Toward Mining User Movement Behaviors in Indoor Environments

Shan-Yun Teng[1], Wei-Shinn Ku[2], Kun-Ta Chuang[1]
[1]Dept. of Computer Science and Information Engineering, National Cheng Kung University, Taiwan
[2]Dept. of Computer Science and Software Engineering, Auburn University, USA
Email: syteng@netdb.csie.ncku.edu.tw, weishinn@auburn.edu, ktchuang@mail.ncku.edu.tw

Abstract

In this paper, we explore a new mining paradigm, called User Visited Patterns (abbreviated as UVP), to discover user visited behavior in the mall-like indoor environment. It is a highly challenging issue, in the indoor environment, to retrieve the frequent UVP, especially when the concern of user privacy is highlighted nowadays. The mining of UVP will face the critical challenge from spatial uncertainty. In this paper, the proposed system framework utilizes the probabilistic mining to identify top-k UVP over uncertain dataset collected from the RFID-based sensing result. Moreover, we redesign the indoor symbolic model to enhance the accuracy and efficiency. Our experimental studies show that the proposed system framework can overcome the impact from location uncertainty and efficiently discover high-quality UVP, to provide insightful observation for marketing collaborations.

1 Introduction

With the evolution of modern cities, the time duration staying in indoor spaces becomes longer for people living in metropolises. Indoor activities, such as window shopping in malls or indoor sports in gymnasiums, also become increasingly popular as leisure-time doings. The trend has led to an interest in identifying hidden patterns describing human behaviors in indoor environments [6, 7].

Motivated by this, we explore in this paper a practicably interesting task, named mining *User Visited Patterns*, to identify patterns which characterize the common sequences of visited spaces among users in indoor environments. For example, a *User Visited Pattern* p of the trajectory shown in Figure 1 is $\{S_7, S_1\}$. Specifically, the application need comes from the observation that people tend to linger away the whole evening at several locations (e.g., an underground market, a bookstore, and a coffee shop) after they get off work. On the weekend, people may spend the whole day at a mall or an outlet. The discovery of *User Visited Patterns* can enable new marketing collaborations among vendors in the same indoor space (e.g., a shopping center), such as a joint coupon promotion. Despite its increasing demand, mining *User Visited Patterns* is left unexplored thus far.

Essentially, the design of location-aware mining highly relies on the availability of precise user location information which can be transformed to the place/point of interest [5, 10]. However, it is not appropriate to deploy all devices within stores to retrieve precise user location as the concern of business security and personal privacy [2, 3]. In this work, we especially consider devices in the indoor spaces to be deployed as the illustration shown in Figure 1. It is, unfortunately, highly challenging to achieve the precise positioning in an indoor space due to hardware limitations and privacy concerns. The factor of location uncertainty will critically impact the effectiveness of traditional location-aware mining algorithms, causing the incorrect conclusions for marketing

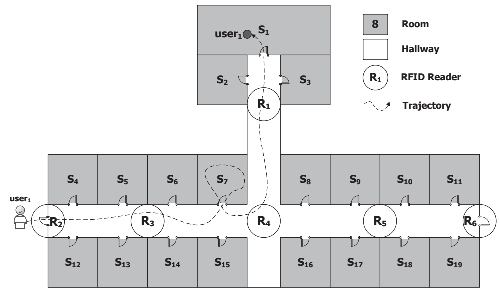


Figure 1: An illustrative example of indoor space and RFID reader deployment.

decisions. To remedy this, recent research advances in the literature have thus devoted to resolving the nature of indoor uncertainty, proposing methodologies of tracking user movements or identifying group moving behaviors in the indoor space [8, 9].

However, it is believed that maintaining the uncertain nature of user location is the key ingredient to the success of user-acceptable marketing strategies. The removal of location uncertainty will benefit the mining quality at the cost of user privacy. For mining *User Visited Patterns*, it is not a viable strategy to employ advanced indoor positioning media, which can precisely detect the location of a user. In this paper, we turn to employ the RFID-based framework as the indoor positioning media, accommodating to its error-prone characteristics in the algorithm design. To resolve the uncertainty problems, we propose a novel symbolic model to characterize the possible location of users, and design a probabilistic based mining algorithm to retrieve *UVP* in the indoor dataset.

