A Mobile App-based Indoor Mobility Detection Approach using Bluetooth Signal Strength

Muztaba Fuad
Department of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
https://orcid.org/0000-0002-1942-1193

Anthony Smith
Department of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
asmith219@rams.wssu.edu

Debzani Deb
Department of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
https://orcid.org/0000-0002-3430-1130

Abstract— **In an indoor space, determining a person's mobility patterns has research significance and applicability in real-world scenarios. When mobility patterns are determined, layout optimization can be implemented in indoor spaces to improve efficiency. This research aimed to determine a person's path using Received Signal Strength Indicator (RSSI) data collected from Bluetooth-enabled mobile devices. Mobile app-based mobility detection using Bluetooth RSSI has the advantage of low cost and easy implementation. The research methodology involves developing a Bluetooth RSSI mobility application system to determine the path of a moving mobile device using a vectorized algorithm. The paper presents challenges in creating such a software system, its architecture, the data collection and analysis process, and the results of mobility detection. This research shows that Bluetooth-enabled mobile devices and Bluetooth RSSI data can be used to determine the path in an indoor space with workable accuracy.**

Keywords*— *RSSI, Path forming, Indoor localization, Mobility pattern.

I. Introduction

Indoor localization is the process of detecting the real-time location using wireless devices in an indoor environment with a bounded error rate [1]. Determining a person's localization and mobility patterns has research significance and applicability in real-world scenarios. With recent technological advances, the use of mobile devices, smartphones, and the Internet of Things (IoT) has exploded. Such uses require the need for indoor localization to give users personalized services and contextualized information. In recent years, indoor localization and tracking have increased in different fields, such as healthcare, retail, and facility management. To determine the localization and mobility patterns in an indoor environment can be compared to the Global Positioning System (GPS) that is commonly used for localization in an outdoor environment. GPS requires line-of-sight between the satellites and the handset. GPS cannot be used to determine indoor environment localization because GPS signals are too weak for indoors, and there is no line-of-sight contact with the GPS satellites.

A. Research Problem

Localization in an indoor environment can be difficult and presents challenges [1] with positioning. There are things that could impact accurate localization in an indoor environment, such as signal strength, device positioning, obstacles, height, etc. [2]-[3]. Unlike outdoor localization, indoor localization must use other positioning methods and hardware. Methods could include using Wi-Fi or Bluetooth signal strengths. Hardware could include mobile devices or beacons. Compared to Wi-Fi, Bluetooth signal strength is intended to enable short-range wireless communications between devices with reduced power requirements.

This research uses Bluetooth-enabled mobile devices to collect Bluetooth RSSI data and use that to detect changes in position over time. The reason to use Bluetooth RSSI instead of Wi-Fi RSSI is due to the energy requirements of Bluetooth and its ability to sense an individual device's signal strength directly instead of sensing it through a router or using an extra networking layer, such as Wi-Fi Direct. The use of dedicated hardware, such as beacons [4], can improve accuracy. However, it comes with added cost and logistical challenges [4]. Although indoor environment positioning and localization are inherently difficult [5], creating real-time mobility patterns using position data adds another set of challenges. Challenges faced in this research include synchronization among collection devices, real-time data access, and processing, creating real-time visualization, and the lack of techniques for providing such real-time mobility detection using only software and no dedicated hardware. Designing such mobile software has added challenges because of the asynchronous nature of its software stack. Appropriate synchronization methodologies must be used to satisfy such requirements, adding extra complexity to the development of such software. The presented approach can help to solve this problem without any specialized hardware and with the help of a traditional mobile application that can gather raw Bluetooth RSSI data between a swarm of devices and then process it to create mobility paths. A stand-alone mobile app, as presented in this paper, makes it easier to collect raw RSSI data from a swarm of devices and process it on the fly to find a mobility path without any need for specialized hardware or knowledge of the indoor space with quick and easy deployment.

B. Motivation

Companies continuously search for ways to be profitable and operate at the lowest cost possible. One of the main ways to cut

This research is funded by NSF grant number 2131100.

costs is to find efficiency gains. When a company improves efficiency, the operating costs will be lowered, improving profits. The healthcare industry wants to operate at a lower cost, improve efficiency, and create a better patient experience. Several healthcare facilities in an anonymous state are partnered with universities. The Community Care Center (CCC) facility in Winston-Salem is partnered with Winston-Salem State University. The CCC is a pro bono facility that provides various medical services to patients. Being a pro bono facility, efficiency is important due to costs, time with patients, and patient care. The CCC is a part of this study to determine mobility within indoor spaces and use the mobility information to place furniture and instruments in the most well-organized way to improve clinicians' workability and efficiency.

