

Article

A Person-to-Person and Person-to-Place COVID-19 Contact Tracing System Based on OGC IndoorGML

Soroush Ojagh ^{*} , Sara Saeedi and Steve H. L. Liang

Department of Geomatics Engineering, University of Calgary, Calgary, AB T2N 4V8, Canada; ssaeeedi@ucalgary.ca (S.S.); steve.liang@ucalgary.ca (S.H.L.L.)

* Correspondence: soroush.ojagh@ucalgary.ca

Abstract: With the wide availability of low-cost proximity sensors, a large body of research focuses on digital person-to-person contact tracing applications that use proximity sensors. In most contact tracing applications, the impact of SARS-CoV-2 spread through touching contaminated surfaces in enclosed places is overlooked. This study is focused on tracing human contact within indoor places using the open OGC IndoorGML standard. This paper proposes a graph-based data model that considers the semantics of indoor locations, time, and users' contexts in a hierarchical structure. The functionality of the proposed data model is evaluated for a COVID-19 contact tracing application with scalable system architecture. Indoor trajectory preprocessing is enabled by spatial topology to detect and remove semantically invalid real-world trajectory points. Results show that 91.18% percent of semantically invalid indoor trajectory data points are filtered out. Moreover, indoor trajectory data analysis is innovatively empowered by semantic user contexts (e.g., disinfecting activities) extracted from user profiles. In an enhanced contact tracing scenario, considering the disinfecting activities and sequential order of visiting common places outperformed contact tracing results by filtering out unnecessary potential contacts by 44.98 percent. However, the average execution time of person-to-place contact tracing is increased by 58.3%.

Keywords: trajectory analysis; graph-based data model; OGC IndoorGML standard; COVID-19 contact tracing



Citation: Ojagh, S.; Saeedi, S.; Liang, S.H.L. A Person-to-Person and Person-to-Place COVID-19 Contact Tracing System Based on OGC IndoorGML. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 2. <https://dx.doi.org/10.3390/ijgi10010002>

Received: 18 November 2020

Accepted: 21 December 2020

Published: 22 December 2020

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

High transmissibility of the novel severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) has made the COVID-19 disease a global health crisis [1,2]. As we approach the end of 2020 and many pieces of research about SARS-CoV-2 have been conducted around the world, transmission ways of coronavirus are still under debate among scholars [3,4]. According to recent studies [5,6], three main transmission ways of coronavirus can be reported: (1) contact transmission that defines situations in which the infected person and someone else has direct contact or touch a common surface; (2) through virus-laden droplet transmission with a diameter larger than 5 μm (referred as respiratory droplets), and (3) through the airborne transmission of droplets with a diameter less than 5 μm (referred as droplet nuclei). Depending on the size of SARS-CoV-2-laden droplets, they can either rapidly fall out of the air in the immediate environment around the infected host (i.e., cause contamination on surfaces close to the emission point) or remain suspended in the air and travel over tens of meters [7]. Van Doremalen et al. [8] evaluated the stability of SARS-CoV-2 on various surfaces under ten different experimental conditions and found that the SARS-CoV-2 can survive up to two days on surfaces. Therefore, the spread of COVID-19 can occur directly by being in direct contact with an infected person or indirectly through touching contaminated surfaces.

Amongst the different strategies used to decrease the infection rate of COVID-19, contact tracing is utilized as a public health practice [2,9]. Using contact tracing, people who

have had close contact with an infected carrier in the past 14–21 days (the incubation period for COVID-19) are identified as people who might be at significant risk of infection [10]. This practice can be conducted manually with health authorities by interviewing infected individuals [11]. However, manual contact tracing is a time- and effort-consuming task that requires experienced contact tracers. So, the rapid viral spread of COVID-19 necessitates utilizing a scalable and digital approach for contact tracing [12]. An increase in the use of mobile technology, scalability of cloud data storage, and capability of physical devices to connect to the Internet using the Internet of Things (IoT) technology can meet the underlying requirements of applying digital contact tracing on a large scale [13].

Due to the interest of different governments in using digital contact tracing to address COVID-19, different smartphone contact tracing applications such as TraceTogether (Singapore) [14], CovidSafe (Australia) [15], and PACT (East-coast) [16] have been launched. A survey of recently introduced COVID-19 contact tracing applications can be found in [11]. Communication between nearby smartphones using their built-in Bluetooth interfaces is the most widely applied approach used by contact tracing applications [11]. Most contact tracing applications [11] only use proximity estimation and duration of contact between nearby smartphones. However, the contact tracing applications are insufficient for accurate contact tracing because they do not consider the impact of location and other contextual information [17].

The accuracy of contact tracing applications can be enhanced by considering the historical location context of users. The SARS-CoV-2 virus can be transmitted by a person touching a common surface that was previously touched by a diagnosed carrier [17]. In other words, when the common spaces were visited, the temporal sequence plays an essential role in enhancing the accuracy of contact tracing applications. The location history of users (provided by GPS) was considered in the SafePaths project in order to enhance the accuracy of contact tracing applications [17]. Similarly, He et al. [9] analyzed the location history of users in a COVID-19 contact tracing application for outdoor environments. However, indoor locations and the sequential order of visiting a common space have been overlooked [9,17]. Several reasons underlie the importance of indoor spatiotemporal trajectories for COVID-19 contact tracing applications: Enclosed indoor environments pose higher risks of community spread than outdoor environments [5,7,18], and people usually spend a larger part of their lives in indoor environments [19,20]. Therefore, an indoor location-based contact tracing approach is required to model complex spatiotemporal topology for multiple floors and intrinsic connectivity [21,22].

To the best of our knowledge, another missing gap in contact tracing applications is the inclusion of semantic contexts such as cleaning and disinfection activities. The disinfection and cleaning of commonly visited locations can effectively stop the chains of coronavirus spread [23] and should be considered by the contact tracing applications. An example scenario explaining the importance of disinfecting activities and the sequential order of visiting a place is given in Appendix A. To bridge the existing research gap, an enhanced digital contact tracing system is required to take the spatiotemporal movement trajectories of users and users' semantic contexts into consideration.

Indoor trajectory data analysis, such as contact tracing, requires a formal spatiotemporal data model as an abstraction of indoor movement trajectory [24]. The indoor raw trajectory is a temporal sequence of the time-stamped geographical coordinates of the moving object [25,26]. Keeping a record of such precise location information requires more battery power, communication, and computation overheads [25]. Trajectory segmentations try to break down a raw trajectory into fragments that carry semantic meaning to address the challenges above. In this paper, raw movement trajectories are semantically partitioned into spatial fragments called stay points. Stay points are spatial objects which carry particular semantic meaning and contain all of the geographical coordinates located within the stay point, and the moving object remains in it for a length of time that is above a certain threshold [27].

For this study, a graph-based data model is proposed for encoding spatial and temporal dimensions as well as user contexts for indoor trajectories. Using a graph data structure, representing trajectory data allows movement trajectories to be stored in natural graph form using recent graph database technologies [27]. The first component of the proposed data model defines a spatial indoor space based on the OGC (Open Geospatial Consortium) IndoorGML standard [22]. The IndoorGML standard considers an indoor space as a set of non-overlapping cell spaces and models the topological relationships between cells using one or more Node-Relation Graphs (NRGs) [28]. The second component describes the time dimension as the duration of stay in each IndoorGML cell using a temporal hierarchy. The third component, called contextual dimension, shows user contextual information such as user job type, activity type, and how vulnerable they are in terms of being exposed to the SARS-CoV-2 virus. The proposed hierarchical graph-based data model allows the indoor movement trajectory to be represented at different levels of granularities. The different levels of granularities provided by the proposed model support the aggregation of semantic indoor trajectories over three dimensions: spatial, temporal, and contextual. For this research, the following contributions are discussed:

1. A spatiotemporal graph-based indoor trajectory data model is proposed to reflect hierarchies in space, time, and user's contextual information. The OGC IndoorGML standard is incorporated to enrich stay points with topological relations amongst indoor cells. The proposed model can store and analyze semantic indoor movement trajectories regardless of the indoor positioning system type.
2. A COVID-19 contact tracing system is developed and investigated using the proposed data model. In comparison to other contact tracing applications, to the best of our knowledge, this paper is the first research to implement and evaluate both types of SARS-CoV-2 transmissions, namely, person-to-person and person-to-place. Additionally, the contact tracing application is further enhanced by including the place's disinfection history based on user's contextual information.
3. The spatial topologic relationships extracted from OGC IndoorGML in the proposed data model is used in a preprocessing technique to filter out semantically invalid trajectory points.

User privacy is a big challenge for contact tracing applications. Using historical location or proximity information can be considered as a threat to user privacy [17]. As mentioned in [29], 90 percent of people can be identified using only four trajectory points. Hence there is always a trade-off between user privacy versus the effectiveness of contact tracing applications. Although the application of privacy protection techniques is outside the scope of this paper, some considerations are taken into account in order to protect user privacy, as discussed further in Section 6. Moreover, the scope of this research focuses on indoor movement trajectories by considering an active Bluetooth Low Energy (BLE) beacon for each indoor cell. Although the topological relationships amongst indoor cells are used to filter out semantically invalid trajectory points, reconstructing missing trajectory points is outside the scope of this paper. Additionally, the automatic extraction of users' contexts such as cleaning activities and job type is outside the scope of this paper.

The rest of this paper is organized as follows: Firstly, the definition of the problems is defined in Section 2, followed by a discussion of related works in Section 3. The architecture of the proposed system is then described in Section 4. Section 5 provides details of the proposed data model, whilst the implementation results are illustrated in Section 6. Finally, Section 7 wraps up this paper with the conclusion, future work possibilities and ongoing problems.

2. Problem Definition

Consider the situation that the goal is conducting the contact tracing between users in indoor space using Bluetooth Low Energy (BLE). Unique advantages of BLE beacons like being lightweight, low cost, widely supported by smart devices, consuming less power, more flexible, and a higher Received Signal Strength Indicator (RSSI) has attracted many

scholars to use them as a dominant indoor localization system [30,31]. As seen in Figure 1 (top-left), the topographic space of a physical indoor environment, including four cell spaces, is shown in Euclidian space. Let us consider a situation that a BLE beacon is placed in each indoor cell. Figure 1 (bottom-left) shows the signal coverage of each BLE beacon. The connectivity graph among indoor cells and BLE beacons in dual space is shown using IndoorGML NRGs in Figure 1 (right).

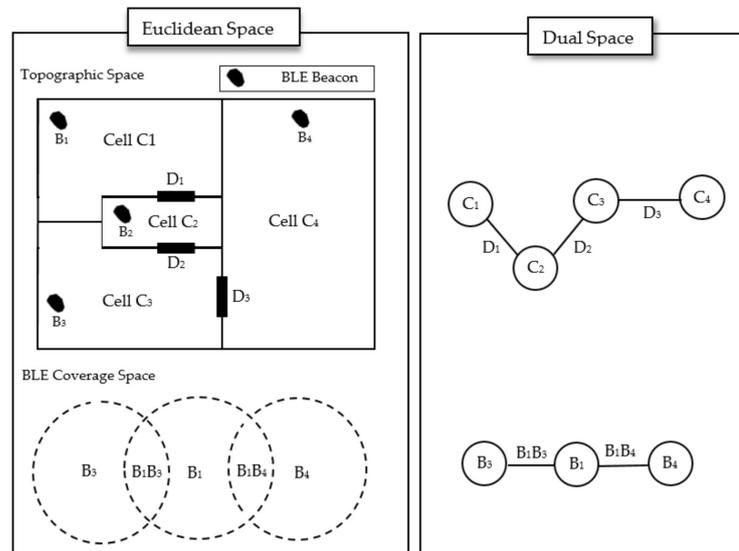


Figure 1. Representation of indoor cells and Bluetooth Low Energy (BLE) beacon coverage in Euclidian and dual space.

The first problem in an efficient indoor trajectory analysis is to model the users' trajectories in a semantic graph-based spatiotemporal data model. Figure 2 shows four users' semantic indoor trajectories considering indoor cells as stay points in user movement trajectories. As an example, user u_1 entered and then exited the indoor cell c_1 in timestamp t_1 and t_3 respectively. In this example, a point will be recorded in the semantic indoor movement trajectory of u_1 as $P_1 = \langle c_1, \Delta_1 \rangle$. In which, Δ_1 is the length of time that u_1 has spent in indoor cell c_1 . So, the semantic trajectory of u_1 can be modeled as $\{\langle c_1, \Delta_1 \rangle, \langle c_2, \Delta_2 \rangle, \langle c_3, \Delta_3 \rangle, \langle c_4, \Delta_4 \rangle\}$. The spatiotemporal representation of semantic indoor trajectories for all four users is shown in Figure 3

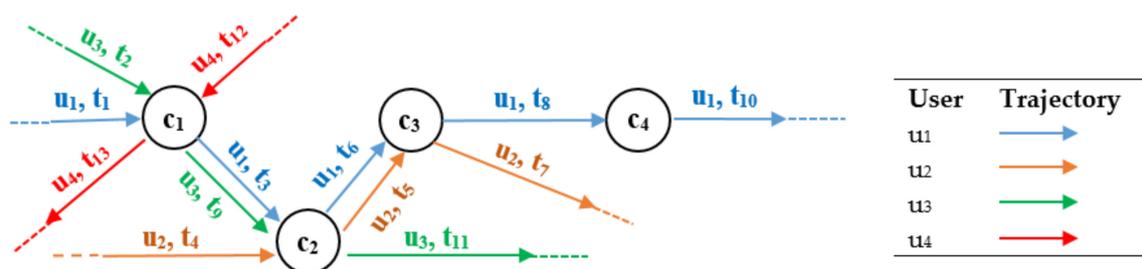


Figure 2. Representation of semantic movement trajectories of four users in the indoor environment.

The next issue in trajectory modelling is to validate if the sequence of stay points is topologically connected. It is evident that missing or unstable RSSI data directly lead to noisy semantic movement trajectories. This issue is caused by delayed signal transmissions and interferences from walls and glass doors [32]. For example, as seen in Figure 1 consider the situation that user u_1 is located in cell c_1 and BLE beacon B_1 is placed in this cell. Consider the situation in which the connection between the BLE receiver (e.g., smartphone)

and B_1 is lost for a while. During this time, the BLE receiver will be connected to the BLE beacon B_4 , considering the signal coverage shown in Figure 1. After reconnecting to B_1 , user semantic trajectory includes three stay points like $\{\langle c_1, \Delta_1 \rangle, \langle c_4, \Delta_2 \rangle, \langle c_1, \Delta_3 \rangle\}$. However, this movement trajectory is semantically invalid and $\langle c_4, \Delta_2 \rangle$ has made the movement trajectory noisy and semantically invalid. This semantic trajectory is invalid because there is no connectivity directly between indoor cells c_1 and c_4 . So, topological relationships among indoor cells (i.e., NRGs) can be utilized as a noise filtering method to extract semantically valid indoor trajectories.

