IndoorViz: A Demonstration System for Indoor Spatial Data Management

Yue Li
East China Normal University
Shanghai, China
51194501052@stu.ecnu.edu.cn

Shiyu Yang
Guangzhou University
Guangzhou, China
shiyu.yang@aliyun.com

Muhammad Aamir Cheema
Monash University
Melbourne, Australia
aamir.cheema@monash.edu

Zhou Shao
Monash University
Melbourne, Australia
joe.shao@monash.edu

Xuemin Lin
The University of New South Wales
Sydney, Australia
lxue@cse.unsw.edu.au

ABSTRACT

Due to the growing popularity of indoor location-based services, indoor data management has received significant research attention in the past few years. However, we observe that the existing indexing and query processing techniques for the indoor space do not fully exploit the properties of the indoor space. Consequently, they provide below par performance which makes them unsuitable for large indoor venues with high query workloads. In this demonstration, we present *IndoorViz* , a new indoor spatial data management system that integrates three novel index structures proposed in [4] and [6] with well designed query processing algorithms and 3D visualization functions. The *IndoorViz* is able to support indoor spatial object indexing, efficient query processing and interactive 3D display.

CCS CONCEPTS

• **Information systems → Location based services**; *Database query processing*; *Query optimization.*

KEYWORDS

Indoor Location-based Service, Indoor Index, Indoor Query Processing

ACM Reference Format:
Yue Li, Shiyu Yang, Muhammad Aamir Cheema, Zhou Shao, and Xuemin Lin. 2021. IndoorViz: A Demonstration System for Indoor Spatial Data Management . In *Proceedings of the 2021 International Conference on Management of Data (SIGMOD '21), June 20–25, 2021, Virtual Event, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3448016.3452761

1 INTRODUCTION

People spend a large part of their lives in an indoor environment, such as office buildings, shopping malls and universities. Therefore, indoor location based service (LBSs) can be very valuable in many different domains such as emergency services, healthcare, location-based marketing, asset management, and in-store navigation, to name a few. In such indoor LBSs and many others, indoor distances play a critical role in improving the service quality. For example, in an emergency, an indoor LBS can guide people to the nearby exit doors. Similarly, a passenger may want to find the shortest path to the boarding gate in an airport, a disabled person may issue a query to find accessible toilets within 100 meters in a shopping mall, or a student may issue a query to find the nearest photocopier in a university campus. Driven by recent advances in indoor location technology and popularity of indoor LBSs, there is a huge demand for efficient and scalable spatial query-processing systems for indoor location data.

However, due to topological features of indoor space, the outdoor technology can not be applied to solve indoor spatial queries. The reason is that the dominant technology such as global positioning system (GPS) only applicable to an outdoor environment. For indoor spaces, locations cannot be identified using GPS accurately. Another reason is that indoor space is totally different from a outdoor space because it consists of indoor entities such as rooms, hallways, doors and walls. Hence, the notion of shortest path in indoor is different from that in outdoor. The shortest distance between two indoor points is the minimum indoor walking distance. It is restricted by the floor plan of an indoor space. To handle fundamental spatial queries, Zhou et al. propose two novel indoor indexes called Indoor Partitioning tree (IP-Tree) and Vivid IP-Tree (VIP-Tree) that optimize the indexing by exploiting the properties of indoor spaces [4]. The basic observation is that the shortest path from a point in one indoor region to a point in another region passes through a small subset of doors (called access doors). Zhou et al. also propose a novel data structure called Keyword Partitioning Tree (KP-Tree) to effectively index the objects with textual information in a single indoor partition [6]. In this demonstration, we present *IndoorViz* which integrates the novel index structures, efficient query processing algorithms and interactive 3D display. The *IndoorViz* support four kinds of important indoor spatial queries, namely k nearest neighbor query, influence query, spatial keyword query and shortest path query. With the help of the well-designed index structures, *IndoorViz* is able to manage large scale indoor datasets and process these important queries efficiently.

SIGMOD '21, June 20–25, 2021, Virtual Event, China

ACM ISBN 978-1-4503-8343-1/21/06…$15.00
https://doi.org/10.1145/3448016.3452761

The rest of the paper is organized as follows. In Section II, we briefly review the index structures we developed for indoor spatial objects management. Then we present the overview structure of *IndoorViz* then introduce the three major modules in detailed in Section III. In section IV, we show the interface and how to interact with *IndoorViz* by four demonstration scenarios.