II. Related Work

Recent indoor localization approaches have promising results in pinpointing locations using Bluetooth. However, some approaches use low-powered beacons and the knowledge (such as size) of the indoor space [6]-[7]. Similar research in healthcare is increasingly using similar mobile crowdsensing, such as in clinic and psychological trials [8]-[11], public health [12], and personal well-being [13]. There are currently several techniques [14]-[15] for indoor localization, and extensive research [17]-[19] has been conducted in this area. Although these techniques each have some strengths and weaknesses, the presented approach has different goals and methodologies. The application domain of this research requires easy application of the technology without sophisticated hardware, the need for precious accuracy, or efficient energy usage. Since mobility detection is the goal, the approach does not need high precision but rather a general mobility pattern. Additionally, the use of stationary devices connected to power gives this approach the flexibility needed to use the Bluetooth proximity detection [20] technique in a real-time fashion, bundled in one app, which is easy and quicker to deploy.

Priwgharm et al. [21] explored a comparative study on indoor localization based on RSSI measurement in a wireless sensor network. To calculate the location of the target sensor, two main techniques were used, which included range-based and fingerprinting-based techniques. The range-based technique uses min-max and lateration for indoor localization, while the fingerprinting-based technique uses the nearest neighbor algorithm and average k-nearest neighbor algorithm as pattern-matching methods. For the range-based technique, the experiment results show that the lateration provides better location estimation accuracy than min-max. The results show that the nearest neighbor algorithm provides the best results for the fingerprinting-based technique. Between the two techniques, the lateration of the range-based gave the best result of location estimation. Although this research shares similar approaches, it doesn't use dedicated sensor networks and is based on using only mobile devices.

Wang et al. [3] focused on RSSI-based Bluetooth indoor localization. Two BLE-based localization schemes are presented: Low-precision Indoor Localization (LIL) and High-precision Indoor Localization (HIL). The LIL and HIL use the collected RSSI measurements to generate a small region where Bluetooth-enabled beacons are planted at a pre-determined area. Although this research shares similar goals, the presented research only uses mobile devices and applications in them instead of dedicated beacons, and sensing and path forming are all packaged in one mobile app. This makes the presented approach more cost-effective and easier to use.

III. System Overview and Working

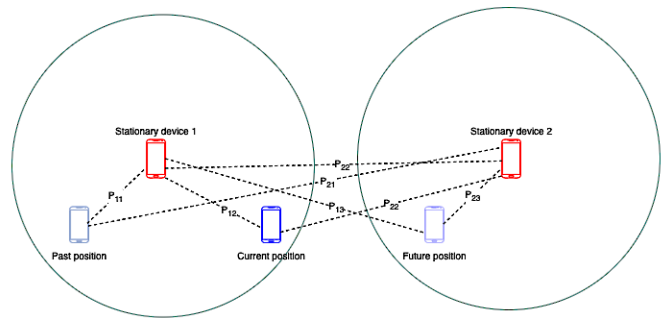

Fig. 1. Device organization and change of signal strength.

Fig. 1 shows the overall idea of this approach. Assume that there are two stationary devices (in red) in two rooms of the building and a mobile device (in blue) is moving between the rooms. Each stationary device can sense other Bluetooth-enabled devices within a radius of signal strength or proximity. These are devices (like a tablet, stationary in one place) placed strategically (like next to high-value/high-use instruments, close to treatment rooms, near high-traffic areas, etc.) around the facility. As the mobile device moves, each stationary device will sense its signal strength (P_{st} where s = stationary device and t = time stamp) and record that over time. So, for the given situation, the collected data for each stationary device will look like the following:

T_{past}	P_{1past}	P_{2past}
$T_{current}$	$P_{1current}$	$P_{2current}$
T_{future}	$P_{1future}$	$P_{2future}$