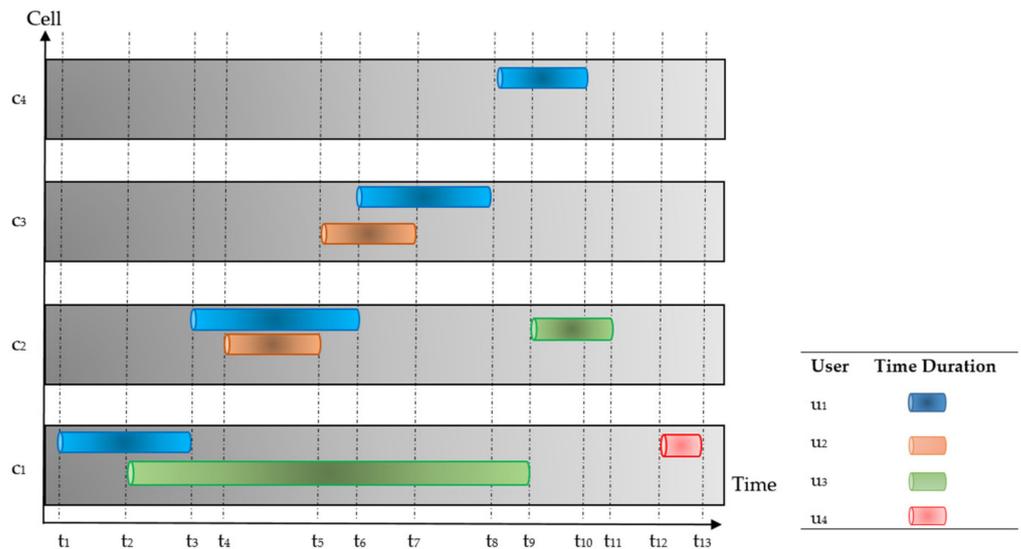


Figure 3. Spatiotemporal representation of semantic indoor movement trajectories of four users.

The last but not the least challenge in our study is to evaluate contact tracing application as sequential trajectory analysis considering users' contexts (e.g., disinfecting activity). Let us assume user u_1 as an infected person with COVID-19 and u_3 as cleaning staff. In this scenario, different spatiotemporal queries might be of interest to contact tracers. For example, a list of users who visited indoor cells just after those cells being visited by the infected user (i.e., user u_1) and before they were cleaned by the cleaning staff (i.e., user u_3). As seen in Figure 3 all four indoor cells are visited by the infected user. However, only user u_2 was in contact with the infected user u_1 in indoor cell c_2 before this room being cleaned.

3. Literature Review

In this section, we categorized the state-of-the-art related to our study into four categories: Current COVID-19 contact tracing applications are briefly reviewed in the first category. Next, existing trajectory segmentation approaches are studied as they require less computation power, communication cost, and are more human-readable. The third category focuses on trajectory representation methods. The fourth category looks over existing data models for indoor environments. Finally, the major differences between this study and other related studies will be summarized at the end of this Section.

3.1. COVID-19 Contact Tracing Applications

The exponential increase in the number of people with the coronavirus has motivated many governments and developers to leverage technology to curb the spread of COVID-19 [33,34]. As the world continues to fight against the COVID-19 pandemic, retracing close contacts of a COVID-19 Confirmed Person (CCP) to find and notify possibly exposed people at the earliest possible stage (i.e., contact tracing) is widely accepted as an available approach to “flatten the curve” [35,36]. In practice, the laborious and slow process of manual contact tracing (i.e., interview-based contact tracing) and rapid transmission of coronavirus necessitate utilizing scalable digital contact tracing systems [11,35].

Researchers and developers have proposed different digital contact tracing systems to ease the burden of manual contact tracing on public health departments. To the best of our knowledge, existing contact tracing systems can be categorized into web-based and mobile contact tracing systems. For example, both web-based (i.e., TeamSense [37]) and mobile (e.g., COVID Alert [38] and ABTraceTogether [39]) contact tracing systems are used in Canada to combat the COVID-19 pandemic. In another attempt, BLE-based wearables are recently proposed by Estimote, Inc. [40], which can be used in a web-based contact tracing system for workplaces. TraceTogether [41] is implemented as a digital contact tracing system by the Singaporean government to help automate the laborious task of manual contact tracing. According to SensorTower [42], this application with more than 3.2 million downloads (i.e., 55.36 percent of the total population in Singapore [43]) from the App Store and Google Play [44] is ranked first among all free applications in this country. As another example, Corona-Warn-App [45] is used as a digital contact tracing application in Germany. This application has received over 22.8 million downloads (i.e., 27.17 percent of the total population in Germany [43]) and ranked first among all free mobile applications in this country [46].

There are currently more than 50 smartphone contact tracing applications utilized in more than 30 countries around the world [47]. Different technologies like BLE, Global Navigation Satellite System (GNSS), Radio Frequency Identification (RFID), Wi-Fi, and QR codes have been introduced for contact tracing applications. BLE and GNSS technologies can be considered as leading technologies in COVID-19 contact tracing applications [17,47]. Focusing on coronavirus transmission, obstructions like walls between users can stop virus transmission. Comparing GNSS and BLE technologies, reduced signal strength in BLE technology can represent existing obstructions between users, while GNSS cannot consider (Appendix B). Additionally, BLE technology is more accurate than GNSS for proximity detection in enclosed places like indoor buildings and underground transit [17]. TraceTogether [41] can be mentioned as the first digital contact tracing application in the world using the BLE proximity technology. Most of the existing contact tracing applications, such as CoEpi [48] and Covid Watch [49], tried to estimate proximity between individuals using BLE proximity technology. A survey of recently introduced COVID-19 contact tracing applications can be found in [11]. Estimating proximity between users can be considered as the heart of contact tracing applications [47]. However, relying only on proximity among users will not be sufficient because coronavirus can be transmitted by touching a common surface such as a table, keyboard, and door handle. In other words, the person-to-place way of transmitting coronavirus needs to be considered as location-based COVID-19 contact tracing applications.

Berke et al. [17] tried to investigate the location-based COVID-19 by incorporating the location history of users in SafePaths project. In their contact tracing application, the user's historical location can be collected by either GPS or BLE technologies. In more detail, GPS 2-dimensions (latitude and longitude) coordinates and time are first mapped to a 3-dimensional geospatial grid. Then, time intervals when two users occupied the same place will be detected using the intersection approach across users' GPS histories. Although the authors suggested that their method can be used in the person-to-place way of coronavirus transmission, there is no evidence that the impact of person-to-place virus transmission is considered to the best of our knowledge. To clarify this gap, consider the situation that a commonplace is visited firstly by user A who is a diagnosed carrier, and after this user left the commonplace, it is visited by user B. Consider this situation has happened in a short time interval. Intersecting the location at the common time window results in finding user B as a possibly exposed user. So, the sequential order of visiting a commonplace should be considered in addition to user historical location to model the person-to-place way of transmitting coronavirus among users.

He et al. [9] tried to incorporate users' historical location in COVID-19 contact tracing as a travel companion trajectory mining application for outdoor environments. Authors followed the methodology proposed by Rong et al. [50] to design an efficient

index, called the Spatial First Time index, as a similarity metric for trajectory clustering. Their proposed similarity metric was applied to group trajectory segments that are spatially and temporally similar to the query trajectory (i.e., the trajectory of COVID-19 confirmed case) [9]. Using trajectory clustering, users who have similar trajectories to the query trajectory can be detected. Although they measured users' companionship, they did not study if a location is contaminated with a diagnosed carrier. Additionally, their proposed system only focused on outdoor environments.

Although researchers and developers have been mostly focused on technological advancements of digital contact tracing systems, broad public participation in digital contact tracing applications is required to halt the pandemic [33]. Hellewell et al. [36] evaluated the impact of isolation and contact tracing to control the pandemic using a mathematical model. According to their results, the majority of outbreaks in regions with the basic reproduction number (The expected number of secondary cases that can be infected by the infected host in a population where all individuals are susceptible) (R_0) less than 1.5 can be controllable if 50% or more of contacts are successfully traced [36]. For example, the estimated basic reproduction number in Singapore is (0.8–1.4) [51] and in Germany is (0.9–1.3) [52] with a 90% credible interval. According to SensorTower [42], TraceTogether and Corona-Warn-App contact tracing applications are publicly adopted by 55.36 and 27.17 percent of Singapore and Germany, respectively. Although the TraceTogether met the expected adoption (i.e., 50% of the country's population), the Corona-Warn-App lagged behind the expected adoption. Various factors, such as privacy concerns, anonymity, transparency, and concerns about data overusing, affect the public willingness to use digital contact tracing applications [33]. Recent works [33,35] suggest that automatic contact tracing at any public adoption rate will slow the quick spread of coronavirus. Although a qualitative analysis of the effectiveness and success of digital contact tracing applications is required, we believe that digital contact tracing applications cannot be considered a replacement for manual contact tracing.

To conclude, user location history and sequential order of visiting a place can better model person-to-person way of coronavirus transmission. However, person-to-place virus transmission is still overlooked at the time of writing this paper. Moreover, user location history has only been considered for outdoor environments. As of 16 December 2020, the disinfection history of commonly visited places is ignored by researchers for both indoor and outdoor environments.

3.2. Trajectory Segmentation

The main goal of trajectory segmentation is breaking down a trajectory into fragments that carry semantic meaning, more human-readable, and require less computation power [25]. There are mainly three categories of approaches for trajectory segmentation [25,26]. The first category takes time interval into account to divide a trajectory into fragments [53]. The second category focuses on trajectory shape and breaks down a trajectory using turning points that maintain the trajectory shape [54]. Finally, partitioning trajectories using each segment's semantic meaning, which is used in our proposed data model, can be considered the most widely applied approach of trajectory segmentation. This approach has been vastly applied in different applications such as transportation [55,56], tourism [57,58], and recommender systems [59].

A very natural way of semantic meaning-based trajectory segmentation is splitting a trajectory into segments that show stillness versus movement [25,26]. Depending on trajectory data analysis, stationary points can either be kept or skipped. For example, Yuan et al. [60] skipped the stationary points in their proposed taxi travel speed estimation approach. However, He et al. [9] have only focused on stationary points in their proposed trajectory clustering approach for COVID-19 contact tracing application. Gómez et al. [27] similarly used the concept of stationary points and assumed users' check-ins extracted from location-based social networks as stationary points that have no duration.

Although there is a large body of knowledge for trajectory segmentation in outdoor applications like transportation, only a few research pieces have been focused on indoor settings. Werner et al. [61] reduced the amount of computation using the Douglas–Peucker algorithm as a trajectory segmentation approach, which is mainly concern with the trajectory shape. Gua et al. [62] improved the accuracy of the map matching process in indoor trajectories considering semantics provided by pedestrian dead reckoning and human activity recognition algorithms. Wang et al. [63] used raw odometer data to extract step length, step count, and heading for the purpose of trajectory segmentation. In our research, the raw indoor trajectory is segmented by the definition of stay points and semantically enriched by OGC IndoorGML cell attributes.

3.3. Indoor Trajectory Model

Chen et al. [64] have used the grid partition method to abstract the indoor physical model using hexagonal grids. The IFC data model were used in this research, and movement trajectories were generated using the tool Vita [64,65]. They applied a vertical projection distance approach to transform synthetically generated indoor movement data into determined grids. The OGC has published the IndoorGML standard as a common spatial framework to represent and model indoor spaces for indoor navigation purposes [22]. OGC IndoorGML open standard provides a standard way of abstracting indoor physical environments using a multi-layered graph-based data model. The flexibility of OGC IndoorGML standard in providing spatial units with the ability to have their own semantic and topological relationships in a graph-based data structure makes OGC IndoorGML standard an appropriate alternative to model spatial dimension of indoor trajectories. Alattas et al. [20] analyzed and visualized indoor users' movement data in an evacuation exercise using the extended LADM-IndoorGML. They extracted the location of users by analyzing WiFi logs data collected from the main WiFi network of TU Delft. LADM-IndoorGML is an integration of OGC IndoorGML and Land Administration Domain Model to determine restrictions, right, and responsibilities of different groups of users for each interior space [66]. Alattas et al. [20] used a relational database (i.e., PostgreSQL) to model the 3D geometry of the IFC model and technical model of LADM-IndoorGML as an infrastructure to visualize users' movements. Following individual users and monitoring the number of users in each WiFi access point zone were two types of analyses have been carried out by Alattas et al. [20].

Kontarinis et al. [24] proposed a hierarchical semantic-enabled symbolic model for indoor movement trajectories of users who visited the Louvre Museum located in Paris, France. In [24], the physical model of the Louvre Museum was abstracted using OGC IndoorGML standard into five symbolic graph layers (i.e., building complex, building, floor, room, and region of interest). Regions of interests were defined as 51 non-overlapping zones specified by the museum administration and associated with exhibition themes. Raw movement trajectories of visitors were collected using the BLE beacon infrastructure and smartphone application (app) and then transformed and enriched by considering five symbolic graph layers. So, authors in [24] modelled visitors' indoor movement trajectories as a sequence of their presence in Louver's thematic zones. In [24], the OGC IndoorGML standard was used to semantically enrich raw trajectories. However, some significant differences between our study and their study are: (1) their proposed data model is focused on museum visitors, while we proposed a general spatiotemporal graph-based model which can be used in COVID-19 contact tracing application; (2) although spatial granularity was supported in the proposed model by Kontarinis et al. [24], the user contextual information was overlooked; (3) temporal and contextual granularity will be supported in our proposed model to fully represent the spatiotemporal nature of users' movement trajectories; (4) our proposed data model is implemented in a graph-based database instead of using a thematic representation of indoor trajectories.

3.4. Trajectory Representation

There are three ways to represent movement trajectories using a matrix, tensor, and graph data structure [25]. The matrix representation of movement trajectories is widely applied in recommendation systems. Ojagh et al. [67,68] transformed GPS trajectories of users into a matrix and then applied a collaborative filtering algorithm to provide users with personalized recommendations. Tensor representation of movement trajectories can be considered a natural extension of matrix-based transformation, with additional information as the third dimension of matrix representation [25]. Zheng et al. [69] extended the location-activity recommendation system to a user personalized recommendation system by adding users to the location-activity tensor representation of users' GPS trajectories.

Using a graph data structure, representing trajectory data allows movement trajectories to be stored in natural graph form using recent graph database technologies. Therefore, graph-based trajectory representation prevents the "impedance mismatch" problem between the data model and storage [27]. Recent advances in graph database technologies increase using a graph data structure in various applications [27,70,71]. Hu et al. [72] extracted tourist movement trajectories from users' geo-location data shared on Twitter as social media. In their research, extracted trajectories then turned into graphs using DBSCAN-based clustering. Then, network analytical methods were applied to extracted graphs to discover tourist movement patterns. Niu et al. [73] constructed a dual graph from movement trajectories considering the transportation network as a complex network. The dual graph has then been utilized in a label-based clustering approach to cluster movement trajectories and addresses the main limitation of distance-based trajectory clustering methods. Sabarish et al. [74] proposed a hierarchical clustering method based on the graph data structure to identify similar movement patterns of movement trajectories for moving trucks carrying goods. Gómez et al. [27] proposed a spatiotemporal graph data structure and transformed users' movement trajectories from the location-based social network to perform Online Analytical Processing operations on movement trajectories. Guo et al. [62] enhanced the indoor pedestrian trajectory's accuracy using the concept of a semantically enriched graph data model extracted from the floor plan.

To conclude related studies, person-to-place location history, the sequential order of visiting a commonplace, user-related contextual information such as disinfection activities are missing in digital contact tracing applications. Additionally, the state-of-the-art only focused on outdoor environments. To address the aforementioned research gaps, a graph-based data model is proposed to encode spatial and temporal dimensions as well as user's contexts for indoor trajectories. The first component of the proposed data model defines a hierarchical spatial indoor space. In this model, the raw indoor trajectories using BLE sensors are segmented by the definition of stay points and semantically enriched by OGC IndoorGML cell attributes. The benefit of using OGC IndoorGML is its graph-based structure to model the spatial topology. So, it can be adopted in our proposed data model to represent and store the spatial dimension of movement trajectories in their native graph form. The second component describes the time dimension as the entrance time for each IndoorGML cell using a temporal hierarchy. The third component, called the contextual dimension, shows users' contextual information such as disinfecting activities. We used a graph database to represent the proposed spatiotemporal indoor trajectory model as an efficient approach for contact tracing query processing.

