International Journal of Web Services Research
Volume 19 • Issue 1

Indoor Framework:
Data Structure, Navigation, Routing, and Trajectories in Indoor Spaces

Sultan Alamri, Saudi Electronic University, Saudi Arabia*

ABSTRACT

The tracking of spatial objects in indoor location-based services is becoming increasingly important for many applications. However, much research has focused only on querying and indexing in indoor spaces without considering the indoor variations. Therefore, this paper presents an indoor framework which includes data structures of indoor environments comprised of various building features and multiple floors. Moreover, the indoor framework includes indoor navigation and routing for both directed and undirected indoor environments, indoor density which takes into account the room capacity, and movement trajectories in single and multi-floor structures. Using synthetic data, the authors conducted extensive experiments to evaluate the proposed framework. The results show that this indoor framework can be implemented efficiently and effectively.

KEYWORDS

Indoor Routing, Indoor Spaces, Indoor Trajectories, Query Processing

INTRODUCTION

The spatiotemporal databases for indoor environments provide an important basis for a variety of applications with the availability of modern positioning devices such as global positioning systems (GPSes), Wi-Fi, and Bluetooth (Taniar & Rahayu, 2013; Alamri, 2018b, 2021). In indoor environments, the diversity of buildings and multi-floor environments has a huge impact on influencing the efficiency of the spatial-temporal database (Liu et al., 2020; Alamri, 2018a). Currently, people spend a lot of their time indoors— whether at work or home—particularly because more people are working from home nowadays. In people's daily lives, they encounter a variety of indoor environments such as airports, shopping malls, and colleges, where the monitoring and querying of spatial items has become critical (Dionti et al., 2017; Jin et al., 2016; Alamri et al., 2014). Furthermore, to execute a spatial data query efficiently in indoor environments, their characteristics must be taken into account because they are very different from those of outdoor environments.

This paper proposes an indoor framework that takes into account the features of indoor environments. The proposed framework introduces an undirected graph data structure representing multi-dimensional spaces covering a variety of indoor structures such as multi-floor and overlapping buildings. Moreover, the proposed framework includes indoor routing and navigation. A shortest path routing algorithm for single- or multi-floor environments is included based on the proposed data structure. Furthermore, because the nearest neighbor query is the most common spatial query, the paper proposes a nearest neighbor algorithm for indoor routing, which takes the distance factor into account.

DOI: 10.4018/IJWSR.314630
*Corresponding Author

In indoor spaces, there are many features and factors that can play a key role in query processing and routing (Alamri et al., 2020; Koike-Akino et al., 2020). Thus, this paper considers two examples of indoor factors that can affect indoor routing and navigation. In this framework, the author proposes an algorithm for shortest path routing in directed floors. In some buildings, the movement of objects across floors is controlled in the same way as in outdoor environments. In these cases, spatial objects must use a certain pathway to reach a specific room or venue; this pathway can affect the indoor routing. Another example is the indoor density factor that algorithms address for the shortest path with density. Lastly, the indoor framework considers the indoor trajectories construction in single- and multi-floor environments. The results of the experiments suggest that the proposed indoor framework is effective.

The rest of the paper is organized as follows. Section 2 presents the background, including the indoor positioning technologies, and discusses the distinctions between indoor and outdoor environments. Section 3 presents the proposed indoor framework, including indoor environment data structure; indoor environment querying and routing (consisting of shortest path routing, nearest neighbor routing, shortest path routing in directed floors, and shortest path with density); and indoor trajectories. Section 4 presents the conclusion and includes suggestions for future research.

RELATED WORKS AND BACKGROUND

The mobility of objects in spatial environments requires more frequent database updates. In outside environments, for instance, vehicles such as trucks, aircraft, and buses are moving at varying speeds (Alamri et al., 2013b; Rahman & Kim, 2012). A lot of research effort has gone into resolving the issue regarding the frequency with which moving objects are updated and addressing a wide range of spatial concerns in outdoor areas. For instance, the TPR-tree (Time Parameterized R-tree) uses the R*-tree definition to handle the data structure of spatial objects (Tao et al., 2003). Many others have adopted the definition of the TPR-tree (Tao et al., 2003; Alamri et al., 2013a). Moreover, other works focus on building a data structure for spatial objects to satisfy the variety of indoor queries (Alamri et al., 2020). The most popular spatial data structures are Euclidean space, spatial road network, and cellular space (Susanti et al., 2018). Cellular space refers to indoor spaces that contain space-related objects. In cellular spaces, queries about spatial objects are based primarily on cellular notations, such as “Which objects are in room 87?” The cell/room number is then the identifier of the target location (Dionti et al., 2017).

In Dionti et al., 2017, the researchers developed the prototype of an inter-building routing system that considers the external and internal pathways between buildings. This work explains that the A* algorithm shows optimal performance in retrieving the shortest path between rooms in inter-building structures. Other work has focused on the shortest path, taking into consideration the high-capacity cell environments in a directed indoor space (Alamri, 2018a). However, this work focused on only the hop distance method, where the distance between two cells is calculated based on the number of cells or rooms between them (Alamri, 2018a). Other research examined crowd awareness in indoor path planning queries (Liu et al., 2021) and proposed an indoor crowd awareness fastest path query and indoor least crowded path query. The COVID-19 pandemic and the impact of social distancing in indoor spaces and crowded venues inspired this framework for query processing. The first method returns a path with the quickest travel time in the presence of crowds, while the second returns a path with the fewest objects encountered along the way. Essentially, the researchers used a crowd model that organizes indoor topology and captures indoor partition flows and densities. For the indoor trajectories, (Alamri et al., 2021) uses a barometric sensor imbedded in the mobile device to reconstruct the trajectory in a multi-floor interior environment. Another work focused on determining the stay area in the trajectory of an indoor moving object (Zhou et al., 2021). The stay area is defined as a collection of low-speed, tightly spaced, and densely distributed position points. In this work, the authors proposed a stability value-based discovery technique in which any position locations fulfilling a stability value less than the stability threshold are referred to as stay points, and a set of them is referred to as the found stay region (Zhou et al., 2021).