Once such data is collected over time, it can be used to determine the direction of gradual change of signal strength, and the mobility path can be determined from such directional vectors. The app, developed as part of this research, runs on all stationary and mobile devices and has three different roles: broadcaster, collector, and aggregator. The broadcaster role is assigned to mobile devices, and in that role, it keeps the device's Bluetooth activated so that other devices can sense it. The stationary devices can have the other two roles. As a collector, the app collects RSSI data about mobile devices. As an aggregator, it collects data from other collector nodes and aggregates them for further processing and producing the mobility path. The aggregator app utilizes the devices'

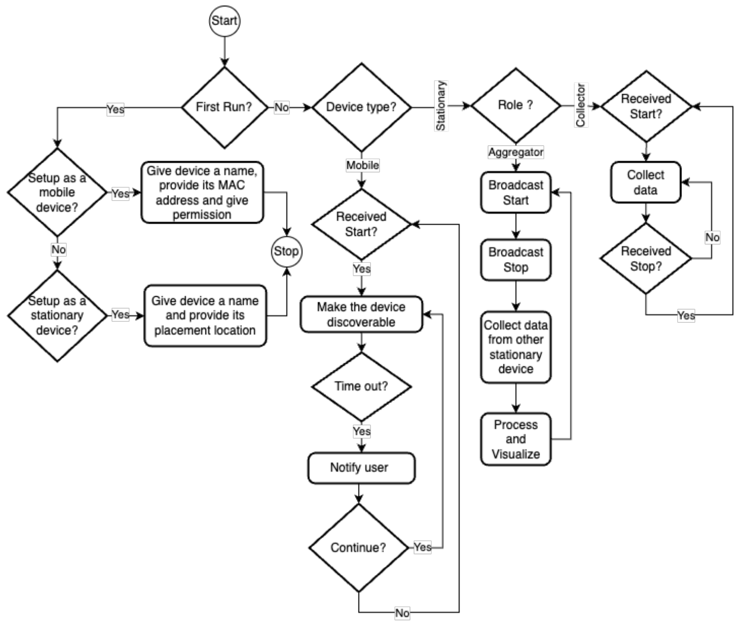

Fig. 2. System operation.

computational power to perform data formatting, duplicate detection, data aggregation, and other pre and post-processing before creating the mobility path following the developed algorithm [22]-[23]. The first stationary device, which is added to the swarm of stationary devices, acts as the aggregator. While placing the stationary devices, a name and location information has to be provided to set up the app in that device to act as a collector and/or an aggregator. Similarly, a mobile device needs to give permission to broadcast Bluetooth signal and, during the first run, also needs to provide a name and the Bluetooth MAC address of that device. The user is made aware of the data collection and privacy policy before use of the app. The stationary devices use device-to-device (D2D) communication (Wi-Fi direct) to transfer data between them for aggregation purposes. The device, which becomes a collector node first in the list of stationary devices, also acts as the aggregator node for those stationary devices. Two stationary devices can overlap their proximity radius and measure the RSSI level of the same mobile device simultaneously. Fig. 2 shows the flowchart for the working of the mobile app system. During the first run of the app on a device, the user has to choose the role of that device as either mobile or stationary. After the selection is made in all devices (mobile and stationary), the app has to be closed and run again to allow the stationary devices to synchronize themselves with the information of identifying the mobile devices and determine which stationary device will act as the aggregator. The aggregator allows the user to send start and stop signals to all the other stationary devices so that they can start scanning for mobile devices. The user needs to run the app on the mobile device and start moving around the space. Once the mobile device receives the start signal from the aggregator, it will make the device discoverable so that the stationary devices can sense its presence. Since the app runs on regular Android devices instead of rooted devices, such a discoverable state has a system timeout. Therefore, the app will vibrate and remind users that they again need to turn on the discoverable mode to be still able to be sensed by the stationary device.