4. Methodology

In this section, the methodology and preliminary definitions applied for this research are explained. The proposed graph-based hierarchical data model for semantic indoor movement trajectory data analysis is then described. This section is focused on COVID-19 contact tracing. However, the proposed data model can also be used as a general-purpose indoor movement trajectory.

4.1. Semantic Trajectory Segmentation

Considering the situation in which four BLE beacons are deployed in an indoor environment, the raw indoor movement trajectory of u_1 is shown in Table 1. b_i^j indicates the RSSI value measured by the user smartphone for BLE beacon B_i for timestamp t_j . As shown in this table, u_1 smartphone received four RSSI measurements from four BLE beacons (i.e., B_1, \dots, B_4) located near it.

Table 1. Raw indoor movement trajectory of a sample user.

User ID	Time	B_1	B_2	B_3	B_4
1	t_0	b_1^0	b_2^0	b_3^0	b_4^0
1	t_1	b_1^1	b_2^1	b_3^1	b_4^1
1	t_2	b_1^2	b_2^2	b_3^2	b_4^2
1	t_3	b_1^3	b_2^3	b_3^3	b_4^3
1	t_4	b_1^4	b_2^4	b_3^4	b_4^4
...
1	t_j	b_1^j	b_2^j	b_3^j	b_4^j

Table 1 shows the indoor movement trajectories for a single user and is called Raw Movement Trajectory (RMT). Given the RMT, the notation of raw indoor movement trajectory can be defined as follows:

Definition 1 (Raw Indoor Trajectory). An indoor raw movement trajectory (i.e., RMT) for the user u_i is a temporal sequence of RSSI values from all of the visible BLE beacons deployed by an indoor positioning system. The measured RSSI values from visible BLE beacons depend on the user geospatial locations. Thus the raw indoor trajectory of a user can be formulized as:

$$RMT = \left[\left(u_i, t_0, \left\{ b_i^0 \mid i \subseteq N_0 \right\} \right), \left(u_i, t_1, \left\{ b_i^1 \mid i \subseteq N_1 \right\} \right), \dots, \left(u_i, t_j, \left\{ b_i^j \mid i \subseteq N_j \right\} \right) \right] \quad (1)$$

in which, $N_j \subseteq N$ denotes the number of visible BLE beacons close to the user smartphone at each timestamp j , and N is a set of all BLE beacons ($\{B_1, B_2, \dots, B_N\}$) deployed in an indoor environment. The set $\{b_i^j \mid i \subseteq N_j\}$ represents a set of RSSI measurements (b_i^j) from visible BLE beacons at timestamp j . The order of $t_0 < t_1 < \dots < t_j$ is considered a natural order in the time frame for geospatial points visited by the user in the RMT.

As seen in Table 1 the RMT contains a huge amount of data which make the trajectory data analysis time-consuming. Additionally, contact tracers might not be interested in such detailed information regarding the RSSI value for all visible beacons for each timestamp. The proximity zone that the user is located in, and also the amount of time that he/she stayed there might be of greater interest for contact tracers. Hence the raw indoor movement trajectories can be segmented using the semantic concept of ‘‘Place of Stay (PoS)’’. The PoS semantically refers to the proximity zone of the BLE beacon with the highest RSSI value in which the user has spent Δ_k length of time. The notation of PoS is defined as follows:

Definition 2 (Place of Stay). Consider δ_{min} and δ_{max} as the minimum and maximum RSSI values associated with the proximity zone of the BLE beacon B_i with the highest RSSI value. All geospatial points with their measured RSSI value belonging to the $[\delta_{min}, \delta_{max}]$ range are defined as the PoS associated with the BLE beacon B_i . The length of time (i.e., Δ_k) that the user spent at the PoS is the time difference between the entrance and exit time of the user in this proximity zone.

Also, user contextual information can be considered in semantic indoor trajectories. In the COVID-19 contact tracing application, we denoted job type, activity type, and the level of user vulnerability to COVID-19 virus exposure as user’s contexts in each PoS. User

contextual information can be manually entered by the user. Table 2 illustrates the semantic trajectory segmentation of user u_1 .

Table 2. Semantic indoor movement trajectory of a user.

User ID	Time	Time Duration	Place of Stay	Health Status	Activity Type	Vulnerability
1	t_0	Δ_1	B_1	Healthy	Cleaning	v_1
1	t_1	Δ_2	B_2		Visiting	
1	t_2	Δ_3	B_3		Visiting	
1	t_3	Δ_4	B_4		Visiting	

The notation of semantic indoor movement trajectory is defined as follows:

Definition 3 (Semantic Indoor Trajectory). *The semantic indoor movement trajectory (i.e., SMT) is a temporal sequence of visited PoSs including user ID, the time that the user enters the PoS, duration of time stayed in the PoS, and user contextual information in each PoS. The notation of semantic movement trajectory of user u_1 (i.e., SMT_1) according to Table 2 is defined as follows:*

$$SMT_1 = [(u_1, t_0, \Delta_1, B_1, \text{Healthy}, \text{Cleaning}, v_1), \dots, (u_1, t_3, \Delta_4, B_4, \text{Healthy}, \text{Visiting}, v_1)] \quad (2)$$

in which, $B_i \in B_{Dom}$ and B_{Dom} (BLE beacon with the highest RSSI value) is an indicator of the visible set of BLE beacons (i.e., $\{b_i^j \mid i \subseteq N_j\}$) related to the PoS proximity zone.

4.2. Multilayered Spatial IndoorGML-Based Data Model

To support interoperability between indoor location-based services, OGC published an XML-based exchange standard indoor data model called IndoorGML in 2014 [75]. The cellular space can be considered the underlying concept of IndoorGML in order to provide an abstraction of the physical indoor environment [76].

Definition 4 (Cellular Space). *The cellc is defined as the basic unit type of the indoor primal space of the IndoorGML spatial data model. The cellular space \mathbb{C} is considered the union of non-overlapping cells $c_i \in \mathbb{C}$ which is an abstraction for the given physical indoor space \mathcal{P} [76].*

To abstract the indoor physical space using IndoorGML, both geometrical and topological properties need to be defined within the cellular space. Geometrical properties that define spatial extent for cells and their boundaries can facilitate the computation for indoor distance [75]. However, geometrical properties are not necessarily required for many applications (e.g., contact tracing). Using the Poincaré duality theorem three-dimensional cells and their relationships, topographic indoor spaces can be transformed into corresponding dual spaces. In the dual spaces, a topological, or an equivalent adjacency graph, visualizes indoor cells and their adjacency relationships using nodes and edges. Edges in the adjacency graph can be generally classified into navigable (e.g., doors) and non-navigable (e.g., walls) links. As shown in Figure 4 a connectivity graph can be extracted by considering only the navigable links in the adjacency graph. As the connectivity graph illustrated in Figure 4 does not represent the geometrical properties of cells, it is called a logical connectivity graph [75].

IndoorGML offers a mechanism called the multi-layered space model that supports different interpretations for indoor spaces [75,76]. For example, BLE signal coverage can be defined as a new interpretation for the topological space as shown in Figure 5. Each interpretation has its own cellular space. Geometrical and topological properties can be defined for the corresponding cellular spaces. Additionally, inter-layer types of connections can be used to visualize relationships amongst different cellular spaces. The multi-layered space will be defined as an overlay of interpretations. An example of this is when eight BLE beacons were installed in the topological space as shown in As illustrated in Figure 5 the BLE signal coverage in dual spaces can be considered for the extraction of a logical connectivity graph for BLE beacons. The multi-layered space model

that includes inter-layer relationships amongst different interpretations is shown in Figure 6. Following the same concept, IndoorGML offers a hierarchical graph that covers spatial granularity in cellular spaces [75,76]. An example would be the situation in which the indoor space illustrated in Figure 4 belongs to building *b*. As shown in Figure 6 building *b* will be defined as the highest layer of the hierarchical structure offered by the IndoorGML.

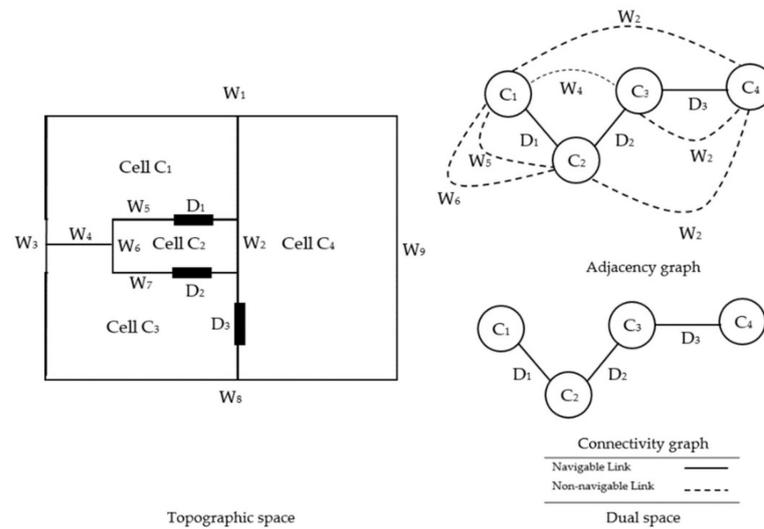


Figure 4. Extracting adjacency and connectivity graph from two-dimensional topologic indoor space.

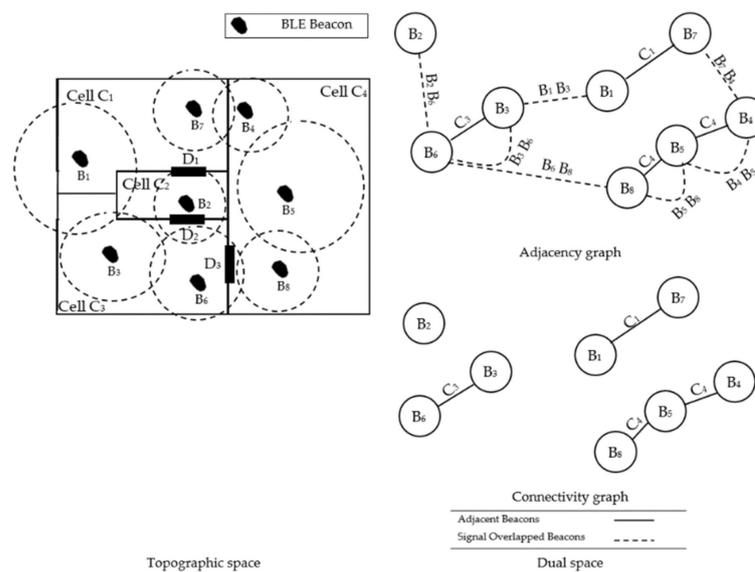


Figure 5. Extracting adjacency and connectivity graph for BLE beacons from two-dimensional topologic indoor space.

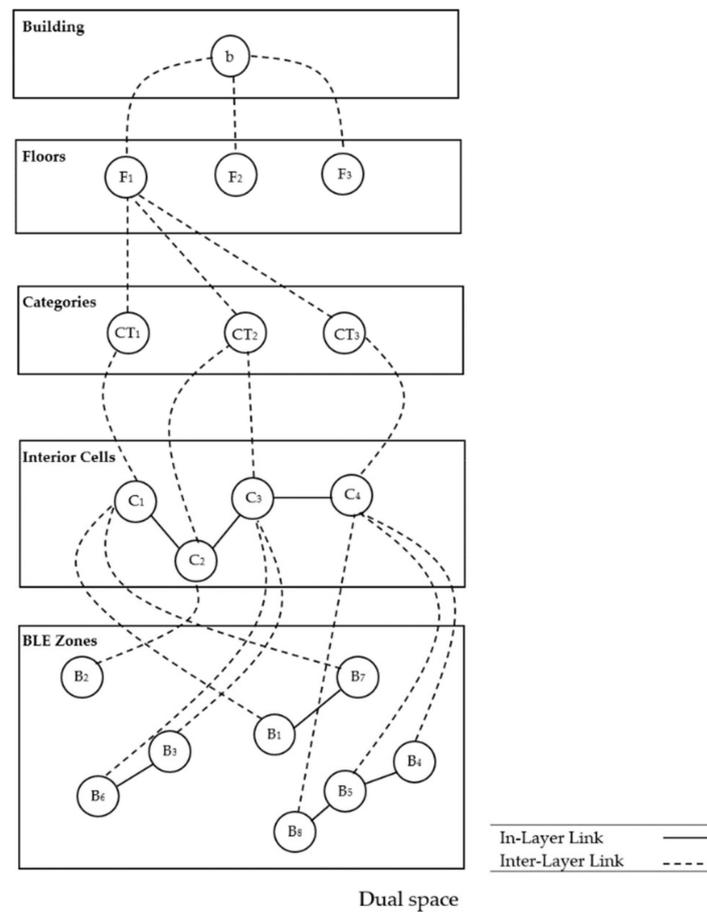


Figure 6. An example of hierarchical structure considered for spatial granularity.

4.3. Proposed Graph-Based Indoor Trajectory Modelling

As seen in (2), three main components were used to define the semantic indoor trajectory. The first component determines the spatial space using the concept of PoS. The second component describes temporal space using a pair of coordinates similar to (t_j, Δ_k) in which t_j indicates the time the user entered the PoS and Δ_k indicates the length of time the user spent in the PoS. The third component, called the contextual dimension, shows user contextual information such as their job type, activity type, and how vulnerable they are in terms of exposure to the SARS-CoV-2 virus.

Three granularity levels will also be defined in order to support aggregation in semantic indoor trajectories: temporal, contextual, and spatial. The temporal hierarchical structure of t_j is defined as *Instant* \rightarrow *Minute* \rightarrow *Hour* \rightarrow *Day* \rightarrow *Month* \rightarrow *Year* in the proposed data model. In addition to temporal spaces, hierarchical structure is also considered for contextual information. Vulnerability is considered as part of the incorporated contextual information for this research. A diverse range of factors influences COVID-19 vulnerability (e.g., age, background medical conditions) [77]. However, for the sake of simplicity, mobile users are only asked to report how vulnerable they think they are in terms of being exposed to the SARS-CoV-2 virus. Vulnerability levels can be entered as an integer value between zero and ten. This integer can then be categorized into high-level and low-level.

The spatial hierarchy of the proposed data model is based on the OGC IndoorGML multi-layered spatial representation. The spatial granularity of the proposed data model is shown in Figure 6. The proposed indoor spatial hierarchical structure is defined as *BLE Zone* Δ *Interior Cell* \rightarrow *Category* \rightarrow *Floor* \rightarrow *Building*.

Spatial hierarchical layers shown in Figure 6 from bottom to top are defined as follow:

- The BLE zones layer (as shown in Figure 5) is investigated as the finest level of the spatial hierarchy because multiple BLE zones can cover an interior cell;
- The Interior Cells layer divides the indoor space into individual spatial units divided by walls. Interior cells may have different semantic categories: Laboratory, meeting room, personal office, washroom, corridor, stair, elevator, and kitchen;
- The Category layer is defined as a group of interior cells with a similar type of semantic category. For example, the laboratory category of interior cells can be assigned to either chemical or mechanical laboratories;
- The Floors layer is considered a higher spatial granularity of categories. For example, the third floor includes the library, office, and chemical laboratory categories;
- The Building layer is comprised of the various floors associated with it.

An example of modelling semantic indoor trajectory that considers different hierarchical structures (i.e., temporal, contextual, and spatial) is shown in Figure 7. In Figure 7 circles with dashed blue lines show the trajectory data points in SMT. For example, the central blue dashed point shows the SMT of a user with ID#1 who is a diagnosed carrier. The user has an entrance time of 01: 12: 23Z to the proximity zone of BLE beacon B2 for the duration of 1177 s as shown: $[u_1, 2020-07-25T01:12:23Z, 1177, B_2, COVID19, visiting, 8]$ From the proposed data model, we can infer that the user was located at cell ID#123 which belongs to the library category on the third floor of building b1.