The technology used to locate objects in indoor spaces is different from that used for outdoor spaces because indoor environments have various obstacles (Ruiz-López et al., 2010; Lassabe et al., 2009) generally not found outdoors. The GPS in indoor spaces is inadequate because of the barriers, walls, and partitions that have a substantial impact on the GPS reading's accuracy (Nessa et al., 2020; Luo et al., 2013). In addition, indoor spaces could contain multiple stores; hence, a GPS is not useful since it is based on two-dimensional coordinates (coordinate bases x or y). Furthermore, to locate the moving objects with acceptable accuracy in indoor spaces, many positioning systems need to be modified to enable them to detect moving objects efficiently in indoor spaces. The strategic placement of Wi-Fi has become popular for indoor environments. Wi-Fi has undergone a great deal of development and modifications in recent years (Shin et al., 2010). Furthermore, indoor Wi-Fi placement triangulation is based on the use of multiple connection points (nodes) fixed on the multi-floor structure (Forno et al., 2005; Alamri et al., 2020; Chen et al., 2021). Here, the nodes are close together. The three-dimensional space ID was hidden in this instance so that it could be used in a multistory structure. Furthermore, several indoor locating solutions based on the pattern of observations include Wi-Fi fingerprinting, which compares observations to previously marked sites, and trilateration, which employs a geometric formula to compute the distance from the transmitter to identify the system (Navizon, 2013; Cisco, 2013; Ishihara & Kawashima, 2020). Wi-Fi-based positioning systems are exploiting the fast expansion of wireless nodes (access points) in indoor and urban environments. Other technologies such as Bluetooth beacon, and camera positions, have also been used for indoor positioning (Wang et al., 2015). Furthermore, Bluetooth technology is used here to measure the distances between multiple locations indoors using geometric triangulation, which can provide high accuracy in spatial object positioning (Subedi et al., 2020).

INDOOR FRAMEWORK

This section begins with a description of the data structure of the indoor framework. The indoor routing, indoor navigation, indoor density, and movement trajectories are also explained.

Indoor Environment Data Structure

In this section, the data structure used as the basis for the indoor framework is described. An undirected graph structure is constructed using a three-dimensional space representation (mainly) of the data sets. Here the author represents multi-floor buildings in three dimensions based on the location of the coordinates x, y, and z coordinates of altitude. An example of a building is shown in Figure 1.

Figure 1. A building example layout

Definition 1. An indoor environment is a connected undirected graph (*Node, Connection*), where *Nodes* = {N_1, *N2,....*N_n} is a set of cells, and *connection = {C1,C2,. . ., Cn}* is a set of connections, each of which is a set of two nodes; that is each connection links two independent nodes.

Example: Figure 2 shows an indoor building and illustrates the initiating of the *x* and *y* coordinates of the nodes.

Figure 2. A building example with *x* and *y* coordinate of the nodes

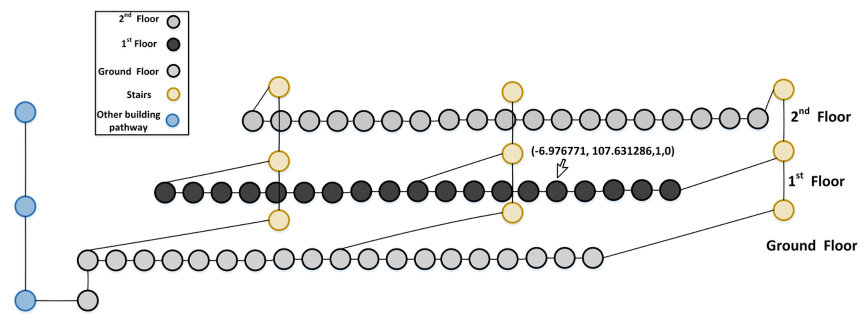
An indoor environment can comprise one or more than one building. This factor is one of the challenges of building data structures of indoor environments. Other aspects that should be considered are special elements such as doors, wings, rooms, levels, stairs, elevators, and pathways. Hence, mapping of the indoor environment is the first step to obtaining a logical model. Here, the logical model is an undirected graph with nodes and edges (also known as connections). The mappings from domain concepts to modeling concepts are summarized as follows. A room is mapped to a node, and a door is considered to be an edge connecting two nodes. A corridor or pathway (also known as a circulation area) is a linear structure consisting of connected nodes. If the pathway or the corridor is not too long, it can be considered as one node. Otherwise, it is divided into several nodes. Note that the stairs are treated similarly to the corridor or pathway; hence, they can be mapped to one or more cells. Furthermore, because the elevator has many difficulties, in this study, it is considered to be a single node. Table 1 explains the modeling concepts.

Table 1. Mapping the indoor environment to modeling concepts

Domain Concept	Modeling Concept
Room	A node
Door	A connection
Corridor/pathway	One or more nodes
Stair	One or more nodes

A 3D structure is shown in Figure 3. In this example, the building has three levels where each room is a node that is connected by an edge to a different node. Each node has coordinates (*x, y, z, h*). Here, *x, y* is the longitude coordinate in decimals, the *y* attribute is the latitude coordinate in decimals, and the *z* attribute is the floor level (height) or the level number of the building that is systematically coordinated and implemented in the system. The coordinate *h* is used for overlapping buildings, and 0 is used for a non-overlapping building. Here, the *x* and *y* coordinates use the shifting method whereby

Figure 3. A building's nodes illustrating the node connections in three different levels



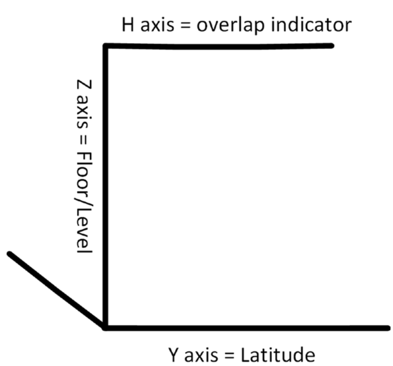
the coordinates of the nodes will not be overlayed. A model that consists of four-dimensional spaces can be seen in Figure 4. An example of a node is shown in Figure 3, where room 12 is indicated as a node the coordinates of which are (-6.976771, 107.631286,1,0), where 1 indicates the floor number of the coordinate z and 0 indicates H where there is no overlapping (Definition 4).

Definition 2. An indoor environment is a connected undirected graph where a (*Node*) has coordinates (*x, y, z, h*) where *x* is the longitude coordinate in decimals, the *y* attribute is the latitude coordinate in decimals, and *z* attribute is the floor number, and the *h* attribute is the overlap indicator for a complex building.

Definition 3. Given an indoor graph *G* and a node, N_i is connected to N_j if: N_i connected to

N_j through corridor or pathway (N_i ⊢" N $_j$).

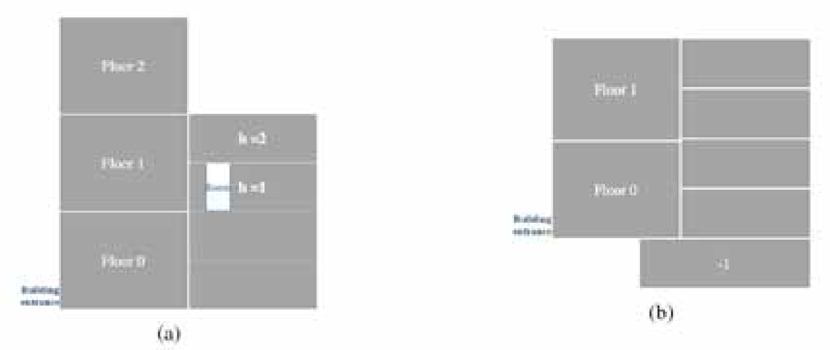
Figure 4. A 4D model of indoor space



Overlapping Building

In some cases, a building might be more complex than other regular buildings, and some levels might overlap. Figure 5 shows an example of a building where the east wing has a high level, and the west wing is lower, but the same height as the first floor in the east wing. Therefore, in this case, the coordinate *h* will play an important role in differentiating the room or the nodes. If a floor is overlapped in some part of the building, then the *h* attribute is different because of the overlapping as shown in Figure 5.