The main contributions of this paper are three-fold:

- We present the concept to explore *User Visited Patterns* in the indoor environment.
- We comprehensively process the uncertainty of RFID raw data by proposing a novel symbolic model and probabilistic based mining method to enhance the performance of our framework.
- Empirical studies with synthetic data show the efficiency and accuracy of our framework.

In this work, we discuss the algorithm of mining UVP from the RFID-based uncertain data set in the indoor environment. The remainder of this paper is organized as follows. In section 2 and Section 3, we give the preliminaries and the system framework. The experimental results are conducted in Section 4. Finally, this paper concludes with Section 5.

2 Preliminaries

2.1 Indoor RFID Data

We describe indoor RFID raw data using an example shown in Figure 1. The indoor space is partitioned into several rooms and hallways. In addition, the numbered circles represent the reader detection ranges. Moving objects (human users in this paper) are attached with RFID tags. When a moving object u_i is within the sensing range of an RFID reader, its presence is detected and reported by the reader. Specifically, each raw RFID reading is in the format of ($deviceID, userID, t$), which means the user represented by $userID$ is detected by the device identified by $deviceID$ at time t. For example, as shown in Table 1(a), user u_1 is detected by device R_2 at times t_1, which forms a raw reading r_1. For the trajectory of moving object u_1 illustrated in Figure 1, its indoor RFID data is shown in Table 1(a), where $readingID$ identifies a reading of an RFID reader.

Table 1: Illustrative examples of RFID raw reading data and User Visited Table.

(a) RFID raw reading data

readingID	**deviceID**	**userID**	**t**
r_1	R_2	u_1	t_1
r_2	R_2	u_1	t_2
r_3	R_3	u_1	t_6
r_4	R_3	u_1	t_7
r_5	R_4	u_1	t_{31}
r_6	R_4	u_1	t_{32}
r_7	R_1	u_1	t_{40}
r_8	R_1	u_1	t_{41}
r_9	R_1	u_1	t_{42}

(b) User Visited Table (UTT)

recordID	**deviceID**	**userID**	t_s	t_e
rd_1	R_2	u_1	t_1	t_2
rd_2	R_3	u_1	t_6	t_7
rd_3	R_4	u_1	t_{31}	t_{32}
rd_4	R_1	u_1	t_{40}	t_{42}

2.2 User Visited Table

We consolidate the raw reading data and represent it as a *User Visited Table* with schema ($recordID$, $deviceID$, $userID$, t_s, t_e). Each record in the table states that a moving object $userID$ is continuously detected by the device $deciceID$ in the period from time t_s to time t_e. In addition, $recordID$ identifies a record. For example, as shown in Table 1(b), user u_1 is detected by device R_2 starting from time t_1 to t_2. The *User Visited Table* transformed from Table 1(a) is represented in Table 1(b). Finally, with the records in the *User Visited Table*, a trajectory $T_{u_i} = \{rd_1, rd_2, ..., rd_x\}$ can be formed for a particular moving object u_i.

2.3 Problem Formulation

We give the necessary definitions as follows.

Definition 1 (User Visited Event). *Suppose that a moving object u_i is detected by the RFID reader R_j starting from time t_{sj}, and R_j continues sensing the appearance of u_i until time t_{ej}. Afterwards, u_i keeps being detected by reader R_k, starting from time t_{sk} and ending with time t_{ek}. Thus, we define a User Visited Event e_n of u_i, which is denoted by the 3-tuple $(R_j, R_k, t(e_n))$. And $t(e_n)$ denotes the time interval $[t_{ej}, t_{sk}]$ from R_j to R_k. In this paper, we call User Visited Event as event for short.*