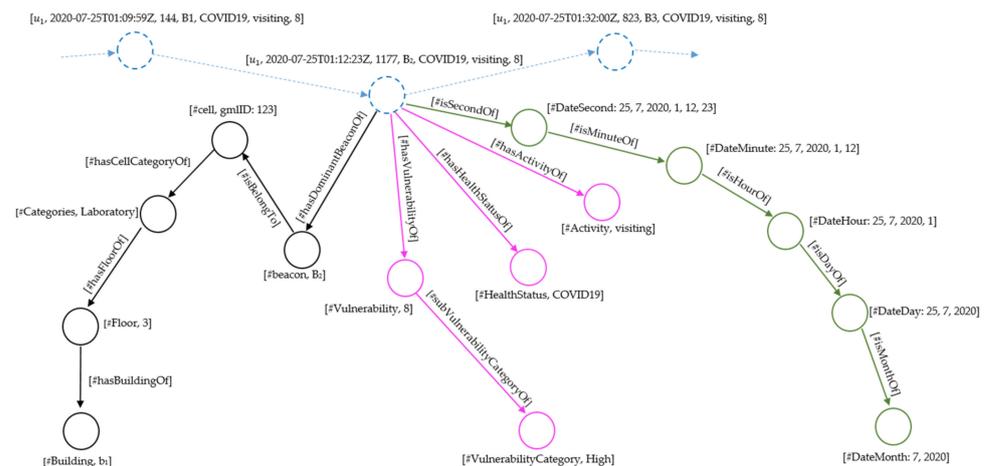


Figure 7. An example of the proposed graph-based indoor trajectory modelling.

5. System Architecture

To evaluate the proposed data model in contact tracing application, cloud-based architecture is developed. The architecture includes three layers (Figure 8): “Data Collection”, “Cloud Data Storage and Management”, and “Visualization”. We have adopted “cloud Data Storage” and “Visualization” layers from SensorUp (<https://sensorup.com/>) architecture. The data collection layer (i.e., a smartphone Android app) is mainly responsible for measuring RSSI values from different visible BLE beacons. The user raw indoor movement trajectories are then segmented in the smartphone app to reduce the communication cost. Smartphone app provides users with both “offline” on device and “online” cloud storage. The user can choose to be in an “online” or “offline” mode. When a user is entered in the proximity zone of a BLE beacon in online mode, the smartphone app will start counting seconds. When the user leaves the dominant BLE beacon’s proximity zone, a record will be sent to the cloud layer. The record consists of different types of information, including the dominant BLE beacon information (i.e., dominant BLE beacon Identification (ID) and average RSSI), temporal data (the date and time of the user entered and exited the

proximity zone and the length of time that user has stayed there), and user-related contexts (i.e., user's health status, activity type, and vulnerability level as entered by the user). In "offline" mode, such a similar record is stored internally in the smartphone's database (i.e., SQLite database). In "offline" mode, whenever the users willing to share their movement trajectory, a bulk data transfer mechanism is applied to share all stored trajectory points with the cloud server.

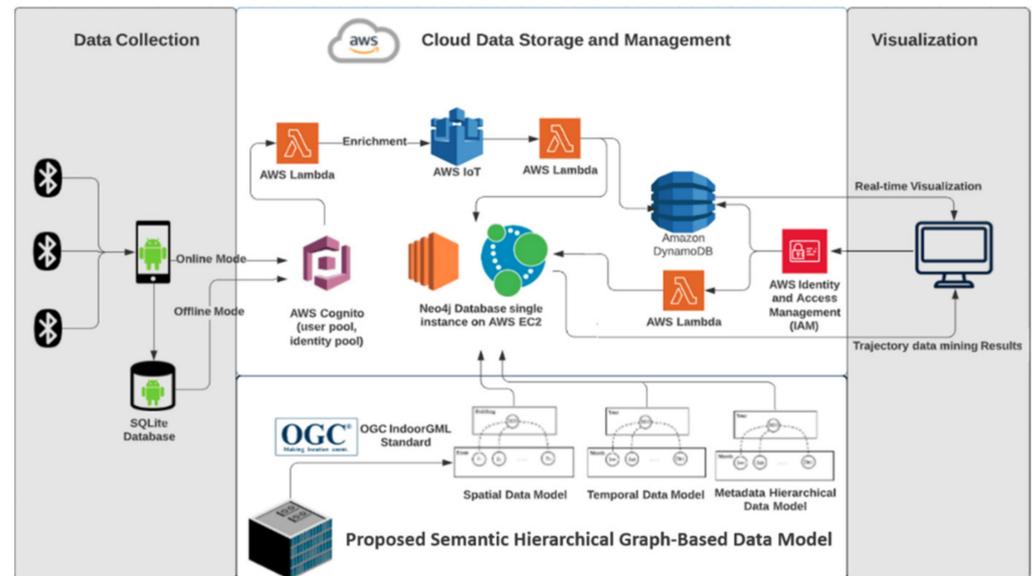


Figure 8. System architecture.

The second layer is the cloud data storage and management layer developed using Amazon Web Services (AWS). This layer is mainly responsible for handling users' and end-users' identity and access management and data storage. As shown in Figure 8, Amazon AWS Cognito (<https://aws.amazon.com/cognito/>) is used as a fully managed service for authentication and authorization of smartphone apps. AWS Cognito supports identity and access management using Cognito User Pool and Cognito Identity Pool as standard authentication and authorization services. Amazon AWS Lambda (<https://aws.amazon.com/lambda/>) is applied in this layer as a server-less and event-driven computing service. As an instance, an AWS Lambda is triggered to enrich the indoor trajectory as soon as a record from an authenticated smartphone app user is received. A data record is taken as an array of trajectory data point records in JavaScript Object Notation (JSON) format in this enrichment process. For each trajectory point, a mapping function will be applied to map the captured BeaconID to a unique OGC IndoorGML cell in the proposed data model. Then, the output is published to AWS IoT as a standard GeoJSON record. Amazon AWS IoT (<https://aws.amazon.com/iot/>) service is used in this research as a managed cloud service to support cloud data sharing for billions of devices and trillions of messages.

The data records published on AWS IoT Core will then be stored in Amazon DynamoDB (<https://aws.amazon.com/dynamodb/>) and the Neo4j graph database using another AWS Lambda. Amazon DynamoDB is used as a fully managed No-SQL scalable database offered by Amazon. An instance of the Neo4j database is also deployed in Amazon EC2 (<https://aws.amazon.com/ec2/>) as the most popular open-source graph database according to DB-Engines ranking (<https://db-engines.com/en/system/Neo4j>). The Neo4j graph database can natively support graph data storage, including nodes and relationships among nodes. The reason behind using two databases in this research is that Amazon DynamoDB is used to visualize enriched trajectory data using SensorUp Explorer web dashboard. While the Neo4j database is used to store and manage the spatiotemporal trajectory for the proposed semantic indoor trajectory data model. The proposed semantic

hierarchical graph-based data model implemented by the Neo4j database is responsible for importing hierarchical data: spatial, temporal, and contextual.

Finally, the last layer (i.e., visualization) provides end-users (e.g., contact tracers and building managers) with visualization tools to interact with the proposed system. AWS Identity and Management (IAM) (<https://aws.amazon.com/iam/>) is used in this research to manage end-users' access to AWS.

6. Implementation

6.1. Real-World Data Sets

To evaluate the proposed indoor movement trajectory data model, a real-world experiment is designed. A smartphone app is developed in this experiment, and cloud data storage and management were set up. A total number of 20 users are asked to install the smartphone app and collect data on their Android smartphones. Four of the users have randomly selected as CCP and two users as cleaning staff. Calgary Centre for Innovative Technology (CCIT) building located in the University of Calgary (UofC) Campus is selected as the test area of our experiment. Figure 9 shows the 3rd-floor plan of the CCIT building, BLE beacons' locations, and connection between indoor cells. For this experiment, 41 BLE beacons from six different BLE beacon manufacturers are utilized in 41 indoor cells. Table 3 lists the details of different types of BLE beacons used in our experiment, as shown in Figure 10.

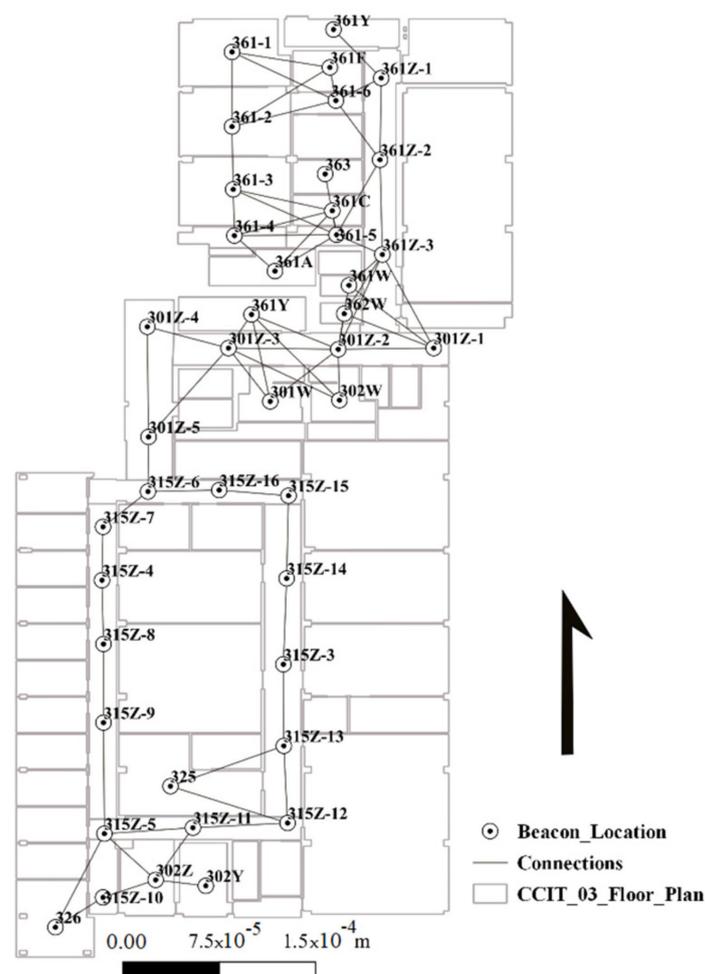


Figure 9. Floor plan of the third floor of Calgary Centre for Innovative Technology (CCIT) building.

Table 3. Details of BLE beacons used in real-world experiment.

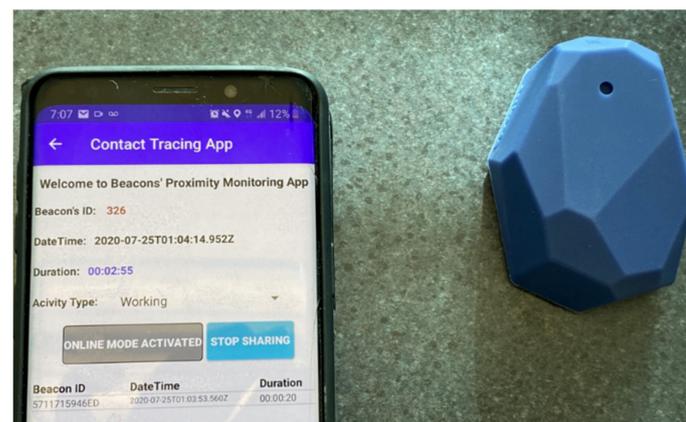
BLE Beacon Type	Manufacturer	No. of BEL Beacons in Experiment
Bluetooth Beacon	Estimote (https://estimote.com/)	8
IBKS PLUS	Accent Systems (https://accent-systems.com/)	8
IBKS 105	Accent Systems	9
IBS CARD	Accent Systems	7
RadBeacon Dot	Radius Networks (https://www.radiusnetworks.com/)	5
RadBeacon Chip	Radius Networks	4

**Figure 10.** Six different types of BLE beacons from three different BLE manufacturers used in real-world experiment.

The smartphone app's main responsibilities are measuring RSSI values for all visible BLE beacons, conducting trajectory segmentation, and pushing data records to the cloud data storage. Figure 11a,b show the User Interface (UI) of the developed smartphone app when it is capturing RSSI of a Bluetooth Estimote and an IBKS PLUS BLE beacon, respectively. In the real-world experiment, all users were asked to spend four hours on 25 July 2020 in the CCIT building. A GeoJSON payload showing a single enriched PoS captured by the developed smartphone app is shown in Figure A3. In this experiment, a total of 582 PoS records were received in the AWS IoT Core. The ultimate goal of our real-world experiment is to evaluate the functionality of the proposed graph-based data model in the contact tracing application.

6.2. Validating Semantic Indoor Movement Trajectories

In the real-world experiment, we designed an experiment to directly detect semantically invalid indoor movement trajectory caused by additional, missing or unstable RSSI data. Considering topological relations between indoor cells (i.e., logical connectivity graph) extracted from OGC IndoorGML, they can largely eliminate semantically invalid trajectory segments. In this research, a logical connectivity graph in dual space is considered in the AWS Lambda function to filter semantically invalid trajectory segments. The applied algorithm is shown in Appendix E.6. Figure 12 shows the semantic hierarchical spatial data model and connections (in the Neo4J database) of CCIT building, including OGC IndoorGML cells, in-layer, and inter-layer relations.



(a)



(b)

Figure 11. The user interface of the developed app on Samsung Galaxy S9 smartphone capturing the proximity of a user from (a) Bluetooth Estimote beacon (b) IBKS PLUS beacon.

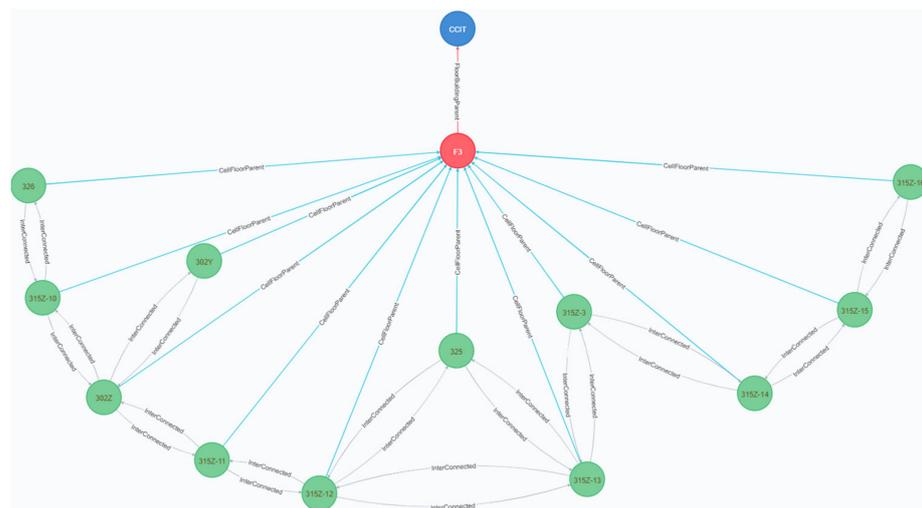


Figure 12. Hierarchical spatial graph-based representation of the 20 selected building including OGC IndoorGML cells (green circles), floor (red circle), building (blue circle), inter-layer topological connections among IndoorGML cells (gray arrows), intra-layer topological connections among IndoorGML cells and their corresponding floor (light blue arrows), and intra-layer topological connections floor and corresponding building (red arrow).