Figure 5. A building example where the floor overlapped in different wings



Definition 4. An indoor building consists of *Nodes* = $\{N_1, N2,....N_n\}$ is a set of cells, and *connection* = *{C1,C2,. . ., Cn}*, if the floor height in wing *A* is different from that in wing *B*, then: the coordinate *h* of *Ni* in *B* is different from *A*.

Definition 5. If an indoor building contains a basement, each node has the coordinates (*x, y, z, h*), the *z* attribute will be negative for the basement.

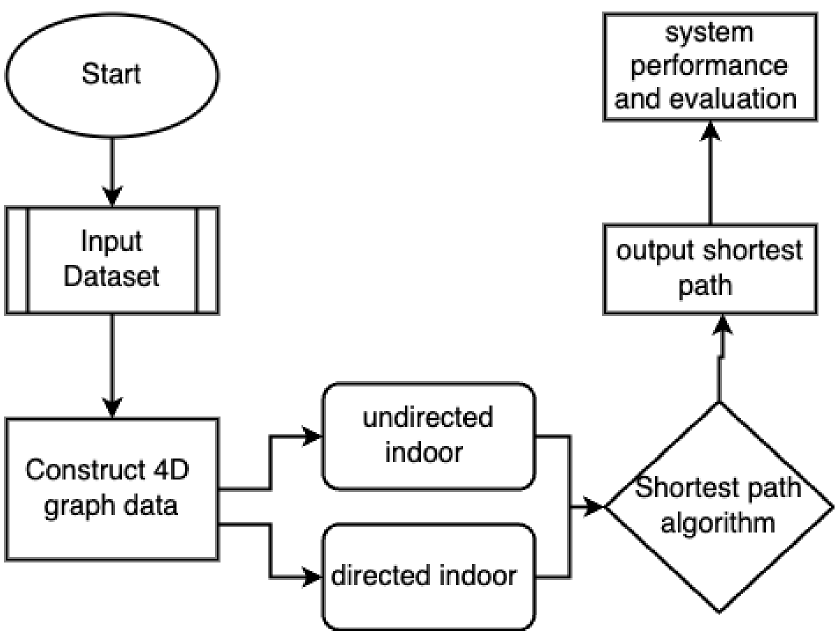
Indoor Environment Querying and Routing

Shortest Path Routing

The routing system method can be applied to a 4D structure. Consider the main process of indoor routing presented in Figure 6.

The query processing methodology will obtain the location of the starting point (room or node) and then the destination point. The shortest path routing approach then retrieves the shortest route

Figure 6. Indoor routing system main process



from those nodes (or points) using the shortest algorithm. In this paper this approach was applied to two types of shortest routes:

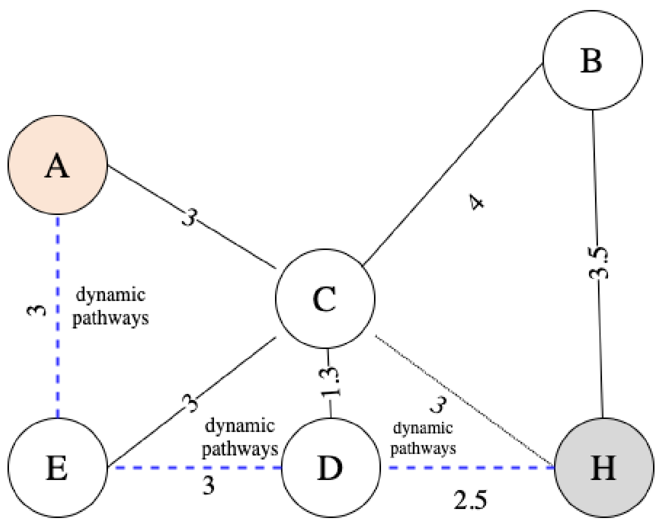
- The shortest route where all nodes in the graph are considered
- The closed shortest route that will eliminate all the open space nodes like the outdoor corridor that has no roof

Note that we adopted a modified A* algorithm because it is highly efficient. The algorithm begins to retrieve the shortest path to the user based on the shortest distance between node connections as explained previously, taking into account the preassigned dynamic pathways (i.e., pathways that are not crowded).

Definition 6. An indoor building that consists of *Nodes* = $\{N_1, N2,....N_n\}$ is a set of cells, and *Connection* = $\{C_1,C_2,..., C_n\}$, the optimal candidate route between, will be the shortest distance considering the dynamic pathway with $density_{value}$.

A small example of an undirected graph is shown in Figure 7. Assume that the input source node is *A*, that the input destination node is *H*, and that each cell has a regular cell value.

Figure 7. Small undirected graph example



Definition 7: Let *G* be a graph with *C* representing a set of indoor cells where $C = \{C_1,C_2,C_3...,C_n\}$. C_l $density_{value}$ at time (*T*) of the graph *G* are defined as: no $Distance_{PositiveValue}$ if C_l in regular density, add $Distance_{PositiveValue}$ if C_l in max density.

Here, the $density_{value}$ is preassigned and indicated by the admin. This varies based on the structure, the size of the interior, the number of cells, and the time. For instance, the density impact of the shortest path in Figure 9 can be determined as follows: between density of 1 to 15, it will be determined as a cell with regular density and no $Distance_{PositiveValue}$ will be applied. Between 16 to 30 persons, it will be taken into consideration as a maximum density-cell, and it should have a higher positive value for distance because they have a greater impact on the routing. Also, the dynamic pathway is preassigned and indicated by the admin. This pathway will be given a negative value (which means it will shorten the distance). Here the dynamic pathway is to be assumed that it is smoother than the movements between the cell of the moving objects; thus it is assigned with a negative value.