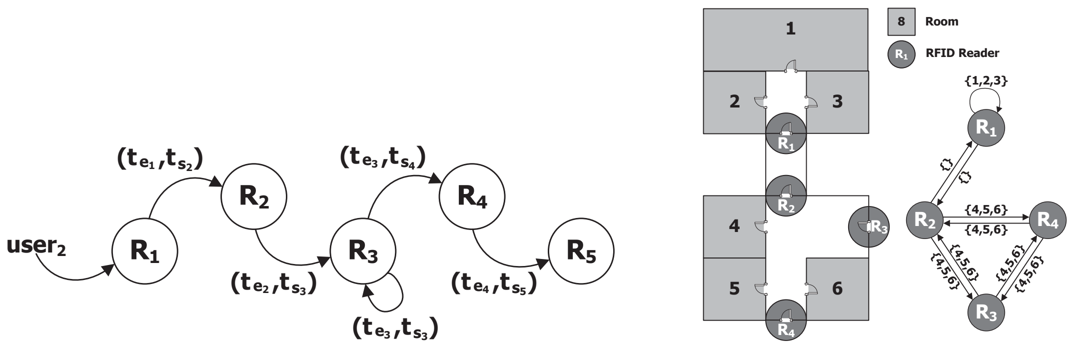
Definition 2 (User Visited Path). *Suppose that u_i is successively detected by R_1, R_2, ..., R_m, where R_m is the last reader in the system that senses the appearance of u_i. The corresponding User Visited Path, abbreviated as path, is denoted by $a_i = \{e_1, e_2, ..., e_n\}$, where e_k is an event, for $1 \leq k \leq n$.*

The illustrative example of the path of user u_2 is shown in Figure 2(a). In addition, instead of following the same principle which is generally used in previous works, we assume no readers are deployed in the room space due to the privacy issue. It is relatively easy to recognize if a user has a valid visit (or window shopping) in a room. The manager of a store can tell the visited time by their experience. For example, customers staying within the store longer than 3 minutes can be identified as valid. As such, we give the following definition.

Definition 3 (Valid State in Rooms). *For a room S_l, we can define the time stayed in S_l as valid, i.e., $t_{stay}(S_l) = [t_{min}(S_l), t_{max}(S_l)]$, where $t_{min}(S_l)$ and $t_{max}(S_l)$ are the minimum time and maximum time that we can state a user staying in room S_l, respectively. In general, $t_{max}(S_l)$ can be defined as infinite.*

Definition 4 (Uncertain Visited Rooms). *Given an event $e_i = (R_j, R_k, t(e_n))$, Uncertain Visited Rooms $uvr_{j,k}$ can be defined as the set of rooms which are placed in the area between readers R_j and R_k. As such, $uvr_{j,k} = \{S_{i,1}, S_{i,2}, ..., S_{i,n}\}$, where each room $S_{i,n}$ is a possible space that a user may get into when the event e_i occurs. In this paper, we state Uncertain Visited Rooms as U-room.*

Definition 5 (Uncertain Visited Transaction). *Given the U-room set of any event in the path a_i, we can*



(a) A user visited path　　(b) A symbolic model

Figure 2: Illustrative examples of a user visited path and a symbolic model.

completely transform the RFID sensing data to the transaction tr_i consisting of U-room sets, i.e.,

$$tr_i = < \{S_{1,1}, S_{1,2}...S_{1,k_1}\}, ..., \{S_{m,1}, S_{m,2}...S_{m,k_m}\} > .$$

In this paper, we state the Uncertain Visited Transaction as U-transaction.

Problem Formulation (Top-k User Visited Patterns Discovery): Suppose that a *User Visited Pattern* (abbreviated as UVP in the sequel) is defined as the form:

$$p_i = \{S_{i,1}, S_{i,2}, ..., S_{i,k}\},$$

and its support value $f(p_i)$ equals to its expected occurrence count in U-transactions. Given the uncertain database D and the desired number k, the goal of our framework is to discover top-k *UVP* from U-transactions according to the expected support of each *UVP*.