A. Data Collection and Processing

Each stationary device collects the following data when a data collection phase starts:

1. Date and Time when the RSSI is sensed.
2. MAC address of the mobile device.
3. RSSI Value of the mobile device.

An example of the data gathered by a stationary device can be seen in Fig. 3, where it sensed two mobile devices (identified by their unique MAC address) in two slightly different timestamps. Once all data is aggregated in the aggregator node, the following path-forming algorithm is used to create the path of mobility using that data. The RSSI mobility algorithm that was created is split into four different steps as follows:

- Step 1: Aggregate

This step cleans the data for further processing. The following are its sub-steps:

a) Separate the data per mobile device basis.
b) If the option to round up the milli second in the time stamp to the nearest second is selected, perform the corresponding rounding up of the timestamp.

```
2023-02-09 10:17:03.292,B4:CE:40:7B:97:44,-64
2023-02-09 10:17:03.767,80:9F:F5:9D:11:A9,-78
```

Fig. 3. Example of data gathered by the stationary device.

c) For each mobile device, if there are duplicate readings (happens due to how a device's Bluetooth scanner works or because of rounding up milli seconds), convert them into one using either of the following user choices:
   i. Use the mean of the RSSI values to convert duplicate entries to one.
   ii. Use the median of the RSSI values to convert duplicate entries to one.

▪ Step 2: Normalize

This is an optional step to put data into equally spaced timestamps. This step work on data per mobile device basis. This step is required if a stationary device did not sense a mobile device at a particular timestamp; however, others did. With this step, the data is placed into equally time-spaced buckets. Users can normalize data after Step 1 or during Step 3. The user can select the bucket size by selecting a time span. The following are its sub-steps:

a) The user selects a time span on which to normalize all data.

b) Find all unique timestamps in the data.

c) While there are unique timestamps:

   i. If the current timestamp is within *timestamp* + *time span*, then,
      1. Create a new data entry where the timestamp is equal to the current timestamp, and the RSSI value would be the average of all the RSSI values between this timestamp and *timestamp* + *time* span.
   ii. Otherwise, the next time stamp would be the one that is greater than *timestamp* + *time* span.

▪ Step 3: Combine

This step combines data for the same mobile device sensed at different stationary devices into one. This helps the next step to create per mobile device-based mobility path. This step gives users two different options:

- Option 1: Per row basis
   i. Find the data for the same mobile device in the same row of the table in all stationary device's data table and put them into one row of the new table.
      - If there is no data for that mobile device for a row, put 0 for the RSSI value in the combined data for that stationary device.
- Option 2: All-row basis
   i. Find the data for the same mobile device for each timestamp and put them into one row of the new table.
      - If there is no data for that mobile device for a timestamp, put 0 for RSSI value in the combined data for that stationary device.

▪ Step 4: Vectorize

The vectorizing step creates directional vectors by checking for changes in signal strengths over time. To complete the vectorizing of the data, the data must be sorted by timestamp. The user can ignore zero RSSI values or not ignore zero RSSI values in the data. A neighborhood radius can be selected if the user selects not to ignore zeros, and a RSSI value will be calculated from that radius of values. The vectors are represented with a tail and a head that indicate the movement of the mobile device. The vectorization step is comprised of two sub-steps: Vectoring and Merging.

Vectorizing:

a) Sort the data ascendingly by time.
b) Determine the start and end of the path:
   i. The stationary device with the strongest RSSI value in the first row of the data is the starting point of the path.
   ii. The stationary device with the strongest RSSI value in the last row of the data is the ending point of the path.
c) Between any two rows, for each stationary device:
   i. If the RSSI value is reducing, create a vector towards the end.
   ii. If the RSSI value is increasing, create a vector towards the start.

Merging:

a) For each vector:
   i. If the tail is equal to the start and the head is equal to the end, ignore that vector.
   ii. If the tail is equal to the start, but the head is not equal to the end, append the head to the path.
   iii. If the tail is not equal to the start, but the head is equal to the end, append the tail to the path.
   iv. Otherwise, append both the tail and the head in order to the path.
b) Append the start to the beginning of the path and the end to the last of the generated path.

More details about the vectorizing process are presented in [22]-[23].

IV. Experiments and Results

This section presents the results from different experiments using different parameters for each experiment.

A. Experimental Setup

A controlled indoor space, which is 60 feet long and 45 feet wide, was chosen to collect data. The indoor space had no walls and average furniture density. A total of twelve stationary devices were placed in the indoor space, as shown in Fig. 4. The stationary devices were placed by visually estimating the placement equally throughout the room. Each stationary device was placed at an equal height and equally spaced apart when determining the placement. Two paths were simulated for experiments. Since the paths are known, the algorithm's accuracy could be tested and verified. The participants for each experiment carried a mobile device while walking on the predetermined paths.