6.3. Simulation Data Sets

In addition to the real-world experiment, a simulation is developed to generate users' trajectories for the same building setting. To generate synthesized indoor trajectories,

CLI (Command-Line Interface) application with Node.js is developed. The source code of the developed Node CLI application is publicly available on GitHub (https://github.com/soroushojagh/Indoor_Trajectory_Data_Analysis). In this application, the logical connectivity graph between indoor cells extracted from OGC IndoorGML is considered to synthesize semantically valid indoor movement trajectories. The Random Walk technique, as a stochastic approach, is used to synthesize random indoor movement trajectories for 20,000 users. This simulated dataset contains 453,640 PoS records for all 20,000 users to evaluate the proposed method using a higher number of users. However, as we did not have the plan of many buildings at the UofC Campus, we consider all trajectories in the same building for a two-week time period. In this simulated dataset, we considered 20% and 10% of users as CCP and cleaning staff types, respectively. So, there are 4000 CCP users and 1000 cleaning staff users in total in the simulated dataset. Simulated indoor movement trajectories of all 20,000 users are publicly available in GitHub (https://github.com/soroushojagh/Indoor_Trajectory_Data_Analysis/tree/master/Data/User_Trajectories) for further trajectory analysis.

6.4. Data Privacy

The Amazon user and identity pools were used as a fully managed service for user authentication and authorization respectively. For the Amazon user pool, a unique identification (ID) token was assigned to each smartphone user in order to keep the user anonymous. With regards to the Amazon identity pool, user access to back-end services were managed based on their authorization. Additionally, users have full control over what data they are willing to share with the cloud. To be more precise, they can stop sharing information with the cloud whenever they choose not to. Additionally, by using user authorization controls, no user is allowed to trace the location of other users. Any user requests for possible contact with diagnosed carriers will receive only the results of the trajectory analysis. Since no further information will be provided for users regarding the time and place they were potentially exposed, the trajectory and ID token of diagnosed carriers will remain anonymous.

6.5. Data Visualization Tool

For the visualization purpose of this research, SensorUp Explorer is used as a spatiotemporal web dashboard developed by SensorUp Inc. A short demonstration video of live trajectory data visualization in SensorUp Explorer and Amazon DynamoDB for this research's real-world experiment is shown in Video S1.

6.6. Storing Semantic Indoor Trajectories

Considering the concept of the semantic indoor trajectory (2), a temporal sequence of PoSs is stored in the Neo4j graph database. Each PoS reflects a node with relation to user context and proximity zone. This node is labelled as a Check-in type with temporal information and user-related metadata properties. As an example, consider the situation that user u_1 entered the proximity zone of B_1 in time t_1 and stayed there for Δ_1 seconds and finally left this proximity zone in time t_2 . In this example, a Check-in type of node like ch_1 is created in the Neo4j database. This node has two relationships. The first relationship shows the relation of ch_1 with user 1 who has created such a PoS. Meanwhile, the second relationship shows the relation between ch_1 with the OGC IndoorGML cell hierarchy. This node also has temporal properties, including entrance time, duration of stay, and exit time. The indoor semantic trajectory of u_1 in a real-world experiment is shown in Figure A4. Details of the number of PoSs, nodes, and relationships stored in the Neo4j graph database in both real-world and simulated experiments are summarized in Table 4.

Table 4. Details of stored semantic indoor movement trajectory in the Neo4j graph database.

No. of Users	No. of PoSs	No. of Relations	Total No. of Nodes
20	488	1149	551
200	4815	9803	5058
2000	47,783	95,739	48,826
20,000	453,640	907,453	473,683

6.7. Contact Tracing Application

In this section, a list of spatiotemporal trajectory data queries for the COVID-19 contact tracing application was selected. Each of the spatiotemporal queries was executed in graph databases with different data sizes. It is worth mentioning that Cypher Graph Query Language was used for this research as a declarative graph query language for the Neo4j database.

Query 1 (Contaminated cells by a CCP): In this query, the goal is finding possibly contaminated geospatial cells visited by a single CCP. According to [78], it is assumed that only people who were in close contact with the user for longer than 15 min would be possibly infected. Accordingly, a cell is contaminated if a CCP has visited it for more than 15 min. The Cypher code for this spatiotemporal query can be found in Appendix E.1.

Query 2 (Contaminated cells by all CCPs): This query is an aggregation on Query 1 for all CCPs. There are four, 40, 400, and 4000 CCPs in our real-world and simulated databases, to be more precise.

Query 3 (Temporally constrained contaminated cells by all CCPs): In this query, the list of contaminated cells (i.e., coming from Query 2) is filtered by a selected time window. For example, the Cypher code to find all contaminated cells visited by CCPs from 2020-07-25T02:29:52.461Z to 2020-07-25T02:58:59.461Z can be found in Appendix E.2.

Query 4 (Contact tracing for a single CCP): This query uses the contact tracing method proposed by [9,17] to analyze person-to-person contacts for 15 min duration of time in a commonly visited cell. A list of possibly infected users by considering their contacts with a single CCP user is reported.

Query 5 (Contact tracing for all CCPs): This query is an aggregation on Query 4 for all CCPs. This query will prove that our proposed data model is able to consider all CCPs instead of a single CCP. So, a list of possibly exposed users who were in close contact with each of the CCPs for longer than 15 min will be reported in this query. The algorithm of this query is presented in Algorithm 1. The Cypher code for this query can be found in Appendix E.3.

Algorithm 1: Contact Tracing

Input: SMT for all CCP and all ordinary users

Output: A list of possibly infected users

Initialize:

1. $cCells[] \leftarrow$ Visitedcells by all CCPS for 15 minutes (Query 2)
 2. $oCells[] \leftarrow$ Visitedcells by ordinary users for 15 minutes
 3. $pInfectedUsers \leftarrow []$
 4. **For Each** ordinary user $As u_o$:
 5. **For Each** $oCell \in oCells$:
 6. **If** ($oCell \in cCells$)
 7. **If** ($\Delta_{CCP} \cap \Delta_{u_o} \geq 15$ minutes)
 8. $pInfectedUsers \leftarrow u_o$
 9. **RETURN** $pInfectedUsers$
-

Query 6 (Temporally constrained contact tracing for all CCPs): This query is similar to Query 5 and the results of Query 5 are filtered for a specific time window. For example, a list of possibly infected users who were in close contact with all of CCPs within a selected

time window from 2020-07-25T02:29:52.461Z 2020-07-25T02:58:59.461Z is reported in this spatiotemporal query.

Query 7 (Contaminated cells considering cleaning activity): This query is similar to Query 2 filtered by cleaning activities. Visiting a contaminated place by a cleaner in a sequential order is assumed as a disinfected place (i.e., not-contaminated status) in our proposed method. For example, if a CCP user visits a cell and then it is cleaned by a cleaning user, it is assumed that this cell will not be classified as a contaminated cell to transmit the coronavirus further. The Cypher code for this query can be found in Appendix E.4.

Query 8 (Enhanced contact tracing): In this query, the person-to-place way of coronavirus transmission, the sequential order of visiting places, and the disinfection history of places will be incorporated in the contact tracing application. This query shows the flexibility of our proposed data model to consider additional parameters in COVID-19 contact tracing. The algorithm of this query is presented in Algorithm 2. The Cypher code of this query can be found in Appendix E.5.

Algorithm 2: Contact Tracing

Input: SMT for a single CCP and all ordinary users

Output: A list of possibly infected users

Initialize:

```

1.      cCells[] ← Visited cells by CCPs for 15 minutes(Query 2)
2.      tc ← CCP EntranceTime
3.      oCells[] ← Visited cells by ordinary user uo for 15 minutes
4.      to ← Ordinary user EntranceTime
5.      dCells[] ← Disinfected cells by all disinfecting users ud
6.      td ← Disinfecting user ExitTime
7.      pInfectedUsers ← []
8.      For Each ordinary user :
9.          For Each oCell ∈ oCells :
10.             If (oCell ∈ cCells) :
11.                 If NOT[oCell ∈
12.                     dCell AND MAX (tc|tc < to) < MAX (td|td < to) AND MAX (td|td < to) < to] :
13.                     pInfectedUsers ← uo
13.      RETURN pInfectedUsers

```

7. Results and Discussion

7.1. Validating Indoor Real-World Trajectories

Extracting semantically invalid indoor movement trajectories is evaluated as the third aim of this research. In our real-world experiment, all 20 users were asked to keep track of all visited BLE beacons in their proximity zone using unique BeaconIDs that were written on each BLE beacon (as shown in Figure 11). These reported BeaconIDs data were used as ground truth indoor movement trajectories. For the real-world experiment, a total of 582 PoSs was detected by all users. After comparing PoS records stored on Amazon DynamoDB and ground truth trajectories, 34 PoS records were determined to be invalid PoSs. Invalid PoS records were caused by missing or unstable RSSI values measured by the smartphone app. After applying our developed preprocessing algorithm in an AWS Lambda function (as demonstrated in Appendix E.6), 31 PoSs were recognized as invalid PoS records. The results of applying the preprocessing algorithm on the indoor movement trajectory in real-world experiments are shown as a confusion matrix in Table 5. From Table 5, it can be concluded that the preprocessing algorithm detected 73.53 percent of the semantically invalid PoS records. The missing false-negative cases in our preprocessing algorithm were caused by multiple connections (links) between the PoS records. After removing all of the 31 noisy PoSs reported by the developed preprocessing algorithm, another 551 nodes were loaded into the Neo4j graph database. For instance, there are three topological connections between the cells in the building test area of our experiment (Figure 9) “301Z-3”, “301Z-4”, and “301Z-5”. In the ground truth trajectory, the user

moved from “301Z-3” to “301Z-5”. However, the smartphone measurements showed this trajectory: [301Z-3, 301Z-4, 301Z-5]. Therefore, based on the ground truth trajectory, “301Z-4” is an invalid trajectory point. However, the proposed preprocessing method cannot detect this point since it is topologically connected to the other two trajectory points. Figure 13 shows these cells and trajectory points for both the ground truth and experiment. “301Z-4” is invalid with regard to the ground truth.

Table 5. Confusion matrix for preprocessing indoor trajectory method.

		Ground-Truth Trajectory	
		Valid Nodes	Invalid Nodes
Preprocessed Trajectory	Valid Nodes	551	34
	Invalid Nodes	554	31

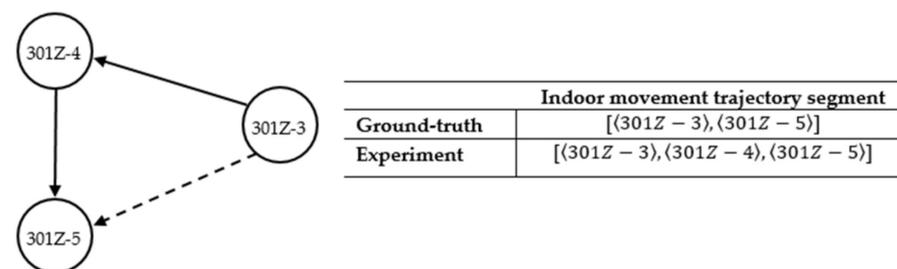


Figure 13. Representation of an existing situation in which the developed preprocessing algorithm cannot detect invalid trajectory point: dashed lines show ground-truth and solid line shows collected data from our experiment.

7.2. COVID-19 Contact Tracing Results

To evaluate the paper’s second contribution, the proposed semantic graph-based data model’s functionality is evaluated in contact tracing applications (discussed in Section 6). Each query was executed a hundred times on four different Neo4j graph databases with a different number of nodes. The first database consists of real-world movement trajectories collected by 20 users with 551 nodes. Simultaneously, the rest three databases are simulated trajectories, including 200, 2000, and 20,000 users with 5058, 48,826, and 473,683 nodes, respectively. To show the graph database query execution time for various nodes, the performance results of Query 4 are reported in Table 6 as an example. Detailed information, including minimum, average, and standard deviation of all query execution on all four databases are listed in Table A1. For visualizing the performance results, the average and standard deviation of the first three databases are represented in Figure 14.

According to the study done by Silva, F. D., [79], graph size plays an essential role in query execution time. As the general trend, it can be seen that query execution time for all queries increases with the increasing number of nodes. Looking further into average query execution times reveals increasing the number of nodes 100 times for Query 1, 2, 3, 5, and 7 leads to a rise in the average query execution time by almost less than ten times. While this statistic for Query 6 and 8 is almost 20 and 80 times, respectively.

Table 6. Query 4 (contact tracing) execution time for different sizes of the graph databases.

No. of Nodes	Query 4 Execution Time (ms)		
	Minimum	Average	Standard Deviation
551	2.00	2.98	8.24×10^{-1}
5058	3.00	5.02	1.17
48,826	6.30×10	3.52×10	1.77×10
473,683	3.61×10^2	4.28×10^2	3.47×10

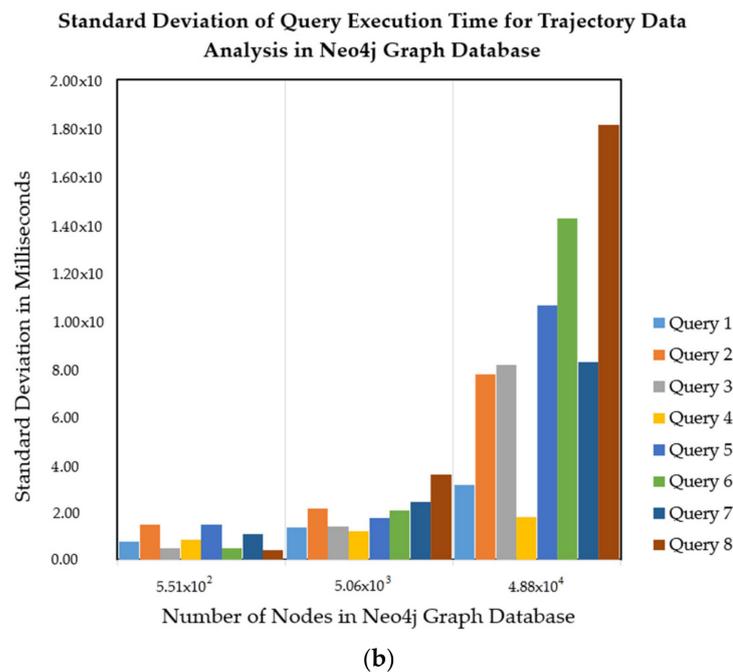
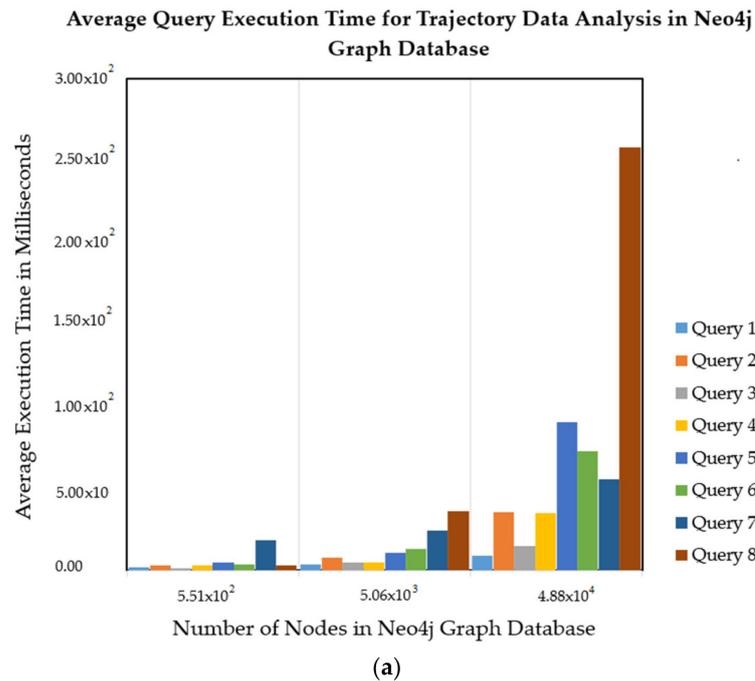


Figure 14. Representation of executing trajectory data analysis for 100 times in the Neo4j graph database with 551, 5058, and 48,826 nodes: (a) Showing average query execution time in milliseconds for different trajectory data analysis, (b) Representing standard deviation of query execution time in milliseconds for different trajectory analysis.