2 INDEX OVERVIEW

In this section, we first briefly introduce the basic of Door-to-Door(D2D) graph [1] which are widely used in indoor space representation. Then we introduce the index structures which are the key elements of our demo system, namely Vivid IP-Tree (VIP-Tree) and Keyword Partitioning tree (KP-Tree). For more technical details, please refer to our full research papers [4] and [6].

2.1 D2D graph

The indoor space usually consists of doors and partitions, such as room and hallway. A widely used representation method is the D2D graph, in which each door is represented as vertex and connectivity between doors are represented as edge. For example, if two doors i and j are connected to the same partition, then there is an edge between i and j. For each edge, we use the edge weight to represent the indoor distance between two doors. Figure 2 is the corresponding D2D graph of the indoor space shown in Figure 1. The edge weight is omitted for clarity of display.

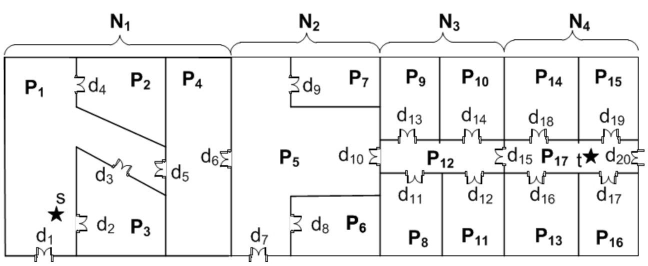

Figure 1: An indoor venue containing 17 partitions and 20 doors [4]

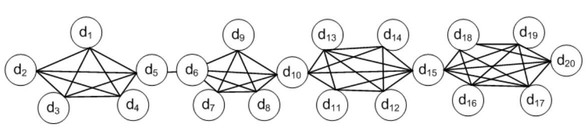

Figure 2: Door-to-Door Graph [4]

2.2 Vivid IP-Tree

Indoor Partitioning Tree (IP-Tree)[4]: IP-Tree is an indexing structure based on a D2D graph that proposed in [8]. The basic idea is to combine adjacent indoor partitions (e.g., rooms, hallways, stairs) to form leaf nodes and then iteratively combining adjacent leaf nodes until all nodes are combined into a single root node. Fig. 3 shows an IP-Tree of the indoor venue shown in Fig. 1 where the indoor space is converted into four leaf nodes (N_1 to N_4). Each leaf node consists of several indoor partitions. Specifically, $N_1 = \{P_1, \cdots, P_4\}$. The leaf nodes are iteratively merged until root node is formed, e.g., N_1 and N_2 are merged to form N_5.

Vivid IP-Tree (VIP-Tree)[4]: VIP-Tree is very similar to IP-tree except that it stores, for each door d_i in the indoor space, the following additional information. Let N be the leaf node that contains the door d_i. For every door d_j that is an access door in one of the ancestor nodes of N, VIP-tree stores $dist(d_i, d_j)$ as well as the next-hop door d_k on the shortest path from d_i to d_j. This information can be efficiently computed by our efficient shortest distance/path algorithms using IP-tree. Due to our effective indexes, efficient search and join algorithms, and cache mechanism, almost all of the queries can get the results in seconds.

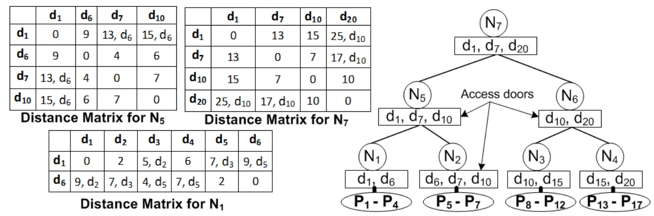

Figure 3: Indoor partitioning tree (IP-Tree) [4]

2.3 Keyword-Partitioning Tree

In order to handle spatial-textual objects in indoors space such as products in supermarket and books in library, we develop a novel index structure, named Keyword-Partition tree [6]. A KP-tree is created for each indoor partition and allows efficient retrieval of relevant objects when the search reaches a particular partition. Each node R in KP-Tree consists of a list of keywords represented as $R.\mathcal{T}$. Specifically, for every node R, $R.\mathcal{T}$ is the union of the keywords contained in its children. In KP-tree, each object o is attached with a node R if $o.\mathcal{T} = R.\mathcal{T}$. A node in KP-Tree is also linked to its pre-computed object and node matrices. An object matrix records distance from each door d of the partition to each object o attached with the node. With the help of VIP-tree and KP-tree, spatial keyword query can be processed efficiently.