3 System Framework

3.1 Symbolic Model Design

Symbolic model is a graph that describes the topology of indoor spaces in which each separating space such as a room or a hallway is represented as a vertex [4]. In addition, edges capture the connectivity (undirected graph) or accessibility (directed graph) between two vertices. In our work, all readers are placed in the hallways, and the specific room for a user getting into is invalid. As such, our symbolic model can not be the same as the traditional one [4]. To solve the uncertainty issue, our symbolic model is designed as follows: each node represents a reader R_i in the indoor environment, the edge connecting nodes R_i and R_j is labeled as the U-room set $uvr_{i,j}$ between readers R_i and R_j.

A small indoor deployment and its corresponding symbolic model are illustrated in Figure 2(b). Our symbolic model is built as a simple digraph in which one loop may be presented at each vertex, and each ordered pair is the head and the tail of at most one edge.

3.2 User Visited Event Filtering

With the reading records in the user visited table provided from indoor RFID readers, a path $a_i = \{e_1, e_2, ..., e_n\}$ can be formed for a particular moving object u_i. However, not all of the events in this path are useful and meaningful in this work. For some particular cases, the events can be filtered before executing the mining process.

No-Stay Event: As shown in Figure 1, u_1 only goes through the RFID readers R_2 and R_3 without walking into any room. Since our goal is to discover the top-k *UVP*, an event is useful only when any space information can be retrieved. However, for a few events, the users only go through the RFID readers without staying in rooms. Therefore, given an event $e_i = (R_j, R_k, t(e_i))$, we define a $DT_{max}(R_j, R_k)$ as the maximum time of walking from R_j to R_k. If the dwell time $t(e_i)$ of an event e_i is not longer than $DT_{max}(R_j, R_k)$, we can remove this event from the user path.

No-Space Event: In Figure 1, the user u_1 passes through the readers R_4 and R_1. However, the set of U-room set between readers R_4 and R_1 is empty. Therefore, we can also remove such events from the user path as no space information can be retrieved.

Path Split to Transactions: For a path of a moving object u_i, we define a time period t_{split} to obtain a set of transactions. As can be expected, a moving object u_i may leave the indoor environment, and return another day. Occasionally, its presence keeps being detected by the indoor readers and is recognized as in the same path as last time it stayed in the indoor environment, since the RFID tag of u_i is not different. However, if u_i leaves the indoor environment for a long time period (e.g., half a day), the path should be cut off. As a result, we use t_{split} to identify a breaking point to split a path into transactions. If the dwell time $t(e_n)$ of an event e_i is longer than t_{split}, the event e_i is removed from the path, and events e_{i-1} and e_{i+1} are separated into two different transactions.

Finally, based on the proposed symbolic model, we transform each event to its U-room set and generate the U-transactions of users for mining process.

Algorithm 1 $\mathcal{P}$-Apriori

Desc.: X_l:a candidate itemset of size l; L_l:an itemset of size l; x_i: a candidate x_i; $p(x_i)$: expected support of a candidate x_i;

Input: D: U-transactions tr_m; k: top-k; f_{min}: support threshold;

Output: UVP: top-k UVP;

procedure $\mathcal{P}$-APRIORI(D,k,f_{min})
  $L_1 := \{items\}$;
  for $(l = 1; L_l! = \emptyset; l$++) **do**
    X_{l+1}:= candidates generated from L_l;
    for $(m = 1; m <= |D|; m$++) **do**
      for each $e_i \in tr_m$ **do**
        for each $x_i \in C$ **do**
          if $e_i.uvr$ contains $x_j \in X_{l+1}$ **then**
            compute $\mathcal{P}(e_i, x_j)$;
            $f(x_j) = f(x_j) + \mathcal{P}(e_i, x_j)$;
    remove x_i with $f(x_i) \leq f_{min}$ from X_{l+1}
    L_{l+1}:= candidates in X_{l+1};
    P:= $P \cup X_{l+1}$;
  P:=sort(P);
  UVP:=$\{p_1, p_2, ..., p_k\} \in P$;
  return UVP

3.3 UVP Discovery

To retrieve top-k *UVP* from U-transactions of users, we extend the Apriori algorithm [1]. As aforementioned, U-room set $uvr_{j,k} = \{S_{i,1}, S_{i,2}, ..., S_{i,n}\}$ of event $e_i = (R_j, R_k, t(e_n))$ containing possible spaces that a user

Table 2: Simulator parameters.