As seen in Figure 14a, the average execution time for Query 8 (person-to-place contact tracing) is relatively more extensive than other queries. For Query 8, the average query execution time for databases with 0.5k and 5k nodes is less than 20 milliseconds. An increasing number of nodes from almost 5k to 50k results in a rise of almost less than six times for all queries in different sizes of databases. Similarly, as seen in Table A1, increasing the number of nodes from 50k to 500k results in the same for all queries except Query 8. For Query 8, an increasing number of nodes from 50k to 500k results in a considerable rise by almost 38 times of average execution time. So, it can be concluded that Query 8 would be

more sensitive than other queries to the number of nodes as it is based on the person-to-place contact tracing application.

Looking further into details shows that queries focused on a group of users (e.g., Query 2) require more average execution time than similar queries that are focused only on a single user (e.g., Query 1 with a focus on a single CCP). It can also be concluded when the number of query trajectories increases 400 times, the required query execution time increases almost ten times. Additionally, it can be seen that applying temporal constraints on queries (e.g., Query 3) leads to less required query execution time. Considering temporal indexing for trajectory type of nodes is the underlying reason for this reduction in average query execution time.

As seen in Figure 14b, the standard deviation of query execution time increases by increasing the number of nodes for all queries. The largest standard deviation between all queries in three different databases is for Query 8 with 18.1 milliseconds. It can be concluded that Query 8 has the lowest precision for databases with 5k and 50k nodes. However, this Query has the highest precision in the database with 0.5k nodes. Moreover, although the standard deviation of Query 4 slowly increases with an increase in the number of nodes, it has the lowest rate of change among all queries. So, it can be concluded that Query 4 has the lowest sensitivity with regard to the size of databases.

7.3. Enhanced COVID-19 Contact Tracing Results

As discussed earlier, the use of user location history and an overlapping time window of 15 minutes are proposed for the state-of-the-art digital tracing app [9,17]. For this research, Query 5 is designed to consider all of the aforementioned factors in the people-to-people contact tracing application. Query 8 is designed to consider the people-to-place method of coronavirus transmission, sequential order of visiting places, and disinfection history of places. Those queries were conducted in the Neo4j databases with different number of nodes. Experimental results show that the number of reported possible COVID-19 infected users decreased in Query 8 in comparison to Query 5 (Figure 15) Query 8 successfully filtered 44.98 percent of users who were reported by Query 5 after applying the disinfecting history of the rooms. However, the average execution time of Query 8 increased by 58.3 percent (Table 7).

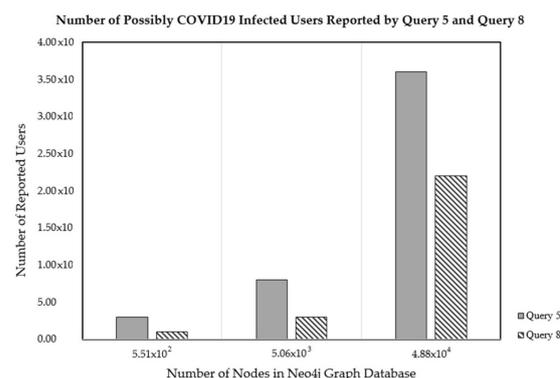


Figure 15. Number of possible COVID-19 infected users reported by different contract tracing applications in the Neo4j database with different number of nodes.

Table 7. Comparison of average execution time for Query 5 (contact tracing) and Query 8 (enhanced contact tracing).

Spatiotemporal Queries	Average Query Execution Time in Milliseconds		
	Minimum	Average	Standard Deviation
Query 5	4.37×10	6.26×10	4.58
Query 8	8.76×10	9.91×10	7.48

In this experiment, we considered 10 percent of the users as providing cleaning activities. In another experiment, a different number of cleaning users is evaluated to show the importance of disinfecting activities in coronavirus transmission. A simulated dataset with 20,000 users is evaluated by considering three different percentages (i.e., 5 percent, 10 percent, and 20 percent) of users as cleaners. As shown in Figure 16 disinfecting activities reduce the number of possible COVID-19 infected users in Query 8 by 20.06 percent, 32.34 percent, and 48.16 percent when 5 percent, 10 percent, and 20 percent of the users are considered as cleaning users. It can be concluded that the sequential order of disinfecting activities has a considerable effect on the COVID-19 contact tracing application. In other words, the task of conducting the COVID-19 medical test for the number of possibly exposed users can be decreased by considering disinfecting activities. Our proposed graph data model provides the ability to incorporate this factor for the contact tracing application.

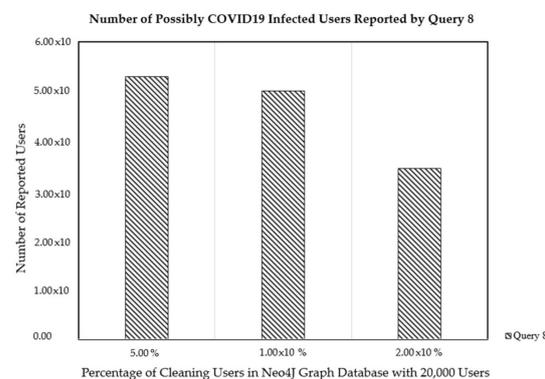


Figure 16. Number of possible COVID-19 infected users reported by Query 8 in the Neo4j database with 20,000 users and different percentages: 5 percent, 10 percent, and 20 percent of cleaning users.

8. Conclusions and Future Work

This paper introduces a graph-based semantic indoor trajectory data model that can be utilized in different indoor trajectory analyses. The OGC IndoorGML standard and its multi-layer space model are incorporated in the proposed data model for the semantic segmentation of raw indoor movement trajectories and hierarchical representation of cell spaces in a building (i.e., BLE beacon coverage, rooms, category of rooms, floors, and buildings). Three spatial, temporal, and contextual hierarchical structures were considered in the proposed data model in order to support different granularity levels for trajectory data representation. The digital COVID-19 contact tracing problem was selected as a use case for this research in order to prove the functionality of the proposed data model for trajectory data analysis. There is a large body of research concentrating on contact tracing applications in person-to-person scenarios for outdoor settings. Hence, this paper focuses instead on indoor settings and both person-to-person and person-to-place scenarios in order to expand state-of-the-art digital contact tracing.

Two experiments were designed to evaluate the main contribution of this research. A smartphone app was developed to collect raw movement trajectories from 20 users for the first real-world experiment. A total of 41 BLE beacons of various types were deployed in a building at the UofC Campus with the assumption that at least one beacon was deployed in each room. Amazon Cloud Web Services was incorporated in order to implement scalable data storage and data management in the Amazon cloud. Taking the logical connectivity graph extracted from OGC IndoorGML into consideration, a filtering algorithm was proposed to clean up the trajectory data. Using the proposed filtering algorithm in the real-world experiment, 73.53 percent of the semantically invalid trajectory points were detected and filtered. In order to further evaluate the performance of the proposed data model, three simulated datasets were generated with 200, 2000, and 20,000 users and a logical connectivity graph in dual spaces considered. The evaluation results of contact tracing applications in both real-world and simulated experiments illustrated that the pro-

posed graph-based data model could be effectively applied even for the most complicated contact tracing queries. The average query execution time of all of the contact tracing applications in the real-world experiments was less than five milliseconds with an average standard deviation of less than one millisecond. However, the average query execution time increased when the number of nodes in the simulated experiments increased.

For this research, the COVID-19 contact tracing application is selected to evaluate the proposed data model's functionality in indoor environments. The effectiveness of a digital contact tracing system depends on various factors such as public adoption [80]. Different factors such as privacy, the government's level of enforcement to use the system, and transparency in data storage and re-use can influence the public adoption of a digital contact tracing system [33,80]. Although assessing the effectiveness of digital contact tracing systems is out of our research scope, evaluating the success of contact tracing systems is required for future pandemics. Additionally, we focused only on indoor environments as they are the most complicated type of physical environments. A seamless positioning system providing seamless outdoor and indoor location information could be an exciting topic for future study. The proposed filtering algorithm in this research detects semantically invalid trajectory points but cannot improve the trajectories using possible logical connection. Further research is required to develop a trajectory reconstruction approach based on the IndoorGML connectivity graph, beacon coverage, and traverse time between cell spaces. In this research, a BLE-based proximity positioning system is deployed in an indoor environment to determine users' location for indoor spatiotemporal trajectories. Environmental factors such as indoor furniture cause reflecting and blocking the signal and impose inaccuracies on proximity estimations. So, evaluating the accuracy provided by the proximity positioning system is on hold for future work. User privacy and data secrecy protection is another direction for future research as well, especially in relation to user privacy in the cloud. In order to apply the proposed contact tracing application to a large-scale product that can be adopted by the public, detailed, scalable user privacy research needs to be conducted. Although privacy protection is outside this paper's scope, basic authentication and security authorization preserving techniques and user ID anonymization were applied to the proposed contact tracing application. Various user contexts (e.g., cleaning activities and job type) can be automatically extracted without human intervention [68,81]. Although user contexts are manually selected in this research, investigating automatic context extraction approaches can improve the scalability of the proposed systems and is on hold for future work.

Supplementary Materials: The following are available online at <https://www.mdpi.com/2220-9964/10/1/2/s1>. Video S1: Live trajectory data visualization in SensorUp Explorer and Amazon DynamoDB.

Author Contributions: Soroush Ojagh designed and developed the hierarchical graph-based data model of the proposed contact tracing system. He implemented and deployed the indoor trajectory preprocessing algorithm. He designed and performed real-world and simulated experiments. Prof. Steve H. L. Liang supervised this project, contributed to the architecture design, and reviewed experimental results. Dr. Sara Saeedi helped supervise the project, contributed to experimental design and IoT computational framework. Soroush Ojagh took the lead in writing the manuscript. All authors provided critical feedback and helped shape the research, analysis, and manuscript. Also, all authors discussed the results and contributed to the final findings. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors would like to acknowledge Jeremy Squires, Senior Software Developer at SensorUp Inc., for setting up the back-end cloud data management and SensorUp Explorer. We would also like to thank Sepehr Sabour, Graduate Researcher at the University of Calgary, for his great advice on graph databases.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A.

Considering the importance of disinfecting activities to reduce the risk of being exposed to the virus [4], individuals are strongly recommended by health organizations to sanitize the immediate space around them after each use [82,83]. As an example scenario, the importance of disinfecting activities in common areas and public spaces such as lobbies is shown in Figure A1. In Figure A1a the transmission of SARS-CoV-2-laden droplets from an infected host is shown. As illustrated in Figure A1b the immediate vicinity of the infected host is contaminated by droplets and then sanitized using a disinfectant wipe in Timestamp2. However, as shown in Figure A1c there are surfaces that are still contaminated and allow the susceptible host to be exposed by the virus in Timestamp3. As shown in Figure A1d the well-trained cleaning staff disinfects the contaminated objects using the right equipment (e.g., electrostatic spray) in Timestamp4. As seen in this scenario, disinfecting activities and temporal sequence of visiting common areas are required to be considered in digital contact tracing. In this example, considering the disinfecting activities and temporal sequence of visiting a common area, the susceptible host needs to be notified by the digital contact tracing system. If we assume that Timestamp4 (i.e., Figure A1d) occurred before Timestamp3 (i.e., Figure A1c) in the example scenario, there is no need to notify the susceptible host in the digital contact tracing system.

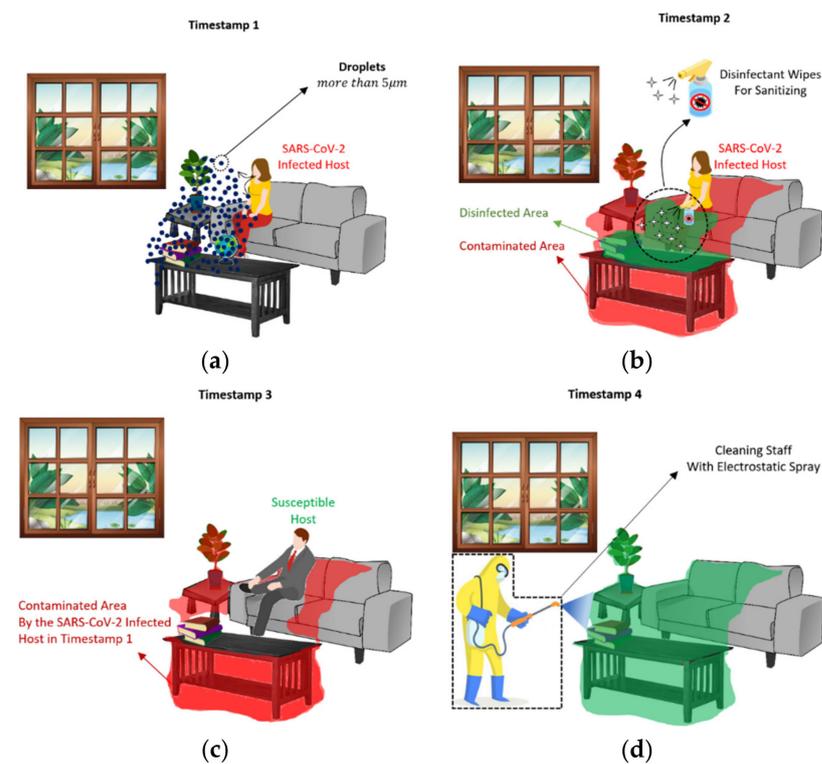


Figure A1. The impact of disinfecting activities to prevent further spread of the SARS-CoV-2 virus: (a) Showing transmission of SARS-CoV-2-laden droplets caused by an infected host; (b) Showing possibly contaminated surfaces in the immediate vicinity of the infected host and disinfected areas after sanitizing; (c) Showing objects that are still contaminated and allow the susceptible host to be exposed by the virus, and (d) Showing disinfected area after sanitizing contaminated objects by cleaning staff.

Appendix B.

In the context of COVID-19 spread, there might exist many situations where individuals are located close to each other while physically separated by obstructions such as walls and glasses. As shown in Figure A2, two individuals are located close to each other but physically separated (i.e., a user is inside a building while the other is in a bus

stop). Considering SARS-CoV-2 transmission ways, physical obstructions in between users can stop virus transmission. Existing obstructions in between BLE beacon and receiver (e.g., smartphone) attenuate the radio signal. For more information on signal attenuation and the impact of different materials on RSSI values, interested readers can refer to the study conducted by Çaliş et al. [84]. As shown in Figure A2, reduced signal strength in BLE technology can represent existing obstructions between users. In contrast, GNSS cannot consider existing physical obstructions among individuals [17].

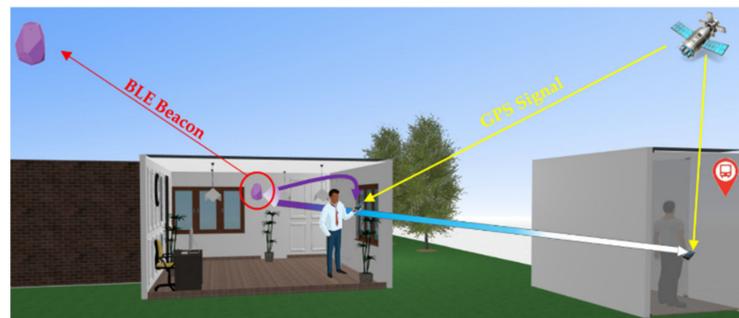


Figure A2. Representation of the difference between BLE and GNSS technology to consider physical obstructions among; The gradient color schematically illustrates radio signal strength in BLE technology (i.e., purple and white colors represent the strongest and weakest radio signal strength).

