Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 9915408, 17 pages
https://doi.org/10.1155/2021/9915408

Research Article

Multilevel Privacy Controlling Scheme to Protect Behavior Pattern in Smart IoT Environment

Asad Khan,[1] **Muhammad Mehran Arshad Khan**,[2,3] **Muhammad Awais Javeed**,[4] **Muhammad Umar Farooq**,[5] **Adeel Akram**,[6] **and Chengliang Wang**[2]

[1]*School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou 510006, China*
[2]*School of Computer Science and Technology, Chongqing University, Chongqing 400044, China*
[3]*The Department of Examinations, GC University Faisalabad, Pakistan*
[4]*School of Information Engineering, Chang'an University, Xi'an 710064, China*
[5]*School of Computer Science and Technology, University of Science and Technology of China, China*
[6]*School of Information Engineering, Xuzhou University of Technology, China*

Correspondence should be addressed to Asad Khan; asad@gzhu.edu.cn
and Muhammad Mehran Arshad Khan; to_rabimehranrana@yahoo.com

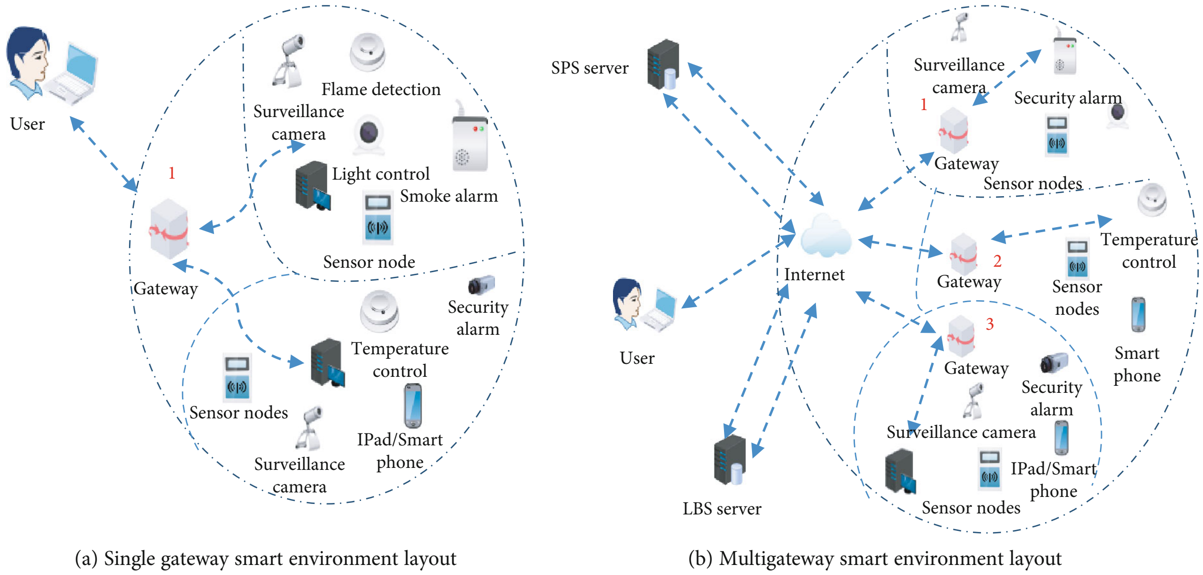
Received 1 April 2021; Revised 7 May 2021; Accepted 21 June 2021; Published 7 July 2021

Academic Editor: Muhammad Shafiq

Traditional approaches generally focus on the privacy of user's identity in a smart IoT environment. Privacy of user's behavior pattern is an important research issue to address smart technology towards improving user's life. User's behavior pattern consists of daily living activities in smart IoT environment. Sensor nodes directly interact with activities of user and forward sensing data to service provider server (SPS). While availing the services provided by a server, users may lose privacy since the untrusted devices have information about user's behavior pattern and it may share data with adversary. In order to resolve this problem, we propose a multilevel privacy controlling scheme (MPCS) which is different from traditional approaches. MPCS is divided into two parts: (i) behavior pattern privacy degree (*BehaviorPrivacyDeg*), which works as follows: firstly, frequent pattern mining-based time-duration algorithm (FPMTA) finds the normal pattern of activity by adopting unsupervised learning. Secondly, patterns compact algorithm (PCA) is proposed to store and compact the mined pattern in each sensor device. Then, abnormal activity detection time-duration algorithm (AADTA) is used by current triggered sensors, in order to compare the current activity with normal activity by computing similarity among them; (ii) multilevel privacy design model: we have divided privacy of users into four levels in smart IoT environment, and by using these levels, the server can configure privacy level for users according to their concern. Multilevel privacy design model consists of privacy-level configuration protocol (PLCP) and activity design model. PLCP provides fine privacy controls to users while enabling users to set privacy level. In PLCP, we introduce level concern privacy algorithm (LCPA) and location privacy algorithm (LPA), so that adversary could not damage the data of user's behavior pattern. Experiments are performed to evaluate the accuracy and feasibility of MPCS in both simulation and real-case studies. Results show that our proposed scheme can significantly protect the user's behavior pattern by detecting abnormality in real time.