Parameters	Settings
Number of objects	[2,000-20,000]
Radius of RFID detection range	1.5 meter
Moving objects' speed distribution	μ=1 m/s and σ=0.1
Reader accuracy	95%

may get into when event e_i happens. Therefore, we define a possible combination set C consisting of room sets $c_x = \{S_1, S_2, ..., S_y\}$, for each $S_y \in uvr_{j,k}$. Each combination $c_x \in C$ has a probability presenting in event e_i for the real world, and the probability can be defined as

$$\mathcal{P}(e_i, c_x) = \begin{cases} 1 - \frac{|t(e_i) - (DT_{max}(R_j, R_k) + \sum_{S_y \in c_x} t_{min}(S_y))|}{t(e_i)}, & \mathcal{P}(e_i, c_x) \geq 0. \\ 0, & otherwise. \end{cases} \quad (1)$$

With this probabilistic model, the mining process is described in Algorithm 1. It takes U-transactions and the top-k as input and returns top-k *UVP*. The 1-item set is firstly computed (line 2). Then, all candidates of X_{l+1} (the candidate itemset of size $l+1$) produced from L_l (the itemset of size l) are generated, and only candidates with expected support values exceeding the threshold are maintained (lines 3-13). The expected support of each candidate is to sum up all probabilities of its occurrence count in U-transactions (lines 8-10). Finally, all pattern candidates are sorted based on support values, and the patterns with top-k support values are returned as *UVP* (lines 14-16).

4 Experimental Results

This section presents experimental studies of this research. The system framework is implemented in Java. All of the experiments are executed on a 3.40 GHz Core i7 machine with 4 GB of main memory, running on the Windows 7 operating system.

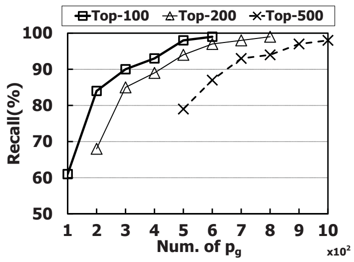
4.1 Experimental Setup

4.1.1 Simulator Implementation

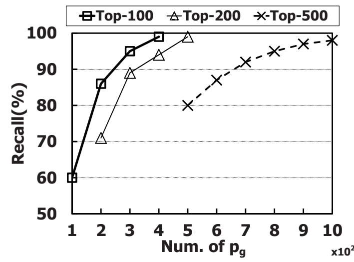
In this paper, the synthetic data of raw RFID reader is generated by applying an indoor raw data generator. This generator, running on the Linux operating system, can simulate the user walking behavior in the indoor environment with the given room layout. The whole simulator consists of true trace generator and raw reading generator. The true trace generator module is used for generating the ground truth traces of moving objects and records of the precise location of each object in every second. It also simulates the objects' speeds using the Gaussian distribution, and the parameters μ and σ of this Gaussian distribution are set to 1 m/s and 0.1, respectively.

In this indoor spatial simulator, we apply a layout of a shopping mall, which is a real MRT-based underground market[1]. In this mall, there are 27 entrances/exits and 168 stores that sell various items (e.g., food, drinks, clothes, shoes, and so on). The generator simulates normal customer purchasing behavior in the indoor environment with 50 RFID readers evenly placed in the hallways. In addition, the dwell time is randomly assigned to simulate the stop-by period spent in a store for a user. The parameters used by this simulator are described in Table 2. Finally, the simulator generates two datasets of user trajectories, that are treated as the

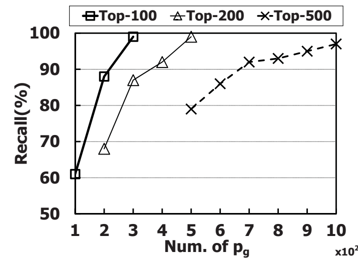
[1] We refer to http://www.datong.taipei.gov.tw/public/Attachment/13811321925.jpg for the layout of Taipei mall.