Appendix C.

A JSON payload showing a single POS record captured by the developed smartphone app and received in the cloud AWS IoT Core is shown in Figure A3.

```
{
  "geojson": {
    "geometry": {
      "coordinates": [
        -114.11484917,
        51.07921792,
        8.6
      ],
      "type": "Point"
    },
    "properties": {
      "JobType": "Cleaning",
      "UserID": "3029936e-e185-412c-a544-b0ef85849f1a",
      "VulnerabilityLevel": 8,
      "ActivityType": "Working",
      "BeaconCellName": "361-4",
      "BeaconID": "IBKS10501000363638",
      "BeaconLatitude": 51.0792179,
      "BeaconLongitude": -114.11484917,
      "BeaconAltitude": 8.6,
      "EntranceDateTime": "2020-07-25T00:42:57.461Z",
      "EntranceGPSTime": 1911602559461,
      "Duration": 331,
      "ExitDateTime": "2020-07-25T00:48:28.461Z",
      "ExitGPSTime": 1911602890461,
      "RSSI": -65
    },
    "type": "Feature"
  }
}
```

Figure A3. A single JavaScript Object Notation (JSON) payload showing the Place of Stay (POS) record of one of the users received in the Amazon Web Services (AWS) IoT Core.

Appendix D.

Related experimental images for this research are shown in this Appendix.

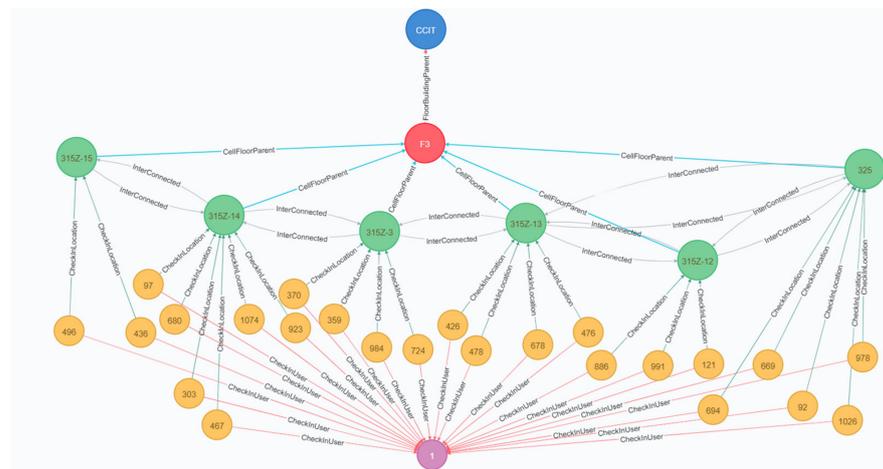


Figure A4. An example of user trajectory modelled by the proposed graph-based semantic indoor trajectory in a real-world experiment using Neo4j graph database for user a single user (i.e., user 1). Nodes are illustrated as circles with different colors: Blue for building, red for floor, green for cells, yellow for trajectory (the number shows the duration of time in seconds for each check-in), and purple for user nodes.

Appendix E.

This Appendix is focused on the Cypher code representation of the indoor trajectory queries in the Neo4j graph database.

Appendix E.1. Query 1

The Cypher code of Query 1 and execution of this query in Neo4j is represented in this subsection.

Cypher Code

```
MATCH (c:Cell)-[[]]-(ch:CheckIn)-[[]]-(u:User)
WHERE u.UserID = '2' AND ch.Duration > 900 //Duration of 15 minutes
MATCH (c)-[:CellFloorParent]->(f:Floor)
MATCH (f)-[:FloorBuildingParent]->(b:Building)
RETURN DISTINCT c.CellID AS Infected_Geospatial_Zone, f.FloorID AS Infected_Floor,
b.BuildingID AS Infected_Building
```

Execution in the Neo4j with 551 Nodes

"Infected_Geospatial_Zone"	"Infected_Floor"	"Infected_Building"
"315Z-4"	"F3"	"CCIT"
"315Z-6"	"F3"	"CCIT"
"315Z-15"	"F3"	"CCIT"
"315Z-14"	"F3"	"CCIT"
"315Z-8"	"F3"	"CCIT"
"315Z-3"	"F3"	"CCIT"
"315Z-16"	"F3"	"CCIT"
"315Z-9"	"F3"	"CCIT"
"315Z-7"	"F3"	"CCIT"

Figure A5. The results of executing Query 1 in the Neo4j graph database with 551 nodes and 1149 relationships: the first column of results shows contaminated indoor cells, while the second and third columns show the floor and building in the hierarchical graph data model.

Appendix E.2. Query 3

The Cypher code of Query 3 is represented in this subsection.

Cypher Code

```
MATCH (c:Cell)-[:]-[:](ch:CheckIn)-[:]-[:](u:User)
WHERE u.UserHealthStatus = 'COVID19' AND ch.ExitTime > 1911608974461 AND ch.ExitTime < 1911610721461 //GPS Epoch
Time of 2020-07-25T02:29:52.461Z and 2020-07-25T02:58:59.461Z
MATCH (c)-[:CellFloorParent]-[:](f:Floor)
MATCH (f)-[:FloorBuildingParent]-[:](b:Building)
RETURN DISTINCT c.CellID AS Infected_Geospatial_Zone, f.FloorID AS Infected_Floor, b.BuildingID AS Infected_Building
```

E.3. Query 5

The Cypher code of Query 5 is represented in this subsection.

Cypher Code

```
MATCH (c:Cell)-[:]-[:](ch:CheckIn)-[:]-[:](u:User)
MATCH (u2:User)-[:]-[:](ch2:CheckIn)-[:]-[:](c2:Cell)
WHERE u.UserHealthStatus = 'COVID19' AND u <> u2 AND c = c2 AND ((ch.EntranceTime > ch2.EntranceTime and
ch.EntranceTime < ch2.ExitTime) OR (ch2.EntranceTime > ch.EntranceTime AND ch2.EntranceTime < ch.ExitTime))
AND(ch.Duration*1000 + ch2.Duration*1000 - ABS(ch2.EntranceTime-ch.EntranceTime)-ABS(ch2.ExitTime-ch.ExitTime))/
2 > 900*1000
RETURN DISTINCT u.UserID AS COVID19_User,u2.UserID AS Possibly_Infected_User,c2.CellID AS Infected_Geospatial_Zone
```

E.4. Query 7

The Cypher code of Query 7 is represented in this subsection.

Cypher Code

```
MATCH (c:Cell)-[:]-[:](ch:CheckIn)-[:]-[:](u:User)
MATCH (c2:Cell)-[:]-[:](ch2:CheckIn)-[:]-[:](u2:User{JobType:"Cleaning_Staff"})
WHERE u.UserHealthStatus = 'COVID19' AND u <> u2 AND c = c2 AND ch.ExitTime < ch2.EntranceTime
MATCH (c)-[:CellFloorParent]-[:](f:Floor)
MATCH (f)-[:FloorBuildingParent]-[:](b:Building)
RETURN DISTINCT c.CellID AS Infected_Geospatial_Zone, f.FloorID AS Infected_Floor,b.BuildingID AS Infected_Building,
u.UserID AS COVID19_User
```

E.5. Query 8

The Cypher code of Query 8 is represented in this subsection.

Cypher Code

```
MATCH (c:Cell)-[:]-[:](ch:CheckIn)-[:]-[:](u:User)
MATCH (u2:User)-[:]-[:](ch2:CheckIn)-[:]-[:](c2:Cell)
MATCH (uc:User{JobType: "Cleaning_Staff"})-[:]-[:](ch3:CheckIn)-[:]-[:](c3:Cell)
WHERE u.UserHealthStatus = 'COVID19' AND NOT u2.JobType = "Cleaning_Staff" AND NOT u2.UserHealthStatus = "COVID19"
AND u <> u2 AND u2 <> uc AND c = c2 AND c = c3 AND NOT (ch3.EntranceTime < ch.EntranceTime AND ch3.ExitTime <
ch2.EntranceTime)RETURN u2.UserID As Possibly_Infected_Users
```

E.6. Semantically Valid Trajectory Extraction

The applied Algorithm 3 to extract the semantically valid indoor movement trajectory is represented in this section.

Algorithm A1: Extracting semantically valid indoor movement trajectory

Input:

A temporal sequence of indoor cells c for a user as a moving object having size = n :

$S = [\langle c_0, t_0 \rangle, \langle c_1, t_1 \rangle, \dots, \langle c_n, t_n \rangle]$ in which $t_0 < t_1 < \dots < t_n$

The adjacency matrix $M_{j \times j}$ showing connectivity between indoor cells in which j is the size of indoor cell:

$$M_{i,j} = \begin{cases} 1 & \text{if there is an inter-layer connection between cell } c_i \text{ and } c_j \\ 0 & \text{otherwise} \end{cases}$$

Output: Semantically valid trajectory \hat{S} having size = m

Initialize:

$\hat{S} \leftarrow []$

$\hat{S}_0 \leftarrow S_0$

counter $\leftarrow 1$

while counter <

if $M_{\hat{S}[m-1] \times S[\text{counter}]} == 1$

$\hat{S}[m] \leftarrow S[\text{counter}]$

counter ++

End-while

Return \hat{S}

Table A1. Details of query execution time for trajectory queries in Neo4j graph database with different sizes.

Spatiotemporal Queries	No. of Nodes	Query Execution Time in Milliseconds		
		Minimum	Average	Standard Deviation
Query 1	551	1.00	2.02	7.74×10^{-1}
	5058	2.00	3.82	1.33
	48,826	4.00	9.05	3.13
	473,683	4.00	1.36×10	5.96
Query 2	551	1.00	3.05	1.46
	5058	5.00	7.90	2.13
	48,826	2.20×10	3.55×10	7.72
	473,683	3.11×10^2	3.49×10^2	2.01×10
Query 3	551	1.00	1.49	5.00×10^{-1}
	5058	3.00	4.91	1.37
	48,826	2.00	1.51×10	8.10
	473,683	2.00	3.12×10	1.50×10
Query 4	551	2.00	2.98	8.24×10^{-1}
	5058	3.00	5.02	1.17
	48,826	6.30×10	3.52×10	1.77×10
	473,683	3.61×10^2	4.28×10^2	3.47×10
Query 5	551	3.00	5.02	1.46
	5058	8.00	1.07×10	1.73
	48,826	1.60×10	9.02×10	1.06×10
	473,683	1.04×10^2	1.46×10^2	2.53×10
Query 6	551	3.00	3.43	4.95×10^{-1}
	5058	1.00×10	1.29×10	2.05
	48,826	4.80×10	7.24×10	1.42×10
	473,683	4.96×10^2	5.29×10^2	1.85×10
Query 7	551	1.70×10	1.85×10	1.06
	5058	2.10×10	2.44×10	2.41
	48,826	4.20×10	5.58×10	8.23
	473,683	6.00×10	9.01×10	1.83×10
Query 8	551	2.00	3.00	8.37×10^{-1}
	5058	3.10×10	3.62×10	3.54
	48,826	2.30×10^2	2.58×10^2	1.81×10
	473,683	9.19×10^3	1.03×10^4	6.26×10

References

1. Whitelaw, S.; Mamas, M.A.; Topol, E.; Van Spall, H.G. Applications of digital technology in COVID-19 pandemic planning and response. *Lancet Digit. Health* **2020**, *2*, e435–e440. [CrossRef]
2. Guinchar, A. Our digital footprint under Covid-19: Should we fear the UK digital contact tracing app? *Int. Rev. Law Comput. Technol.* **2020**, 1–14. [CrossRef]
3. Kampf, G.; Lemmen, S.; Suchomel, M. Ct values and infectivity of SARS-CoV-2 on surfaces. *Lancet Infect. Dis.* **2020**. [CrossRef]
4. Mondelli, M.U.; Colaneri, M.; Seminari, E.M.; Baldanti, F.; Bruno, R. Low risk of SARS-CoV-2 transmission by fomites in real-life conditions. *Lancet Infect. Dis.* **2020**. [CrossRef]
5. Medicine, T.L.R. COVID-19 transmission—Up in the air. *Lancet. Respir. Med.* **2020**, *8*, 1159. [CrossRef]
6. World Health Organization. Coronavirus Disease (COVID-19): How Is It Transmitted? Available online: <https://www.who.int/news-room/q-a-detail/coronavirus-disease-covid-19-how-is-it-transmitted> (accessed on 12 December 2020).
7. Morawska, L.; Cao, J. Airborne transmission of SARS-CoV-2: The world should face the reality. *Environ. Int.* **2020**, *139*, 105730. [CrossRef] [PubMed]
8. Van Doremalen, N.; Bushmaker, T.; Morris, D.H.; Holbrook, M.G.; Gamble, A.; Williamson, B.N.; Tamin, A.; Harcourt, J.L.; Thornburg, N.J.; Gerber, S.I. Aerosol and surface stability of SARS-CoV-2 as compared with SARS-CoV-1. *N. Engl. J. Med.* **2020**, *382*, 1564–1567. [CrossRef]
9. He, H.; Li, R.; Wang, R.; Bao, J.; Zheng, Y.; Li, T. Efficient Suspected Infected Crowds Detection Based on Spatio-Temporal Trajectories. *arXiv*, 2020; arXiv:2004.06653.
10. Riemer, K.; Ciriello, R.; Peter, S.; Schlagwein, D. Digital contact-tracing adoption in the COVID-19 pandemic: IT governance for collective action at the societal level. *Eur. J. Inf. Syst.* **2020**, 1–15. [CrossRef]
11. Ahmed, N.; Michelin, R.A.; Xue, W.; Ruj, S.; Malaney, R.; Kanhere, S.S.; Seneviratne, A.; Hu, W.; Janicke, H.; Jha, S.K. A survey of covid-19 contact tracing apps. *IEEE Access* **2020**, *8*, 134577–134601. [CrossRef]
12. Braithwaite, I.; Callender, T.; Bullock, M.; Aldridge, R.W. Automated and partly automated contact tracing: A systematic review to inform the control of COVID-19. *Lancet Digit. Heal.* **2020**, *2*, e607–e621. [CrossRef]
13. Kumar, K.; Kumar, N.; Shah, R. Role of IoT to avoid spreading of COVID-19. *Int. J. Intell. Netw.* **2020**, *1*, 32–35. [CrossRef]
14. OpenTrace. OpenTrace. Available online: <https://github.com/opentrace-community> (accessed on 3 October 2020).
15. CovidSafe. CovidSafe. Available online: <https://github.com/AU-COVIDSafe> (accessed on 3 October 2020).
16. PACT. East Coast. Available online: <https://pact.mit.edu/> (accessed on 3 October 2020).
17. Berke, A.; Bakker, M.; Vepakomma, P.; Raskar, R.; Larson, K.; Pentland, A. Assessing disease exposure risk with location data; A proposal for cryptographic preservation of privacy. *arXiv*, 2020; arXiv:2003.14412.
18. Morawska, L.; Milton, D.K. It is time to address airborne transmission of COVID-19. *Clin. Infect. Dis.* **2020**, *6*, ciae939. [CrossRef]
19. Jensen, C.S.; Lu, H.; Yang, B. Graph model based indoor tracking. In Proceedings of the 2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware, Taipei, Taiwan, 18–20 May 2009; pp. 122–131.
20. Alattas, A.; van Oosterom, P.; Zlatanova, S.; Hoeneveld, D.; Verbree, E. LADM-IndoorGML for exploring user movements in evacuation exercise. *Land Use Policy* **2020**, *98*, 104219. [CrossRef]
21. Gu, F.; Valaee, S.; Khoshelham, K.; Shang, J.; Zhang, R. Landmark Graph-based Indoor Localization. *IEEE Internet Things J.* **2020**, *7*, 8343–8355. [CrossRef]
22. Li, K.-J.; Conti, G.; Konstantinidis, E.; Zlatanova, S.; Bamidis, P. OGC IndoorGML: A standard approach for indoor maps. In *Geographical and Fingerprinting Data to Create Systems for Indoor Positioning and Indoor/Outdoor Navigation*; Elsevier: Amsterdam, The Netherlands, 2019; pp. 187–207.
23. Centers for Disease Control and Prevention. Cleaning and Disinfection for Households Interim Recommendations for US Households with Suspected or Confirmed Coronavirus Disease 2019 (COVID-19). Available online: <https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/cleaning-disinfection.html> (accessed on 28 March 2020).
24. Kontarinis, A.; Zeitouni, K.; Marinica, C.; Vodislav, D.; Kotzinos, D. Towards a Semantic Indoor Trajectory Model. 2019. Available online: <https://hal.archives-ouvertes.fr/hal-02314572/> (accessed on 28 March 2020).
25. Zheng, Y. Trajectory data mining: An overview. *ACM Trans. Intell. Syst. Technol. (TIST)* **2015**, *6*, 1–41. [CrossRef]
26. Parent, C.; Spaccapietra, S.; Renso, C.; Andrienko, G.; Andrienko, N.; Bogorny, V.; Damiani, M.L.; Gkoulalas-Divanis, A.; Macedo, J.; Pelekis, N. Semantic trajectories modeling and analysis. *ACM Comput. Surv. (CSUR)* **2013**, *45*, 1–32. [CrossRef]
27. Gómez, L.I.; Kuijpers, B.; Vaisman, A.A. Analytical queries on semantic trajectories using graph databases. *Trans. GIS* **2019**, *23*, 1078–1101. [CrossRef]
28. Spanier, E.H. *Algebraic Topology*; Springer Science & Business Media: New York, NY, USA, 1989.
29. Montjoye, Y.-A.; Radaelli, L.; Singh, V.; Pentland, A. Unique in the shopping mall: On the reidentifiability of credit card metadata. *Science* **2015**, *347*, 536–539. [CrossRef]
30. Zhuang, Y.; Yang, J.; Li, Y.; Qi, L.; El-Sheimy, N. Smartphone-based indoor localization with bluetooth low energy beacons. *Sensors* **2016**, *16*, 596. [CrossRef] [PubMed]
31. Andrushchak, V.; Maksymyuk, T.; Klymash, M.; Ageyev, D. Development of the iBeacon’s Positioning Algorithm for Indoor Scenarios. In Proceedings of the 2018 International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T), Kharkiv, Ukraine, 9–12 October 2018; pp. 741–744.

