410, IEEE, Gwalior, India, October 2011.
- [3] F. Liu and Z. Yang, "Study on applications of LBS based on electronic compass," in *Proceedings of the 5th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '09)*, pp. 1–4, Beijing, China, September 2009.
- [4] N. E. Klepeis, W. C. Nelson, W. R. Ott et al., "The National Human Activity Pattern Survey (NHAPS): a resource for assessing exposure to environmental pollutants," *Journal of Exposure Analysis and Environmental Epidemiology*, vol. 11, no. 3, pp. 231–252, 2001.
- [5] I. Yamada, T. Ohtsuki, T. Hisanaga, and L. Zheng, "An indoor position estimation method by maximum likelihood algorithm using RSS," in *Proceedings of the Annual Conference SICE*, pp. 2927–2930, IEEE, Takamatsu, Japan, 2007.
- [6] J.-S. Leu and H.-J. Tzeng, "Received signal strength fingerprint and footprint assisted indoor positioning based on ambient Wi-Fi signals," in *Proceedings of the IEEE 75th Vehicular Technology Conference (VTC Spring '12)*, pp. 1–5, IEEE, Yokohama, Japan, June 2012.
- [7] M. G. Moon and S. I. Choi, "Indoor position estimation using image sensor based on VLC," in *Proceedings of the International Conference on Advanced Technologies for Communications (ATC '14)*, pp. 11–14, Hanoi, Vietnam, October 2014.
- [8] E. Nakamori, D. Tsukuda, M. Fujimoto et al., "A new indoor position estimation method of RFID tags for continuous moving navigation systems," in *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN '12)*, pp. 1–8, Sydney, Australia, November 2012.
- [9] S. Khalifa and M. Hassan, "Evaluating mismatch probability of activity-based map matching in indoor positioning," in *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN '12)*, pp. 1–9, IEEE, Sydney, Australia, November 2012.
- [10] A. Baniukevic, D. Sabonis, C. S. Jensen, and H. Lu, "Improving Wi-Fi based indoor positioning using bluetooth add-ons," in *Proceedings of the 12th IEEE International Conference on Mobile Data Management (MDM '11)*, pp. 246–255, Lulea, Sweden, June 2011.
- [11] M. Ji, J. Kim, J. Jeon, and Y. Cho, "Analysis of positioning accuracy corresponding to the number of BLE beacons in indoor positioning system," in *Proceedings of the 17th International Conference on Advanced Communication Technology (ICACT '15)*, pp. 92–95, IEEE, Seoul, The Republic of Korea, July 2015.
- [12] Global Positioning System—Wikipedia, the free encyclopedia, September 2015, [https://en.wikipedia.org/wiki/Global\\_Positioning\\_System](https://en.wikipedia.org/wiki/Global_Positioning_System).
- [13] iBeacon, 2015, <https://en.wikipedia.org/wiki/iBeacon>.
- [14] Proximity Overview, September 2015, [https://docs.gimbal.com/proximity\\_overview.html](https://docs.gimbal.com/proximity_overview.html).
- [15] AltBeacon/spec·GitHub, September 2015, <https://github.com/AltBeacon/spec>.
- [16] "Near field communication—Wikipedia, the free encyclopedia," September 2015, [https://en.wikipedia.org/wiki/Near\\_field\\_communication](https://en.wikipedia.org/wiki/Near_field_communication).
- [17] D. Eppstein, "Finding the  $k$  shortest paths," *SIAM Journal on Computing*, vol. 28, no. 2, pp. 652–673, 1999.
- [18] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Section 24.3: Dijkstra's algorithm," in *Introduction to Algorithms*, pp. 595–601, MIT Press, McGraw-Hill, 2nd edition, 2001.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