Figure 8. Shortest path example considering the cells densities

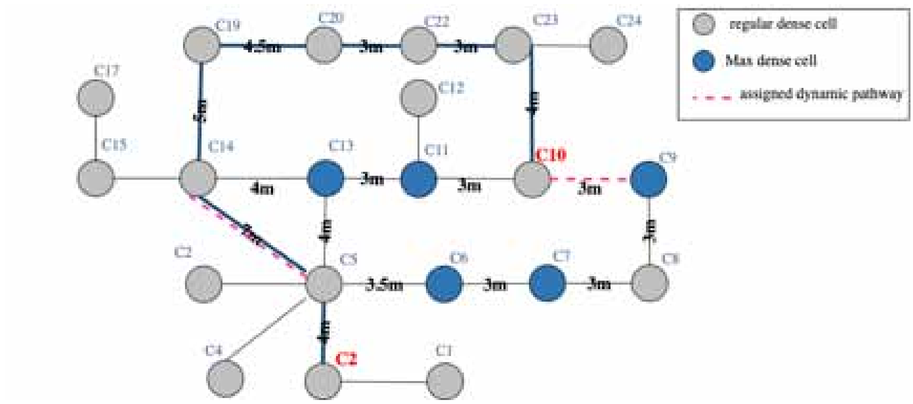
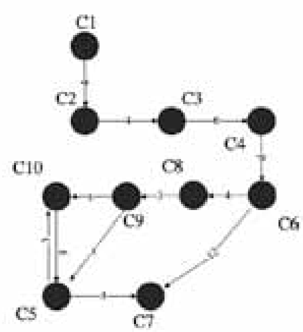


Figure 9. A directed floor with its node connections

$$
\begin{array}{c|cccccccccc}
 & C_1 & C_2 & C_3 & C_4 & C_5 & C_6 & C_7 & C_8 & C_9 & C_{10} \\
\hline
C_1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
C_2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
C_3 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
C_4 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
C_5 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
C_6 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
C_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
C_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
C_9 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
C_{10} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0
\end{array}
$$



The shortest path with density strategy can be used to route in a multistory structure. As a result, stairwells will be made up of one or more nodes/cells linked by one or more edges.

For example, in Figure 8 the possible routes between O_2 and $O_1$0 are four routes as follows: C_2-C_5- C_{13}-C_{11}-C_{10}, C_2-C_5-C_6-C_7-C_8-C_9-C_{10} and C_2-C_5-C_{14}-C_{19}-C_{20}-C_{22}-C_{23}-C_{10}, C_2-C_5-C_{14}-C_{13}-C_{11}-C_{10}. Al- though the first route is the shortest (distance point view) because of the max density cell (e.g., C_{13} and C_{11}), it is not the shortest based on the algorithms used in this paper. In this example, assume that the system admin set the $Distance_{PositiveValue}$ in the max density cell at (+7). Also, assume that the system admin set the *dynamic pathway* with (-2). Therefore, based on that, the $density_{value}$ for the first route is 30m. Therefore, based on the algorithm, the shortest bath is (C_2-C_5-C_{14}-C_{19}-C_{20}-C_{22}-C_{23}-C_{10}) since it has the lowest value.

Directed Floors Routing

The movement of an object on the floor in certain structures is directed in the same manner as it is in outdoor situations. Some interior buildings restrict object movement and orientation. Moving objects, for example, usually use a certain lane and path to get to a specific room. Visitors to some buildings, such as museums, must follow a predefined path and direction until they reach the museum's exit. Airport gates, some railway stations, and retail malls are examples of interior environments that have guided routes.

Definition 8: An indoor environment is a connected directed indoor $C = \{C_1, C_2, .., C_n\}$ where a set of cells (nodes) is regarded as a collection of edges (N_{e1}, N_{e2},N_{e4} ...N_{en}), where the movement of objects is restricted as

$$N_{e1} \overset{to}{\longrightarrow} N_{e2} N_{e2} \overset{to}{\longrightarrow} N_{e4} \ldots \overset{to}{\longrightarrow} N_{en}$$

When you are applying the shortest path algorithm to an indoor environment, it is essential to consider the adjacency matrix that may constrict the directed cells and how they connect to each other. The algorithms will produce the candidates' routes, taking into consideration the dynamic pathway and the density.

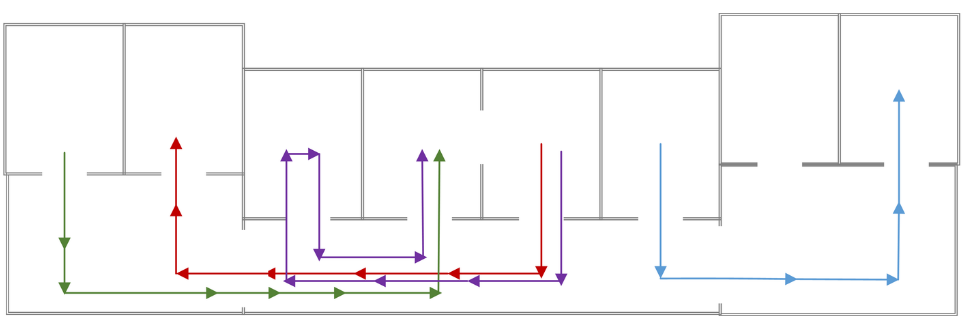
Definition 9. Let G be a directed graph with c cells and l line. The Directed Adjacency Matrix $DAM = [it]$ of the digraph G is an c x l (0, 1) matrix defined by:

$$DAM = \begin{cases} 1 & if \;\; i < - > t \\ 0 & if \;\; i > - < t \end{cases}$$

Reconstruction of Indoor Environments Trajectories

The technology of tracking is inevitably related to trajectory, which in turn depends on whether the environment is indoor or outdoor. Figure 10 shows the trajectory in an indoor environment. There are many advantages of knowing the trajectory in both environments: It allows the identification and analysis of patterns, the prediction of human trajectory in crowded indoor places, the identification of transportation routes, and a better understanding of the relationship between atmospheric transportation patterns and air quality in specific areas.

Figure 10. An example of the indoor trajectory



In this section, two algorithms are applied. The task of the first algorithm is to record the accelerometer and magnetometer sensor information obtained from a mobile device. The recorded numbers are accelerometer sensor measurements on the x, y, and z axes, compass angle readings, and the number of steps. The signal vector magnitude (SVM equation) is used to detect occurrences of running or stopping.

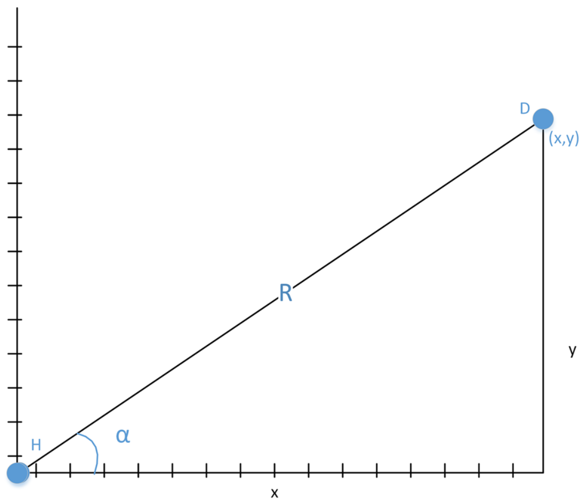
$$SVM = \sqrt{Acc_y^2 + Acc_y^2 + Acc_z^2}$$

Here, Acc_x, Acc_y, and Acc_z are the accelerometer sensor values of the axis. Also, SVM is the calculated acceleration value for the walking movement. Furthermore, the second algorithm is applied to reconstruct the trajectory. It begins with preprocessing data, where it takes the angle values and the number of recorded steps. Next, the cases are labeled: 1 for detecting walking and 0 if it stops. After labels are assigned, the rows of labels with values of 1 are matched, as well as the angle values. If the label on a row has a value of 1, then the new variable is assigned to the same row and saved. The new variable contains a set of angles that are used for reconstruction using trigonometry.