(a) $|D| = 10,000$

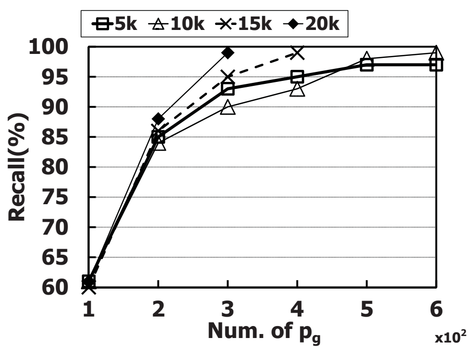


(b) $|D| = 15,000$

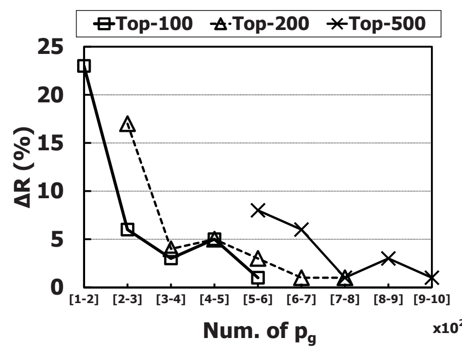


(c) $|D| = 20,000$

Figure 3: The recall of the top-k UVP.



(a) The recall of top-k UVP



(b) The recall difference of top-k UVP

Figure 4: The recall and the recall difference of top-k UVP.

raw reader data and the ground truth data with the precise locations of moving objects. Therefore, we have both ground truth and raw reader data for performance evaluation.

4.1.2 Evaluation Metrics

Recall: In this paper, the fraction of the top-k ground truth *UVP* p_t that is relevant to the generated *UVP* p_g is successfully retrieved. It cares about the probability that a relevant pattern is retrieved, so as to our work, the recall is defined as $\frac{|p_g \cap p_t|}{|p_t|}$.

4.2 Evaluating Our Method

4.2.1 Accuracy performance

We fix the number of trajectories ranging from 100,000 to 200,000 and vary the number of top-k to see the accuracy. The results on the recall of top-k *UVP* are shown in Figures 3(a), 3(b), and 3(c). Obviously, the recall rate is monotonically increasing as the number of generated patterns raises. In addition, two important points are drawn: (1) When the same number of p_g are generated as p_t, the recall value remains within a narrow range of a specific percentage value. (2) These curves stated as the same number of p_t have similar growing trends. It is clear that the number of transactions does not have an obvious impact on the recall rate.

Furthermore, we fix the number of p_t to 100, and then vary the size of trajectories $|D|$ from 5,000 to 20,000. The results on the recall of frequent *UVP* are shown in Figure 4(a). As it can be observed, we have to generate

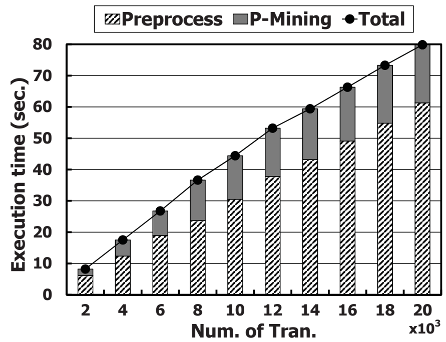


Figure 5: The execution time of preprocess, $\mathcal{P}$-Apriori, and total process.

more *UVP* to reach the recall value of 99% as the size of trajectories decreases. The reason comes from the fact that $\mathcal{P}$-Apriori process could compute the wrong expected support of items in the whole transactions, which would directly affect the support ranking of items and the consequence of top-k *UVP*.