3 SYSTEM ARCHITECTURE

In this section, we introduce the proposed demonstration system *IndoorViz* which are designed to index indoor spatial data and textual data, process indoor spatial queries and visualize query results on 3D floor plan. It adopts the browser-server model. The browser end provides friendly interactive functionality including displaying the indoor space, getting spatial query input and interacting with the 3D floor plan. The server end takes responsibility of indexing raw indoor data, processing indoor spatial queries and returning query results for browser to display. The framework of *IndoorViz* is shown in Figure 4. It consists of the following three parts.

- **Indoor spatial data storage and indexing module:** The storage and indexing module takes the raw floor plan data and other indoor data, such as users' locations, indoor facilities' location and keywords as input. Before building the index, we first convert the raw floor plan to D2D graph. Then

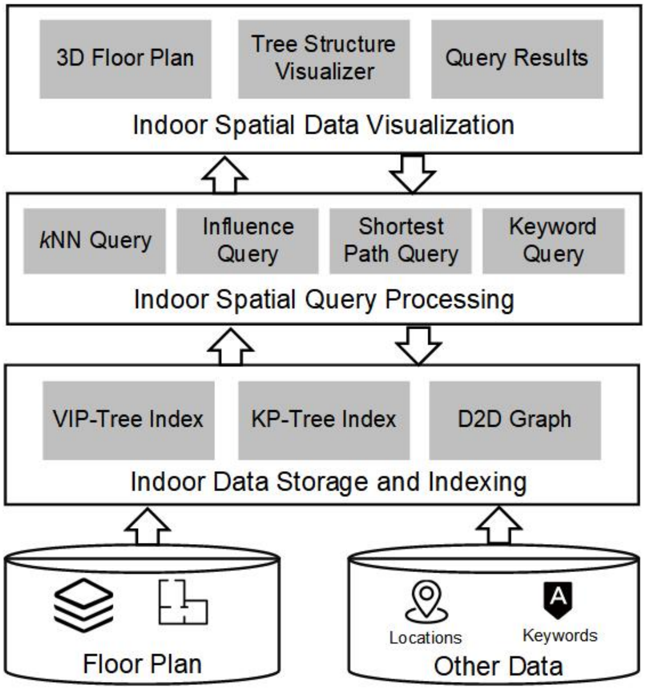

Figure 4: System framework.

we build the index according to the data type. For spatial objects, we use VIP-tree as the index structure and for textual data, KP-tree is adopted as the index structure.

- **Indoor spatial query processing module:** The query processing module is designed to efficiently process indoor spatial queries using VIP-tree and KP-tree. Currently, *IndoorViz* provides four query processors, namely, k nearest neighbor query, influence query, spatial keyword query and shortest path query. The k nearest neighbor query returns the k user which are the k closest users to query location . For influence query, we follow the definition of influence in [3] which is the influence set (also called reverse k nearest neighbor) of the query location. The spatial keyword query returns the k closest indoor objects which contains the query keyword. The shortest path query returns the shortest path from the origin location to the destination location in indoor space. For each query, we developed efficient algorithm by utilizing the benefits of the index structures.
- **Data visualization module:** The visualization module is responsible for displaying the indoor space and query results in a friendly manner. We provide a 3D floor plan visualization for the indoor space. We also provide the index structure visualization function to display the tree structure of the corresponding VIP-tree.

4 DEMONSTRATION

In this section, we will introduce the implementation and setup of *IndoorViz* . Then we demonstrate how to interact with *IndoorViz* for the four queries we introduce above.