1. Introduction

With the rapid advancement of sensor technology and mobile social networks, privacy of user's behavior pattern is becoming an essential part of smart IoT environment. Smart IoT environment typically consists of low power, resource restraint devices, and sensor nodes which are installed over the target region [1]. Sensor technology is associated to user's behavior pattern and human cognitive capture, which have been promoted in almost every smart IoT environment. Smart IoT environment typically consists of variety of embedded sensor nodes, actuators nodes, smart home local gateway, service provider sever (SPS), and users as shown in Figure 1. Personal smart home, business (sales track),



(a) Single gateway smart environment layout

(b) Multigateway smart environment layout

Figure 1: Layout of single and multigateway smart environment.

healthcare (cognitive behavior), and safety (military security and traffic management) are few fields with diverse applications. Furthermore, location-aware services, environmental monitoring, and architectural control are other appliances of smart IoT environment technology. During daily living activities, users interact with smart phones or tablets and can easily download many kinds of location-based server (LBS) applications and data from Google play store or Apple store by submitting their real location and related information to various LBS servers [2, 3]. Basically, if users want to avail the services of smart IoT environment, then they have to share some of their personal data to the service provider server (SPS) through local gateway sensor nodes and actuator devices. Although this kind of services makes daily life of users more comfortable, however, users enjoy these facilities in the smart IoT environment at the cost of their behavior pattern privacy [3]. For instance, users can easily search the location of any room or office by sending message with their location and query data to server through resource-restrained local home gateway [4]. Therefore, the server and low capacity smart IoT environment nodes (SHNs) can continuously access sensitive and personal data from users' requests and observe their personal information, such as their daily behavior pattern including what they do at certain time of a day [5]. More seriously, it can send private information to adversary which could then exploit privacy [6], such as user identity, user office's timing, occupation, home address, and user daily behavior activities. In smart IoT environment, once sensitive data are transmitted over the network, then it will be out of the user's control. All these appalling possibilities conflict with the privacy concerns of users' daily behavior pattern; therefore, we have to focus on users' behavior pattern privacy in a smart IoT environment.

There are two kinds of approaches, for collection of data, to detect abnormal activity: (i) video based and (ii) sensor based. Video-based approaches generally use technology of image processing; however, there are limitations in these approaches:

(i) Identifying the type of user's activity with small scope and small short time duration

(ii) Covering very small area and high cost

(iii) Violating user's privacy

Sensor-based approach is an emerging research area which has been adopted in smart IoT environment in order to tackle abovementioned pitfalls [7]. To some extent, it has been successfully used in smart IoT environment; however, they only process simple trajectory data and occasionally implement centralized data processing [8, 9]. Therefore, many of them have the following disadvantages.

(i) *Lack of Behavior Pattern Privacy*. They only focus on sequence information of activity and ignore important problem of preserving protection of user's behavior pattern privacy.

(ii) *Ignoring Time Duration in Location Privacy*. They ignore the use of time duration in order to detect duration abnormality. Furthermore, it did not consider combining location privacy and user's activity privacy in single approach.

(iii) *Computational Cost*. It consumes large bandwidth and uses centralized approach with long response time.

We focus to cover the abovementioned pitfalls and on protecting the user's behavior pattern privacy in smart IoT environment. The current study does not cover privacy edification of the whole system, and this research is an extension of privacy model. Our work is aimed at solving two main challenges in smart IoT environment, (i) ensuring privacy of user's behavior pattern, e.g., if a user is in a particular building from 9 : 00 to 14 : 00 and adversary can access this information, however, adversary cannot know where he/she was at 10 : 00 a.m. within the building and in which room