32. Ramadhan, H.; Yustiawan, Y.; Kwon, J. Applying Movement Constraints to BLE RSSI-Based Indoor Positioning for Extracting Valid Semantic Trajectories. *Sensors* **2020**, *20*, 527. [[CrossRef](#)] [[PubMed](#)]
33. Simko, L.; Chang, J.L.; Jiang, M.; Calo, R.; Roesner, F.; Kohno, T. COVID-19 Contact Tracing and Privacy: A Longitudinal Study of Public Opinion. *arXiv*, 2020; arXiv:2012.01553.
34. Bianconi, A.; Marcelli, A.; Campi, G.; Perali, A. Efficiency of COVID-19 mobile contact tracing containment by measuring time-dependent doubling time. *Phys. Biol.* **2020**, *17*, 065006. [[CrossRef](#)]
35. Ferretti, L.; Wymant, C.; Kendall, M.; Zhao, L.; Nurtay, A.; Abeler-Dörner, L.; Parker, M.; Bonsall, D.; Fraser, C. Quantifying SARS-CoV-2 transmission suggests epidemic control with digital contact tracing. *Science* **2020**, *368*, eabb6936. [[CrossRef](#)]
36. Hellewell, J.; Abbott, S.; Gimma, A.; Bosse, N.I.; Jarvis, C.I.; Russell, T.W.; Munday, J.D.; Kucharski, A.J.; Edmunds, W.J.; Sun, F. Feasibility of controlling COVID-19 outbreaks by isolation of cases and contacts. *Lancet Global Health* **2020**, *8*, e488–e496. [[CrossRef](#)]
37. TeamSense. Empower the Deskless Members of Your Team. Available online: <https://www.teamsense.com/features> (accessed on 14 December 2020).
38. Government of Canada. Download COVID Alert Today. Available online: <https://www.canada.ca/en/public-health/services/diseases/coronavirus-disease-covid-19/covid-alert.html> (accessed on 14 December 2020).
39. Government of Alberta. ABTraceTogether. Available online: <https://www.alberta.ca/ab-trace-together.aspx> (accessed on 14 December 2020).
40. Estimote. Workplace Safety with Wearables. Available online: https://estimote.com/wearable/?gclid=CjwKCAiAiML-BRAAEiwAuWVgkjsbosByEODh11g7RBWrxIJ-XSIP5oGOfKhZ3z0F01_ONi9rKAUsBoCyF8QAvD_BwE (accessed on 14 December 2020).
41. Stevens, H.; Haines, M.B. TraceTogether: Pandemic Response, Democracy, and Technology. *East Asian Sci. Technol. Soc. Int. J.* **2020**, *14*, 523–532. [[CrossRef](#)]
42. SensorTower. TraceTogether. Available online: <https://sensortower.com/ios/SG/government-technology-agency/app/tracetgether/1498276074/overview> (accessed on 12 December 2020).
43. Worldometer. Countries in the World by Population. 2020. Available online: <https://www.worldometers.info/world-population/population-by-country/> (accessed on 12 December 2020).
44. Government of Singapore. TraceTogether, Safer Together. Available online: <https://www.tracetgether.gov.sg/> (accessed on 12 December 2020).
45. Corona-Warn-App. Corona-Warn-App Open Source Project. Available online: <https://www.coronawarn.app/en/> (accessed on 12 December 2020).
46. SensorTower. Corona Warning App. Available online: <https://sensortower.com/ios/de/robert-koch-institut/app/corona-warn-app/1512595757/overview> (accessed on 12 December 2020).
47. Shubina, V.; Holcer, S.; Gould, M.; Lohan, E.S. Survey of Decentralized Solutions with Mobile Devices for User Location Tracking, Proximity Detection, and Contact Tracing in the COVID-19 Era. *Data* **2020**, *5*, 87. [[CrossRef](#)]
48. Lewis, D.M. Coepi: Community Epidemiology in Action. 2020. Available online: <https://www.coepi.org/> (accessed on 12 December 2020).
49. Fenwick, R.; Hittle, M.; Ingle, M.; Nash, O.; Nguyen, V.; Petrie, J.; Schwaber, J.; Szabo, Z.; Veeraghanta, A.; Voloshin, M. Sydney Von Arx, and Tina White. Covid Watch. 2020. Available online: <https://www.covidwatch.org/> (accessed on 12 December 2020).
50. Rong, C.; Lin, C.; Silva, Y.N.; Wang, J.; Lu, W.; Du, X. Fast and scalable distributed set similarity joins for big data analytics. In Proceedings of the 2017 IEEE 33rd International Conference on Data Engineering (ICDE), San Diego, CA, USA, 19–22 April 2017; pp. 1059–1070.
51. Centre for the Mathematical Modelling of Infectious Diseases. Estimates for Singapore. Available online: <https://epiforecasts.io/covid/posts/national/singapore/> (accessed on 12 December 2020).
52. Centre for the Mathematical Modelling of Infectious Diseases. National and Subnational Estimates for Germany. Available online: <https://epiforecasts.io/covid/posts/national/germany/> (accessed on 12 December 2020).
53. Krumm, J.; Horvitz, E. Predestination: Inferring destinations from partial trajectories. In *International Conference on Ubiquitous Computing*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 243–260.
54. Lee, J.-G.; Han, J.; Whang, K.-Y. Trajectory clustering: A partition-and-group framework. In Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data, Beijing, China, 12–14 June 2007; pp. 593–604.
55. Arslan, M.; Cruz, C.; Ginhac, D. Semantic enrichment of spatio-temporal trajectories for worker safety on construction sites. *Pers. Ubiquitous Comput.* **2019**, *23*, 749–764. [[CrossRef](#)]
56. Liu, C.; Guo, C. STCCD: Semantic Trajectory Clustering based on Community Detection in Networks. *Expert Syst. Appl.* **2020**, *162*, 113689. [[CrossRef](#)]
57. Sun, Y.; Gu, T.; Bin, C.; Chang, L.; Kuang, H.; Huang, Z.; Sun, L. A multi-latent semantics representation model for mining tourist trajectory. In *Pacific Rim International Conference on Artificial Intelligence*; Springer: Cham, Switzerland, 2018; pp. 463–476.
58. Cai, G.; Lee, K.; Lee, I. Itinerary recommender system with semantic trajectory pattern mining from geo-tagged photos. *Expert Syst. Appl.* **2018**, *94*, 32–40. [[CrossRef](#)]

59. Nardini, F.M.; Orlando, S.; Perego, R.; Raffaetà, A.; Renso, C.; Silvestri, C. Analysing trajectories of mobile users: From data warehouses to recommender systems. In *A Comprehensive Guide through the Italian Database Research Over the Last 25 Years*; Springer: Cham, Switzerland, 2018; pp. 407–421.
60. Yuan, N.J.; Zheng, Y.; Zhang, L.; Xie, X. T-finder: A recommender system for finding passengers and vacant taxis. *EEE Trans. Knowl. Data Eng.* **2012**, *25*, 2390–2403. [[CrossRef](#)]
61. Werner, M.; Schauer, L.; Scharf, A. Reliable trajectory classification using Wi-Fi signal strength in indoor scenarios. In Proceedings of the 2014 IEEE/ION Position, Location and Navigation Symposium-PLANS 2014, Monterey, CA, USA, 5–8 May 2014; pp. 663–670.
62. Guo, S.; Xiong, H.; Zheng, X. A novel semantic matching method for indoor trajectory tracking. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 197. [[CrossRef](#)]
63. Wang, R.; Shroff, R.; Zha, Y.; Seshan, S.; Veloso, M. Indoor trajectory identification: Snapping with uncertainty. In Proceedings of the 2015 IEEE/RISJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 4901–4906.
64. Chen, Y.; Yuan, P.; Qiu, M.; Pi, D. An indoor trajectory frequent pattern mining algorithm based on vague grid sequence. *Expert Syst. Appl.* **2019**, *118*, 614–624. [[CrossRef](#)]
65. Li, H.; Lu, H.; Chen, X.; Chen, G.; Chen, K.; Shou, L. Vita: A versatile toolkit for generating indoor mobility data for real-world buildings. *Proc. VLDB Endow.* **2016**, *9*, 1453–1456. [[CrossRef](#)]
66. Alattas, A.; Zlatanova, S.; Van Oosterom, P.; Chatzinikolaou, E.; Lemmen, C.; Li, K.-J. Supporting indoor navigation using access rights to spaces based on combined use of IndoorGML and LADM models. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 384. [[CrossRef](#)]
67. Ojagh, S.; Malek, M.R.; Saeedi, S.; Liang, S. A location-based orientation-aware recommender system using IoT smart devices and Social Networks. *Future Gener. Comput. Syst.* **2020**, *108*, 97–118. [[CrossRef](#)]
68. Ojagh, S.; Malek, M.R.; Saeedi, S. A Social-Aware Recommender System Based on User’s Personal Smart Devices. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 519. [[CrossRef](#)]
69. Zheng, V.W.; Zheng, Y.; Xie, X.; Yang, Q. Towards mobile intelligence: Learning from GPS history data for collaborative recommendation. *Artif. Intell.* **2012**, *184*, 17–37. [[CrossRef](#)]
70. Cauteruccio, F.; Cinelli, L.; Corradini, E.; Terracina, G.; Ursino, D.; Virgili, L.; Savaglio, C.; Liotta, A.; Fortino, G. A framework for anomaly detection and classification in Multiple IoT scenarios. *Future Gener. Comput. Syst.* **2020**, *114*, 322–335. [[CrossRef](#)]
71. Cauteruccio, F.; Cinelli, L.; Fortino, G.; Savaglio, C.; Terracina, G.; Ursino, D.; Virgili, L. An approach to compute the scope of a social object in a Multi-IoT scenario. *Pervasive Mobile Comput.* **2020**, *67*, 101223. [[CrossRef](#)]
72. Hu, F.; Li, Z.; Yang, C.; Jiang, Y. A graph-based approach to detecting tourist movement patterns using social media data. *Cartogr. Geogr. Inf. Sci.* **2018**, *46*, 368–382. [[CrossRef](#)]
73. Niu, X.; Chen, T.; Wu, C.Q.; Niu, J.; Li, Y. Label-based trajectory clustering in complex road networks. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 4098–4110. [[CrossRef](#)]
74. Sabarish, B.; Karthi, R.; Kumar, T.G. Graph Similarity-based Hierarchical Clustering of Trajectory Data. *Procedia Comput. Sci.* **2020**, *171*, 32–41. [[CrossRef](#)]
75. Kang, H.-K.; Li, K.-J. A standard indoor spatial data model—OGC IndoorGML and implementation approaches. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 116. [[CrossRef](#)]
76. Lee, J.; Li, K.-J.; Zlatanova, S.; Kolbe, T.H.; Nagel, C.; Becker, T.; Ogc Indoorgml. Open Geospatial Consortium Standard. 2014. Available online: <https://www.ogc.org/standards/indoorgml> (accessed on 18 December 2020).
77. Kassir, R. Risk of COVID-19 for Patients with Obesity. *Obes. Rev.* **2020**, *21*, e13034. [[CrossRef](#)]
78. England, P.H. Guidance to Assist Professionals in Advising the General Public. Available online: <https://www.gov.uk/government/publications/novel-coronavirus-2019-ncov-guidance-to-assist-professionals-in-advising-the-general-public/guidance-to-assist-professionals-in-advising-the-general-public> (accessed on 18 December 2020).
79. de Silva, F. Execution Time Analysis of Electrical Network Tracing in Relational and Graph Databases. 2019. Available online: <http://www.diva-portal.se/smash/get/diva2:1304968/FULLTEXT01.pdf> (accessed on 18 December 2020).
80. Dar, A.B.; Lone, A.H.; Zahoor, S.; Khan, A.A.; Naaz, R. Applicability of mobile contact tracing in fighting pandemic (COVID-19): Issues, challenges and solutions. *Comput. Sci. Rev.* **2020**, *38*, 100307. [[CrossRef](#)]
81. Ojagh, S.; Malek, M.R.; Saeedi, S.; Liang, S. An Internet of Things (IoT) Approach for Automatic Context Detection. In Proceedings of the 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 1–3 November 2018; pp. 223–226.
82. Çaliş, G.; Becerik-Gerber, B.; Göktepe, A.B.; Shuai, L.; Nan, L. Analysis of the variability of RSSI values for active RFID-based indoor applications. *Turk. J. Eng. Environ. Sci.* **2013**, *37*, 186–211. [[CrossRef](#)]
83. Centers for Disease Control and Prevention. Cleaning and Disinfecting Your Facility. 2020. Available online: <https://www.cdc.gov/coronavirus/2019-ncov/community/disinfecting-building-facility.html> (accessed on 13 December 2020).
84. WHO. Coronavirus Disease (COVID-19): Cleaning and Disinfecting Surfaces in Non-Health Care Settings. 2020. Available online: https://www.researchgate.net/profile/Nan_Li4/publication/258332276_Analysis_of_the_variability_of_RSSI_values_for_active_RFID-based_indoor_applications/links/0a85e530a4adabc3f5000000/Analysis-of-the-variability-of-RSSI-values-for-active-RFID-based-indoor-applications.pdf (accessed on 18 December 2020).