4.1 Demo Setup

The *IndoorViz* is based Browser/Server architecture. We implemented the server end in C++ to ensure the efficiency of indexing and query processing. And the server is compiled to DLL for the browser end to invoke. The browser end is implemented using JavaServer Pages and JavaScript 3D library is adopted to 3D floor plan display. We use one synthetic dataset and three real datasets:Melbourne Central, Menzies building and Chadstone Shopping Centre for demonstration. Melbourne Central is a major shopping centre in Melbourne and consists of 297 rooms spread over 5 levels. Menzies building is the tallest building at Clayton campus of Monash University consisting of 17 levels (including basement and ground floor) and 1306 rooms. Chadstone Shopping Centre is the largest shopping centre in Australia with total retail floor area over 200,000 m^2 and consists of around 668 stores across 20 levels. To get the objects dataset, we collect around 140,000 products and extract the keywords from their websites. The detailed data sets description can be found in [6] .The user locations are randomly generated following uniform distribution to represent customers in these venues.

4.2 Demo Plan

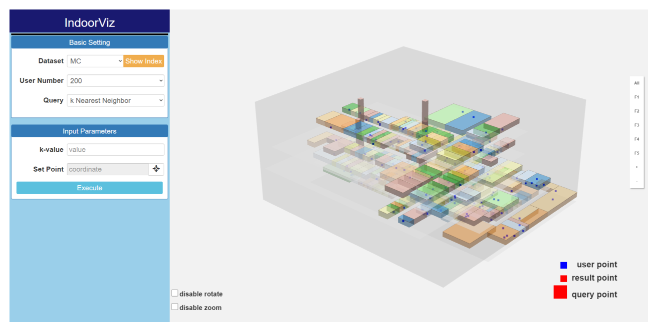

Figure 5: Interface of *IndoorViz*

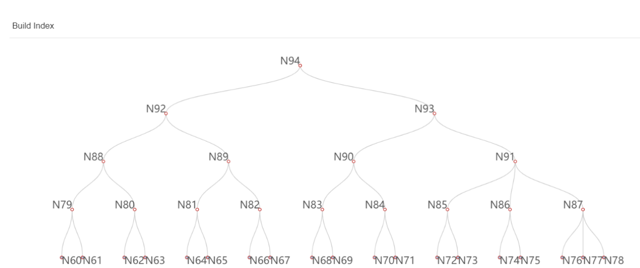

Figure 6: The index structure

1. The Interface.

Figure 5 shows the web interface of *IndoorViz* which consists of two main display areas. The right hand-side part is the 3D display area which can show the indoor space and the query results in 3D . The 3D indoor space supports the operations of zoom-in and out, rotation and drag which enables the 360° display of the indoor space. The left hand-side part is a user input area and detailed results display area. Users can choose the datasets, number of users and the query type from the basic settings and input query parameters according to different queries. There is a show index button besides the datasets selection which can show the VIP-tree structure of the selected dataset. Figure 6 shows the index structure of Menzies dataset. The results display area will show the detailed result information after query processing, such as the object ID, floor number and locations.

2. k Nearest Neighbor Query. k nearest neighbor query is a fundamental query in spatial databases, especially in indoor space. In *IndoorViz* , users can specify the k through the input box and select the query location by clicking the floor plan. After clicking the execute button, the results will display in the 3D floor plan as red points and the detailed results information will show on the left side of the screen. Figure 7 shows the results of k nearest neighbor query with $k = 10$.

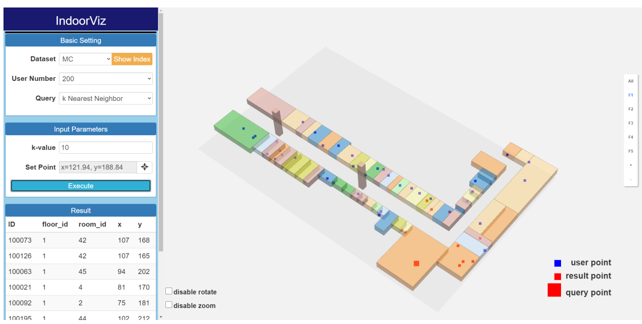

Figure 7: Example of k nearest neighbor query.

3. Influence Query. Influence query returns the size of influence set (also called reverse k nearest neighbors) of a query location [3] which are the potential users who could be influenced by the facility at the query location. By choosing the influence query from the query drop-down menu, *IndoorViz* will show the influence of each room by computing the influence set. The results is shown through heat map which reflects the influence of rooms. Figure 8 shows the results of influence query. For each room, there is a colored circle with number in the center. The number shows the size of the influence set while the radius and color reflect the influence of the room. Specifically, the larger and darker the circle is, the higher influence of the room is. Users can also click the circles to show the detailed influence set information, e.g. the users in the influence set.