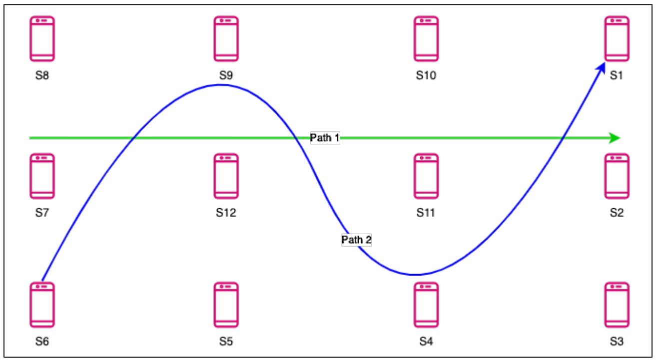

Fig. 4. Stationary device layout and experimental path of the mobile device.

The first path used four stationary devices in a straight line. The devices included S7, S12, S11 and S2. One person started at stationary device S7 and walked past S12 and S11. Once the person arrived at stationary device S2, the person paused and then continued walking back toward S7. While walking back to S7, the person passed S11 and S12, which are in the center of the experiment area. Similarly, the second path used four stationary devices, however, in a curved way. The devices included S6, S9, S4 and S1. One person started at stationary device S6 and walked to S9, turned right, walked to S4, turned left, and continued to walk to S1. Once the person arrived at stationary device S1, the person paused and then turned around to walk back using the same path and returning to S6. While walking back to S6, the person passed S4 and S9. When arriving at S4, the person took a right turn to walk toward S9. Once the person arrived at S9, they turned left and headed to S6. Each data collection phase uses a similar pace for walking and a similar pause time at each stationary device.

B. Results

The experiments were conducted to determine the mobility patterns using different parameters and combinations. Various parameters were used to determine how different parameters affect the accuracy of the mobility pattern. These experiments involve gathering RSSI data of stationary devices while a volunteer takes the mobile device and walks the two paths presented in Section V.A. Parameters were defined using the user interface application.

Since the paths are known in advance, we compare them with the path generated by the application. Table 1 shows each generated path with corresponding parameters. Creating a quantitative accuracy number is out of this research's scope, which requires adapting graph-matching algorithms [23]. However, it is evident from Table 1 that the app is very close to estimating the path. It is noticeable that although the algorithm identifies correct starting and/or ending points, it is indifferent in determining the in-between points of the path. Factors that might have affected that are how close the two stationary devices were, how fast the mobile device was moving, how long the mobile device remained in the vicinity, how the path was traversed, etc. All of these factors need further investigation.

TABLE I. PATHS GENERATED BY THE APP FROM THE COLLECTED DATA.

Path	Parameters	Actual Path	Path Generated
1	Aggregate: Mean Normalize: None	S7→S12→S11→S2	S7→S12→S2→S12→S11→S2
2	Combine: Per row basis Vectorize: Ignore zero	S6→S9→S4→S1	S6→S9→S4→S9→S4→S1

Normalization was not used for the results shown in Table 1. To see its effect and the effect of the corresponding time span during normalization, we ran with the same parameters (as in Table 1) with different time spans for path 2, as it was generated most accurately. Fig. 5. shows the effect of the time span during normalization on the length of the generated path. for normalization. The results reflected that the lower time spans had more stationary devices in the path. As the time span increased, the number of stationary devices decreased in the path. However, after a certain value (6 seconds for this instance), the generated path was missing actual path nodes (indicated by the black line). This indicates that aggregating data improves accuracy; however, doing too much might reduce the effectiveness of this step.

Additional experimentation with different parameters was performed to see their effects (or none) on the accuracy, and is summarized below:

- No effect noticed:
  - Whether we use mean or median to aggregate.
  - Whether we round milli seconds to aggregate when normalize is used.

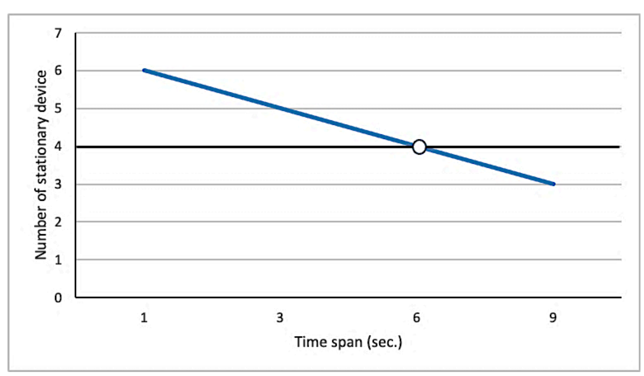

Fig. 5. Effect of time span on path length.