$\sin(90\text{-}\alpha) = y \Leftrightarrow y = r * \sin(90\text{-}\alpha)$
$\cos(90\text{-}\alpha) = x \Leftrightarrow x = r * \cos(90\text{-}\alpha)$

In Figure 11, the start point is *H* where the moving object with the angle of α takes a step. To reach the location of point *D*, it has to take the values of the *x* axis and the *y* axis at point *D*. Here, it calculates the values of the *x* and *y* axes, then the basic functions of trigonometry are used.

Figure 11. An example of people's movement



However, the reconstruction results did not match the actual route. Therefore, an initial trial was conducted to calculate the angle to obtain a reconstruction that optimally represented the real path/route cases. Here, two angle calculations were used: 90-α, or 90-α.

$\sin(\text{-}270\text{-}\alpha) = y \Leftrightarrow y = r * \sin(\text{-}270\text{-})$ $\cos(\text{-}270\text{-}\alpha) = x \Leftrightarrow x = r * \cos(\text{-}270\text{-})$

Results and Discussion

This section presents the experimental data used to evaluate the proposed framework. The simulation was run on a 64-bit Windows 10 Enterprise PC with an Intel Core i7-9750H CPU running at 2.60GHz with 16 GB of RAM. Furthermore, the standard Java platform is NetBeans IDE 8.1 in combination with JDK 8.

Real indoor spaces were applied in this experiment; namely, Telkom University, Buildings D, E, and F. As known, obtaining data for moving objects is difficult; therefore, the cell densities were generated synthetically based on genuine data. It is worth noting that the processes were repeated seven times before the average was calculated.

First, the construction cost of the proposed data structure was measured. The data structure of the indoor environment had an undirected 4D graph where each node has a coordinate (*x, y, z, h*). After the number of moving objects at different random time were considered, it became clear that the proposed framework data structure has an efficient cost as shown in Figures 12 and 13. Note that with an increase in the number of moving objects, the construction cost increases slightly; however, the testing was con ducted using a random number of moving objects at different times. Figures 12 and 13 show the results construction costs of buildings D and E.

Figure 12. The construction cost of the proposed data structure of building D

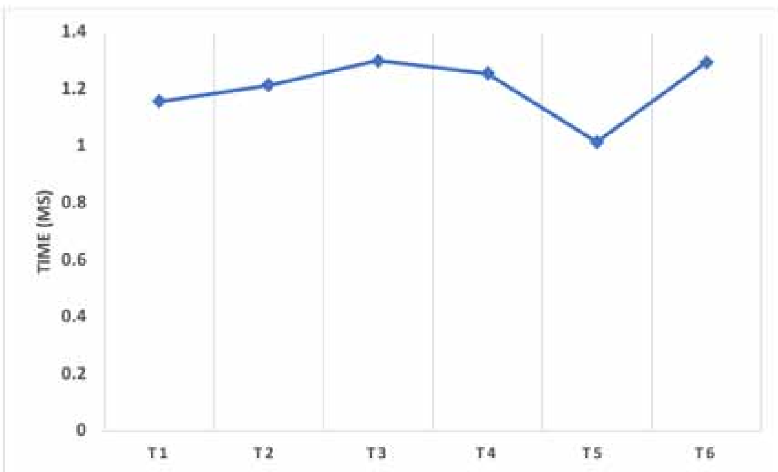


Figure 13. The construction cost of the proposed data structure of building E

Here, the results obtained by the shortest path algorithm are explained. First, for the testing sample, the shortest path between two nodes was chosen for two nodes on the same floor and two nodes on different floors, as well as two nodes on different floors and two nodes on different wings. These different situations were chosen to ensure the effectiveness of the algorithm in different scenarios. Figures 14, 15, and 16 show that the proposed shortest path algorithms always require less execution time than the Dijkstraâs algorithm (four queries in each case). Moreover, the proposed system shows a high accuracy between the distance compared with the real distance. Note that in Figures 14, 15, and 16 the impact of the cells/densities and the dynamic pathways were not tested.