Third, we fix the number of trajectories to 10,000, and then vary the number of p_t ranging from 100 to 500. The difference of recalls with respect to Figure 3(a) are shown in Figure 4(b). It is clear that these recall difference lines decrease faster at first than in the end, which states that the recall curve lines increase faster at first than in the end. As shown in Figure 3, the recalls of curves' starting points are all over 60%. It shows that most of the p_t will be retrieved at the beginning. Then as the number of p_g increases, the fewer p_t we can get. Clearly, the proposed system framework can retrieve high-quality results in the uncertain environment.

4.2.2 Execution time analysis

In this section, we try to analyze the execution time of our system framework. We vary the size of $|D|$ from 2,000 to 20,000, and demonstrate the execution time of the preprocess (including symbolic model construction and event filtering), the $\mathcal{P}$-Apriori, and the entire processing time. As we can observe in Figure 5, the execution time of preprocess is presented as a perfectly linear increasing curve. With regards to the mining efficiency of $\mathcal{P}$-Apriori, it is clear that for a small number of transactions, the execution time is increasing with the significant difference. However, when the number of transactions is huge, the execution time is increasing in an ignorable difference, which shows the stability of our system framework. Finally, the curve of total execution time is linear as that of preprocess. Because the time of preprocess ranging from 6 to 61 is much bigger than that of $\mathcal{P}$-apriori ranging from 2 to 18. Obviously, the time of preprocess dominates the execution time of our system framework.

5 Conclusions

In this paper, we explore the *User Visited Patterns* to identify the indoor mining challenges over uncertain data. Due to the concern of the user privacy, the placement of readers in hallways instead of rooms is considered. We devise a novel system framework to discover the top-k *UVP* from user paths. In addition, we also explore a novel symbolic model and a probabilistic based mining algorithm to efficiently discover the frequent *UVP*. Finally, the framework is studied with empirical observations in order to gain insight into the recall of generated *UVP*. The results show that the proposed framework is effective and efficient to retrieve high-quality patterns.

Acknowledgments

This paper was supported in part by Ministry of Science and Technology, R.O.C., under Contract 105-2221-E-006-140-MY2. The research has also been funded in part by the U.S. National Science Foundation grants IIS-1618669 (III) and ACI-1642133 (CICI).

References

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *International Conference on Very Large Data Bases*, 1994.

[2] M. Gruteser, G. Schelle, A. Jain, R. Han, and D. Grunwald. Privacy-aware location sensor networks. In *Hot Topics in Operating Systems*, 2003.

[3] J. P. Hansen, A. Alapetite, H. B. Andersen, L. Malmborg, and J. Thommesen. Location-based services and privacy in airports. In *International Conference on Human-Computer Interaction*, 2009.

[4] C. S. Jensen, H. Lu, and B. Yang. Graph model based indoor tracking. In *IEEE International Conference on Mobile Data Management*, 2009.

[5] Y. Liu, Y. Zhao, L. Chen, J. Pei, and J. Han. Mining Frequent Trajectory Patterns for Activity Monitoring Using Radio Frequency Tag Arrays. *IEEE Transactions on Parallel and Distributed Systems*, 2012.

[6] H. Lu, C. Guo, B. Yang, and C. S. Jensen. Finding frequently visited indoor pois using symbolic indoor tracking data. In *International Conference on Extending Database Technology*, 2016.

[7] H. Lu, B. Yang, and C. S. Jensen. Spatio-temporal joins on symbolic indoor tracking data. In *IEEE International Conference on Data Engineering*, 2011.

[8] L. Radaelli, D. Sabonis, H. Lu, and C. S. Jensen. Identifying typical movements among indoor objects - concepts and empirical study. In *IEEE International Conference on Mobile Data Management*, 2013.

[9] J. Yu, W.-S. Ku, M.-T. Sun, and H. Lu. An rfid and particle filter-based indoor spatial query evaluation system. In *International Conference on Extending Database Technology*, 2013.

[10] R. Zhang, Y. Liu, Y. Zhang, and J. Sun. Fast identification of the missing tags in a large rfid system. In *IEEE International Conference on Sensing, Communication and Networking*, 2011.