- Effect noticed:
  - When normalize was selected as a parameter, the results show that the paths were more accurate than when normalize was not selected.
  - The size of the time span affects the accuracy of the path. Larger time spans improve accuracy.
  - With the increase of the number of stationary devices, the accuracy of the results decreased when all other parameters remained the same.
  - When the combining step considers all rows, the generated path misses some of the stationary devices in the path.
  - When zero RSSI values are not ignored, the generated path only has the correct source and destination and misses some or all of the intermediate nodes. The neighborhood radius chosen has a direct impact on how many intermediate nodes get missing in the resultant path.

V. Conclusions and Future Work

Indoor localization is an important research area, and creating mobility paths in such an environment is challenging. This paper presents a mobile app-based mobility detection technique that uses Bluetooth RSSI data to create a mobile device's path among strategically placed stationary devices in an indoor space. A mobile application was created to automate device management, data collection, and mobility detection. Experiments were conducted using mobile and stationary devices to gather data from indoor spaces, and a mobility algorithm was developed that works with Bluetooth RSSI data. A list of parameters was identified that could be used in the algorithm to check its effectiveness. Overall, this research concludes that there is a capability to detect mobility patterns in indoor space using Bluetooth data using the presented technique without the need for any specialized hardware like similar approaches.

In the future, experiments that include increasing the area of the pre-determined path while using the same number of stationary devices could create a larger distance between the stationary devices. An additional experiment could be to use the same pre-determined path with fewer stationary devices. The experiments could determine if the distance between stationary devices, the number of stationary devices, and/or the time it takes for a stationary device to collect RSSI values after movement is detected impacts the accuracy of the data collected and results. Further research and development are required to test this approach in a real-life scenario, such as in the Anonymous Clinic, as mentioned in Section I.B. Real-world data will be collected in the clinic shortly to validate the effectiveness of this approach. Additionally, further investigation will be conducted to create a quantitative accuracy benchmark of this approach using graph-matching [23] or sequence-matching [24] algorithms.

Acknowledgment

This research is supported by NSF grant number 2131100.

References

[1] Taser, P. Y., & Akram, V. K. (2022). Machine Learning Techniques for IoT-Based Indoor Tracking and Localization. Emerging Trends in IoT and Integration with Data Science, Cloud Computing, and Big Data Analytics, 123-145.

[2] Pancham, J., Millham, R. C., & Fong, S. J. (2021). Determining the Impact of Perspex Obstacles on Bluetooth Transmission Paths Within a Simulated Office Environment for More Accurate Location Determination. In Computational Science and Its Applications–ICCSA 2021: 21st International Conference, Cagliari, Italy, September 13–16, 2021, Proceedings, Part IX 21 (pp. 410-420). Springer Intl. Publishing.

[3] Wang, Y., Ye, Q., Cheng, J., & Wang, L. (2015, December). RSSI-based Bluetooth indoor localization. In 2015 11th international conference on mobile ad-hoc and sensor networks (MSN) (pp. 165-171). IEEE.

[4] Statler, S., Audenaert, A., Coombs, J., Gordon, T., Hendrix, P., Kolodziej, K.& Rotolo, R. (2016). Beacon technologies (pp. 347-358). Berkeley.

[5] Sharp, I., Yu, K., Sharp, I., & Yu, K. (2019). Signal Strength Positioning. Wireless Positioning: Principles and Practice, 475-504.

[6] Huan Li, Hua Lu, Feichao Shi, Gang Chen, Ke Chen, and Lidan Shou. 2018. TRIPS: a system for translating raw indoor positioning data into visual mobility semantics. Proc. VLDB Endow. 11, 12 (August 2018), 1918–1921.

[7] Tiantian Liu, Huan Li, Hua Lu, Muhammad Aamir Cheema, and Lidan Shou. 2021. Towards crowd-aware indoor path planning. Proc. VLDB Endow. 14, 8 (April 2021), 1365–1377.