Figure 14. Two nodes in the same floor

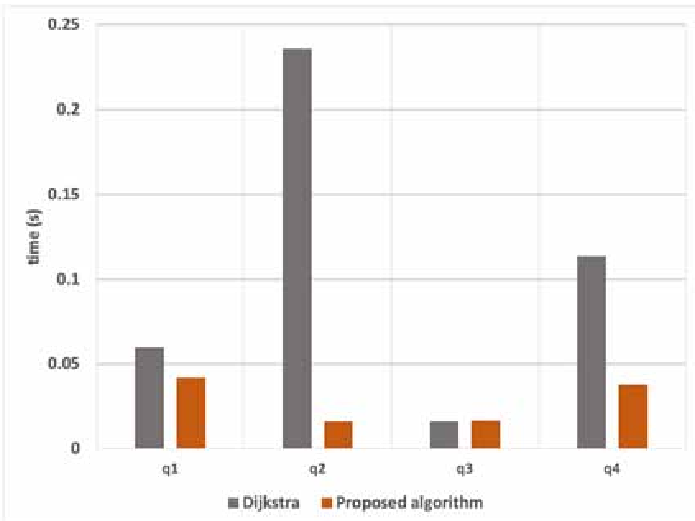


Figure 15. Two nodes in different floors

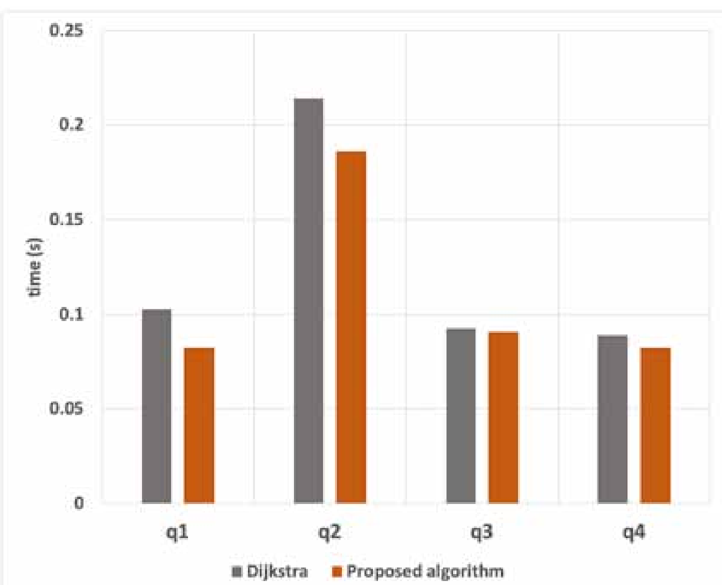


Figure 16. Two nodes in the different floors and different wings

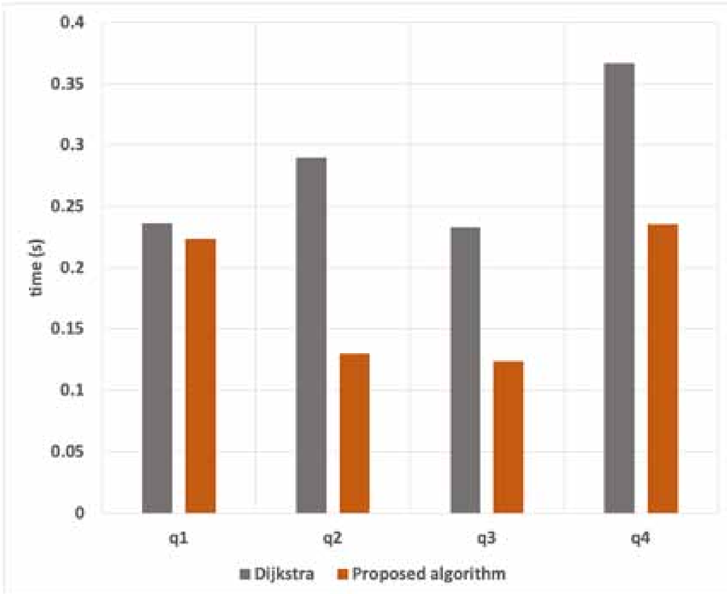
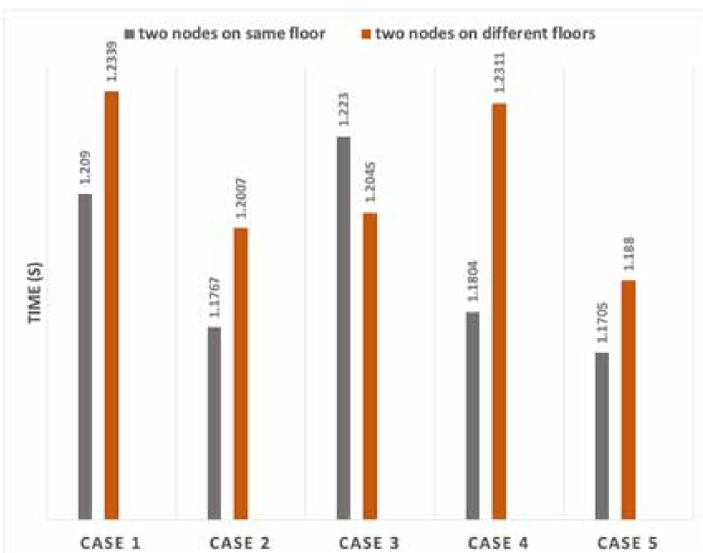


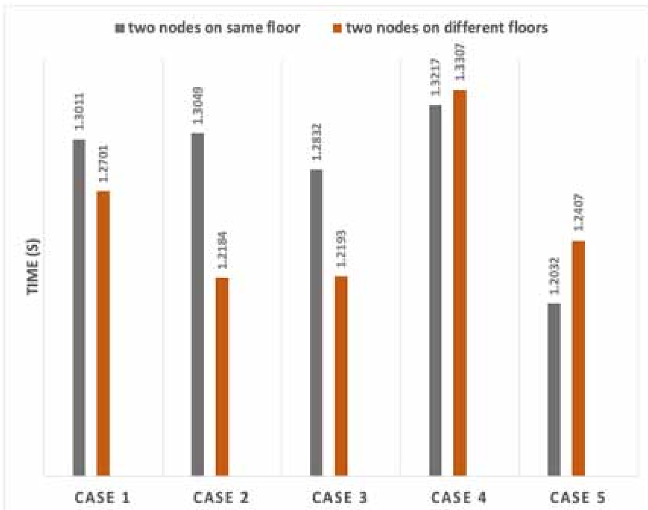
Figure 17 demonstrates the impact of the cells/densities and the dynamic pathways on five routing situations on the same floor and on other floors. This indicates that the majority of cells has regular density (i.e., up to 15 objects were employed in the studies). Although the distances increase, there is no noticeable influence, and the response time remains effective. Figure 18 shows a comparison of five routing scenarios with maximum density on the same level and on separate floors. Here, each case includes high-density cells with some dynamic paths (i.e., more than 13 moving objects, which

Figure 17. Five routing cases with regular density



was used in the experiments). It is worth noting, however, that in some situations, the response time is longer than in cases with regular density. Although the maximum density instances have a little longer response time in some cases, the performance is still efficient, and the density has no influence on the performance of the proposed algorithm. Note also that in the case of directed floors, efficient performance is predicted because the algorithms will perform similarly with the assistance of the adjacency matrix.

Figure 18. Five routing cases with maximum density



To evaluate the reconstruction of indoor trajectories algorithms, a test area for walking, which was laid out as a Cartesian graph on the floor, was used. The numbers on the *x* and *y* axes indicate every step. Note that in this testing, the average length of one step is 40 cm based on a person with a height of 160 cm. The four alternative paths used for evaluations are shown in Figure 19. These pathways are all on the same level. The main difference between these paths is the walking direction. Figure 19 (a) shows the northerly walking direction, Figure 19 (b) shows the easterly direction, and Figure 19 (c) shows the diagonal direction. Figure 19 (d) is a path that has several possible directions. Figure 20 shows a comparison between the real path and the reconstruction.

For the evaluation scenario, the influence of threshold and the changes in the position of the device were tested. Note that the level of accuracy (value) obtained is based on recognizable steps. Furthermore, there are eight actual steps in this evaluation. Figure 21 shows the accuracy of the number of recognizable steps that have the same pattern in each path. Note that threshold 2 has the highest value. Moreover, when the position is at an angle of 30 degrees, this indicates higher steps than the angles of 0 and 60 degrees.

In multi-floor environments, a threshold value ranging from 0.1 to 0.4 was used. Figure 22 shows the level of accuracy for path b to illustrate the number of steps recognized when SVM exceeds the threshold value. As can be seen from the experiment results, the optimal threshold value for identifying a human step is 0.3, with 0° for the device position.