4. Spatial Keyword Query. Spatial keyword query returns the k closest objects to query location that contain every keyword in the query keyword set [6]. To issue this query, users have to input the value of k and the query keywords in the input parameters area and select the query location by clicking the query location in the indoor space. The results will show on the floor plan and the results display area. Figure 9 shows the results of querying the 5 nearest objects that contain the query keyword "water". The results are shown as blue points in the room which may be a convenience store.

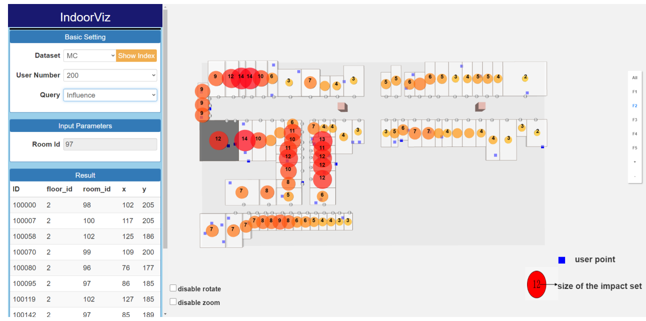

Figure 8: Example of influence query.

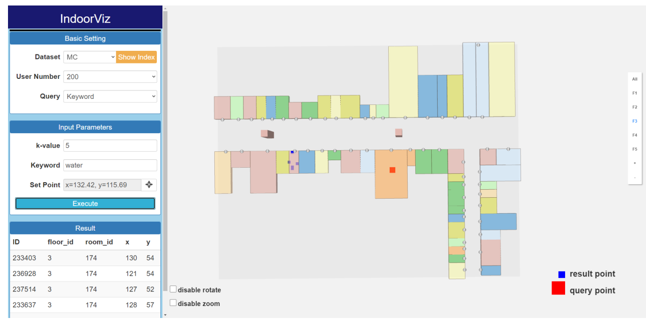

Figure 9: Example of spatial keyword query.

5. Shortest Path Query. In shortest path query, users have to choose the origin and the destination locations from the indoor space. As shown in Figure 10, the shortest path from a room in level 1 to a room in level 3 is shown as the red path which connects the origin room, lift and the destination room. Users can switch from different levels to show the path clearly.

Figure 10: Example of shortest path query.

ACKNOWLEDGMENT

We sincerely thank the anonymous reviewers for their feedback which helped improve our work. The research of Shiyu Yang is supported by NSFC61802127 and Shanghai Sailing Program 18YF1406700. Muhammad Aamir Cheema is supported by ARC grants FT180100140 and DP180103411. Xuemin Lin is supported by ARC DP200101338 and ARC DP180103096.

REFERENCES

[1] B. Yang, H. Lu, and C. S. Jensen. "Probabilistic threshold k nearest neighbor queries over moving objects in symbolic indoor space." In EDBT, 2010

[2] Lu, H., Cao,X. and Jensen,C. S.. "A foundation for efficient indoor distance-aware query processing." In ICDE, 2012.

[3] Yang, S., Cheema, M.A. and Lin, X., "Impact Set: Computing Influence Using Query Logs." in The Computer Journal, 2015.

[4] Shao, Z., Cheema,M. A., Taniar, D., and Lu, H. . "VIP-Tree: An effective index for indoor spatial queries." in PVLDB 2016.

[5] Shao, Z., Cheema, M. A., Taniar, D.. "Trip Planning Queries in Indoor Venues." in Computer Journal 2018.

[6] Shao, Z., Cheema, M.A., Taniar, D., Lu, H. and Yang, S. "Efficiently Processing Spatial and Keyword Queries in Indoor Venues." in TKDE 2020.

[7] Liu, T., Feng, Z., Li, H., Lu, H., Cheema, M.A., Cheng, H. and Xu, J., "Shortest Path Queries for Indoor Venues with Temporal Variations." In ICDE 2020

[8] Feng, Z., Liu, T., Li, H., Lu, H., Shou, L. and Xu, J., "Indoor Top-k Keyword-aware Routing Query." In ICDE 2020.

[9] Li, H., Lu, H., Shou, L., Chen, G. and Chen, K.,"In search of indoor dense regions: An approach using indoor positioning data." in TKDE 2018.