[8] Lee, B. G., Lee, Y. S., & Chung, W. Y. (2008, November). 3d map visualization for real time rssi indoor location tracking system on pda. In 2008 Third International Conference on Convergence and Hybrid Information Technology (Vol. 1, pp. 375-381). IEEE.

[9] Echeverria, V., Martinez-Maldonado, R., Power, T., Hayes, C., & Shum, S. B. (2018, June). Where is the nurse? Towards automatically visualizing meaningful team movement in healthcare education. In International conference on artificial intelligence in education (pp. 74-78). Springer.

[10] Ahlers, D., Aulie, K. G., Eriksen, J., & Krogstie, J. (2016). Visualizing a city within a city–Mapping mobility within a university campus. In Smart City 360° (pp. 492-503). Springer, Cham. https://doi.org/10.1007/978-3-319-33681-7_41

[11] Xu, J., Lu, H., & Bao, Z. (2020). IMO: a toolbox for simulating and querying" infected" moving objects. Proceedings of the VLDB Endowment, 13(12), 2825-2828.

[12] Pryss, R., Reichert, M., Herrmann, J., Langguth, B., & Schlee, W. (2015, June). Mobile crowd sensing in clinical and psychological trials--a case study. In 2015 IEEE 28th International Symposium on Computer-Based Medical Systems (pp. 23-24). IEEE.

[13] Wesolowski, A., Eagle, N., Tatem, A. J., Smith, D. L., Noor, A. M., Snow, R. W., & Buckee, C. O. (2012). Quantifying the impact of human mobility on malaria. Science, 338(6104), 267-270.

[14] Consolvo, S., Everitt, K., Smith, I., & Landay, J. A. (2006, April). Design requirements for technologies that encourage physical activity. In Proceedings of the SIGCHI conference on Human Factors in computing systems (pp. 457-466).

[15] Zafari, F., Gkelias, A., & Leung, K. K. (2019). A survey of indoor localization systems and technologies. IEEE Communications Surveys & Tutorials, 21(3), 2568-2599.

[16] Yassin, A., Nasser, Y., Awad, M., Al-Dubai, A., Liu, R., Yuen, C., ... & Aboutanios, E. (2016). Recent advances in indoor localization: A survey on theoretical approaches and applications. IEEE Communications Surveys & Tutorials, 19(2), 1327-1346.

[17] Du, Y. (2020). Collaborative crowdsensing at the edge (Doctoral dissertation, Sorbonne université). https://tel.archives-ouvertes.fr/tel-02913750/. Accessed May 10, 2022.

[18] Chen, J., Zhang, Y., & Xue, W. (2018). Unsupervised indoor localization based on Smartphone Sensors, iBeacon and Wi-Fi. Sensors, 18(5), 1378.https://doi.org/10.3390/s18051378

[19] Zhuang, Y., Yang, J., Li, Y., Qi, L., & El-Sheimy, N. (2016). Smartphone-based indoor localization with bluetooth low energy beacons. Sensors, 16(5), 596. https://doi.org/10.3390/s16050596

[20] Google Inc. (2021), Exposure Notifications: Using technology to help public health authorities fight COVID-19, https://www.google.com/covid19/exposurenotifications/. Accessed October 19, 2023.

[21] Priwgharm, R., & Chemtanomwong, P. (2011, May). A comparative study on indoor localization based on RSSI measurement in wireless sensor network. In 2011 eighth international joint conference on computer science and software engineering (JCSSE) (pp. 1-6). IEEE.

[22] Panlaqui, B. J., Fuad, M., Deb, D., & Mickle, C. (2022). Path Forming of Healthcare Practitioners in an Indoor Space Using Mobile Crowdsensing. Sensors, 22(19), 7546.

[23] Fuad, M., Deb, D., Panlaqui, B. J. G., & Mickle, C. F. (2022, July). Using RSSI to Form Path in an Indoor Space. In 2022 International Conference on Computer Communications and Networks (ICCCN) (pp. 1-2).

[24] Khan, Z., Bloom, J. S., Kruglyak, L., & Singh, M. (2009). A practical algorithm for finding maximal exact matches in large sequence datasets using sparse suffix arrays. Bioinformatics, 25(13), 1609-1616.