CONCLUSION

This paper introduced a framework addressing various features of indoor spaces. An undirected graph data structure representing multidimensional spaces covering the variety of features of indoor spaces

Figure 19. Examples of four different paths used for testing

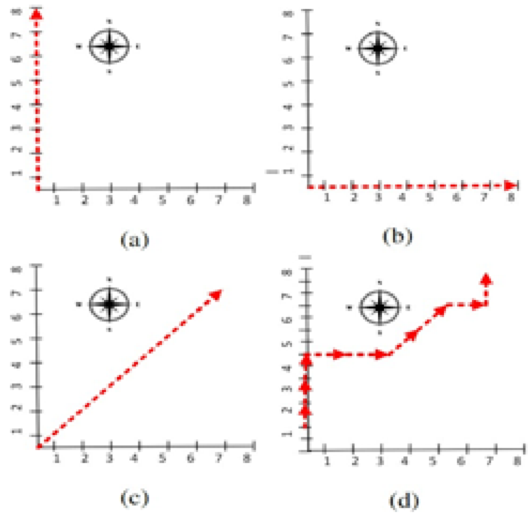


Figure 20. A comparison between the real path and the reconstruction path

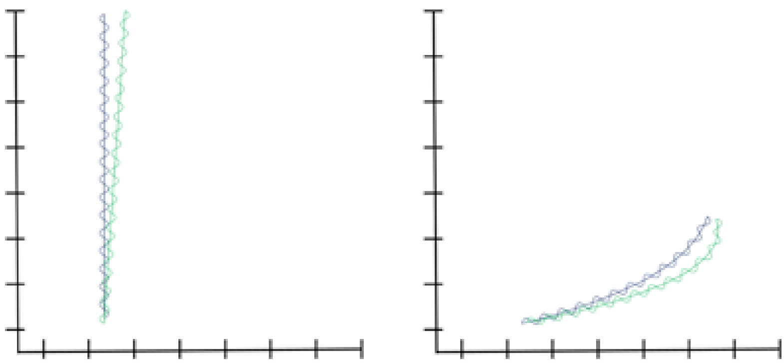


Figure 21. Accuracy level on real steps

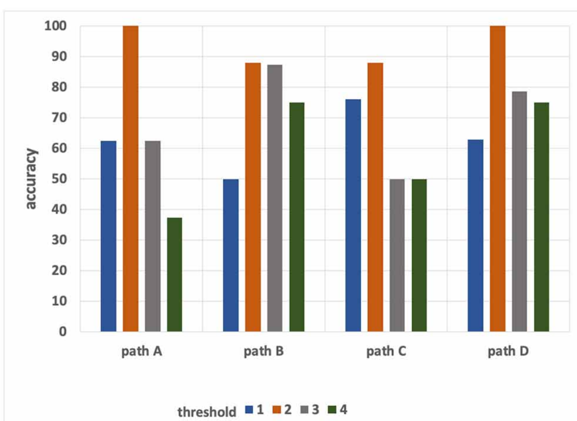
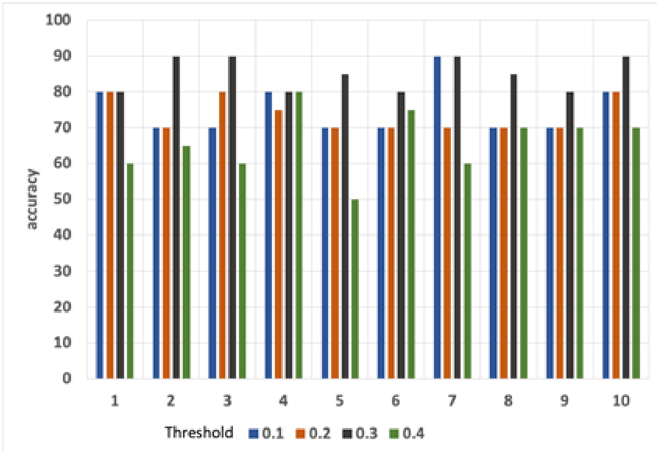


Figure 22. The level of accuracy for path b



such as multi-floor, overlapped buildings has been proposed. Furthermore, the proposed framework includes the indoor routing and navigation. A shortest path routing algorithm for a single or multi-floor environment is included based on the data structure. Because the nearest neighbor query is also the most common spatial query, in this paper, a nearest neighbor algorithm for indoor is considered. In both of these algorithms, the distance factor was taken into consideration. In addition, although many factors can play a key role in the query processing and routing, this paper considers only two important factors that can affect indoor routing and navigation: cell density and dynamic pathways. Moreover, an algorithm for routing in directed floors is considered in this framework. Lastly, the indoor framework considers the construction of indoor trajectories in single- and multi-floor environments. The results show that the proposed indoor framework is efficient.

This paper opens up new avenues for future research. For example, a moving object query in an indoor space could be applied to the framework and examined. Moreover, new factors associated with the index structure of indoor spaces, such as dynamic rooms and static rooms, could be investigated as well as how these factors affect query processing. In addition, indoor social distancing routing could be considered as aspects of the framework, and considering different spatial queries, such as room-range spatial queries, would be an interesting direction for future work.

FUNDING STATEMENT

The authors received no specific funding for this study.

DATA AVAILABILITY

The datasets used and analyzed during the current study are available at https://mega.nz/folder/YwQRTRCK#4g0Ew8GJMpn0GDZ9gEEG1g

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

Alamri, S. (2018a). An efficient shortest path routing algorithm for directed indoor environments. *ISPRS International Journal of Geo-Information*, *7*(4), 133. doi:10.3390/ijgi7040133

Alamri, S. (2018b). Spatial data managements in indoor environments: Current trends, limitations and future challenges. *International Journal of Web Information Systems*, *14*(4), 402–422. doi:10.1108/IJWIS-05-2018-0039

Alamri, S. (2021). Independent map enhancement for a spatial road network: Fundamental applications and opportunities. *ISPRS International Journal of Geo-Information*, *10*(1), 8. doi:10.3390/ijgi10010008

Alamri, S., Nurfalah, K., & Adhinugraha, K. (2021). Multi-floor indoor trajectory reconstruction using mobile devices. *Computer Modeling in Engineering & Sciences*, *128*(3), 927–948. doi:10.32604/cmes.2021.014852

Alamri, S., Taniar, D., Nguyen, K., & Alamri, A. (2020). C-tree: Efficient cell-based indexing of indoor mobile objects. *Journal of Ambient Intelligence and Humanized Computing*, *11*(7), 2841–2857. doi:10.1007/s12652-019-01397-w

Alamri, S., Taniar, D., & Safar, M. (2013a). Indexing moving objects for directions and velocities queries. *Information Systems Frontiers*, *15*(2), 235–248. doi:10.1007/s10796-012-9367-8

Alamri, S., Taniar, D., Safar, M., & Al-Khalidi, H. (2013b). Spatiotemporal indexing for moving objects in an indoor cellular space. *Neurocomputing*, *122*, 70–78. doi:10.1016/j.neucom.2013.03.035

Alamri, S., Taniar, D., Safar, M., & Al-Khalidi, H. (2014). A connectivity index for moving objects in an indoor cellular space. *Personal and Ubiquitous Computing*, *18*(2), 287–301. doi:10.1007/s00779-013-0645-3

Chen, M., Liu, K., Ma, J., Gu, Y., Dong, Z., & Liu, C. (2021). SWIM: Speed-Aware WiFi-Based Passive Indoor Localization for Mobile Ship Environment. *IEEE Transactions on Mobile Computing*, *20*(2), 765–779. doi:10.1109/TMC.2019.2947667

Cisco. (2013). *Location Tracking Approaches*. Author.

Dionti, T. A., Adhinugraha, K. M., & Alamri, S. M. (2017). Inter-building Routing Approach for Indoor Environment. In Computational Science and Its Applications — ICCSA 2017 — 17th International Conference, Trieste, Italy, July 3–6, 2017, Proceedings, Part I (Vol. 10404, pp. 247–260). Springer. doi:10.1007/978-3-319-62392-4_18

Forno, F., Malnati, G., & Portelli, G. (2005). Design and implementation of a Bluetooth ad hoc network for indoor positioning. *IEE Proceedings. Software*, *152*(5), 223–228. doi:10.1049/ip-sen:20045027

Ishihara, M., & Kawashima, R. (2020). Multi-distance function trilateration over k-NN fingerprinting for indoor positioning and its evaluation. *IEICE Transactions on Information and Systems*, *E103-D*(5), 1055–1066. doi:10.1587/transinf.2019EDP7241

Jin, P., Cui, T., Wang, Q., & Jensen, C. S. (2016). Effective Similarity Search on Indoor Moving-Object Trajectories. In *Database Systems for Advanced Applications — 21st International Conference, DASFAA, Dallas, TX, USA, April 16–19, 2016*, *Proceedings, Part II* (*Vol. 9643*, pp. 181–197). Springer. doi:10.1007/978-3-319-32049-6_12

Koike-Akino, T., Wang, P., Pajovic, M., Sun, H., & Orlik, P. V. (2020). Fingerprinting-based indoor localization with commercial MMWave WiFi: A deep learning approach. *IEEE Access: Practical Innovations, Open Solutions*, *8*, 84879–84892. doi:10.1109/ACCESS.2020.2991129

Lassabe, F., Canalda, P., Chatonnay, P., & Spies, F. (2009). Indoor Wi-Fi positioning: Techniques and systems. *Annals of Telecommunications*, *64*(9–10), 651–664. doi:10.1007/s12243-009-0122-1

Liu, T., Feng, Z., Li, H., Lu, H., Cheema, M. A., Cheng, H., & Xu, J. (2020). Shortest Path Queries for Indoor Venues with Temporal Variations. In *36th IEEE International Conference on Data Engineering, (ICDE)* (pp. 2014–2017). IEEE. doi:10.1109/ICDE48307.2020.00227

Liu, T., Li, H., Lu, H., Cheema, M. A., & Shou, L. (2021). Towards Crowd-aware Indoor Path Planning. *Proceedings of the VLDB Endowment International Conference on Very Large Data Bases*, *14*(8), 1365–1377. doi:10.14778/3457390.3457401

Luo, Y., Hoeber, O., & Chen, Y. (2013). Enhancing Wi-Fi fingerprinting for indoor positioning using human-centric collaborative feedback. *Human-centric Computing and Information Sciences*, *3*(1), 1–23. doi:10.1186/2192-1962-3-2

Navizon. (2013). *Indoor GPS and indoor tracking solutions by Navizon*. Author.

Nessa, A., Adhikari, B., Hussain, F., & Fernando, X. N. (2020). A survey of machine learning for indoor positioning. *IEEE Access: Practical Innovations, Open Solutions*, *8*, 214945–214965. doi:10.1109/ACCESS.2020.3039271

Rahman, Y. P. M. S., & Kim, K. D. (2012). RSS-based indoor localization algorithm for wireless sensor network using generalized regression neural network. *Arabian Journal for Science and Engineering*, *37*(4), 1043–1053. doi:10.1007/s13369-012-0218-1

Ruiz-López, T., Garrido, J., Benghazi, K., & Chung, L. (2010). A Survey on Indoor Positioning Systems: Foreseeing a Quality Design. In Distributed Computing and Artificial Intelligence. Advances in Intelligent and Soft Computing (Vol. 79). Springer. doi:10.1007/978-3-642-14883-5_48

Shin, B.-J., Lee, K.-W., Choi, S.-H., Kim, J.-Y., Lee, W. J., & Kim, H. S. (2010). Indoor WiFi positioning system for Android-based smartphone. *2010 International Conference on Information and Communication Technology Convergence (ICTC)*, 319–320. doi:10.1109/ICTC.2010.5674691

Subedi, S., & Pyun, J. Y. (2020). A survey of smartphone-based indoor positioning system using RF-based wireless technologies. *Sensors (Basel)*, *20*(24), 7230. doi:10.3390/s20247230 PMID:33348701

Susanti, R. M., Adhinugraha, K. M., Alamri, S., Barolli, L., & Taniar, D. (2018). Indoor Trajectory Reconstruction Using Mobile Devices. In *32nd IEEE International Conference on Advanced Information Networking and Applications (AINA)* (pp. 550–555). IEEE Computer Society. doi:10.1109/AINA.2018.00086

Taniar, D., & Rahayu, W. (2013). A taxonomy for nearest neighbour queries in spatial databases. *Journal of Computer and System Sciences*, *79*(7), 1017–1039. doi:10.1016/j.jcss.2013.01.017

Tao, Y., Papadias, D., & Sun, J. (2003). The TPR*-Tree: An Optimized Spatio-Temporal Access Method for Predictive Queries. In *Proceedings of the 2003 VLDB Conference* (pp. 790–801). Morgan Kaufmann. doi:10.1016/B978-012722442-8/50075-6

Wang, Y., Ye, Q., Cheng, J., & Wang, L. (2015). RSSI-Based Bluetooth Indoor Localization. In *11th International Conference on Mobile Ad-hoc and Sensor Networks (MSN)* (pp. 165–171). IEEE.

Zhou, Y., Chen, Y., & Pi, D. (2021). Discovery of stay area in indoor trajectories of moving objects. Expert Systems with Applications, 170, 114501.

Sultan Alamri is a senior member IEEE/professor of computer science. He received a master's degree in information technology from the School of Engineering and Mathematical Sciences, La Trobe University, Australia, in 2010, and a Ph.D. degree from the Faculty of Information Technology, Monash University, Australia, in 2014. He is currently a professor at the College of Computing and Informatics, Saudi Electronic University, Saudi Arabia. His research interests include data engineering and management, machine learning, indoor data management, computational geometry, moving objects, spatial databases, geospatial maps, and GIS.