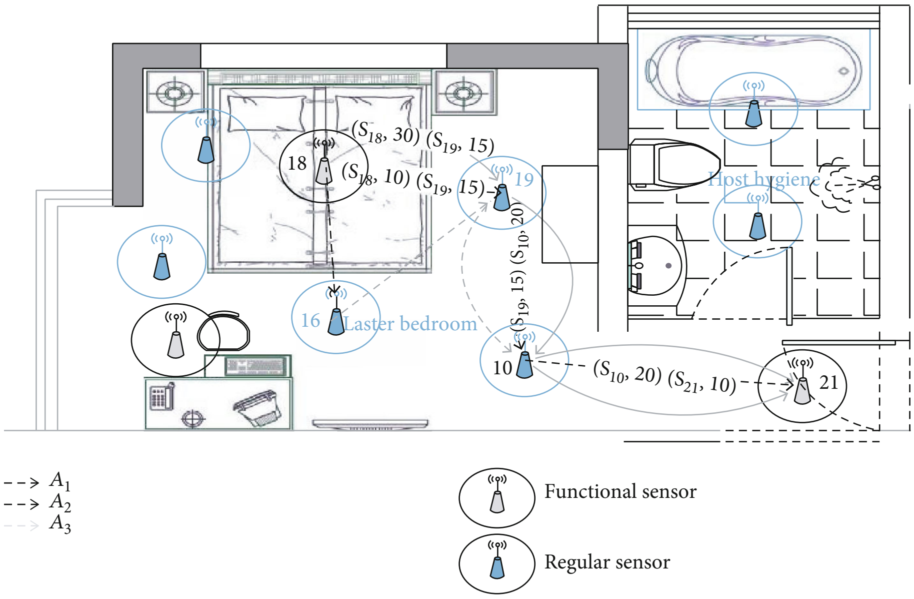


Figure 2: Basic layout of sensor deployment in smart environment. Sensors are classified into REGULAR sensors and FUNCTIONAL sensors. Time duration between sensors and sensors' ID is represented by tuples in lines. These lines also represent activity of trajectories.

he/she is/was at particular time; (ii) the long response time and using large bandwidth during computational process are inappropriate for real-time detection. We proposed multilevel privacy controlling scheme (MPCS) to deal with them. (1) *BehaviorPrivacyDeg* is proposed, in order to (i) keep record of user's activity variation and storing these compact patterns into each sensor with the patterns compact algorithm (*PCA*) and (ii) detecting whether the present activity is abnormal or normal based on the abnormal activity detection time-duration algorithm (AADTA). (2) Protecting user's behavior pattern privacy by using multilevel privacy model, server utilizes PLCP to set privacy level according to concern of user. LPA is used to hide the features of user's real location from adversary or untrusted nodes by generating a number of fake locations. The main research contributions of this paper are as follows:

(i) *BehaviorPrivacyDeg*, a novel technique detecting abnormal activity and compact pattern algorithms, is proposed to cache learned parameters using mining training into every sensor node and to sense abnormal activity at real time based on limited resource restrained of sensors

(ii) The multilevel privacy model has been designed to protect users' behavior pattern privacy. Our model not only utilizes PLCP for optimal configuration of privacy levels but also secures user's data from untrusted nodes caused by unpredictable interference in smart IoT environment

(iii) Activity design model, which consists of activity variation, trajectory variation, and duration variation, to define a small difference between two the same activities because the same pattern of activities cannot be repeated exactly in the same way

(iv) Real data-based simulation and experiments have been conducted which showed that our new approach can efficiently protect users' activity and sensitive data in smart IoT environment

The rest of the paper is organized into the following four sections. We thoroughly overview related previous literature in Section 2. We present our new scheme in Section 3. Simulation and experiments are presented in Section 4. Conclusions are discussed in Section 5.

2. Related Work

A number of research studies have been conducted on protecting privacy of users in smart IoT environment. We hereby briefly discuss and compare their findings. Many privacy protection schemes are introduced as means to protect query privacy and users' location privacy for various situations (e.g., snapshot scenario and continuous scenario in navigation apps.). In [8, 10, 11], authors proposed location perturbation, obfuscation techniques, and temporal cloaking techniques, respectively. Generally, all these techniques are deployed to achieve the privacy goal. These proposed techniques can be gained based on trusted third party such as location anonymizes in [12]. In [10, 13, 14], authors have proposed mobile device-based solutions. In some early works, Chow et al. introduced a solution based on location anonymizer to collect the queries of users and forwarding anonymous data set to location-based server (LBS) to protect users' privacy. However, later it is noticed that location anonymizer resulted in the blockage of entire system. In [15], authors proposed two algorithms, named GridDummy and GirDummy that generate dummy location to achieve k-anonymity for user, considering the location's privacy. These two algorithms generated virtual circle and virtual grid

```
Input:$s_l, d_l, c_i, c_l, s_i, d_i$
Output: $r$-activity-patterns, frequent pattern tree (FP-tree) assigned as $f_t$
    While $(d_i, s_i)$ $do$
    if $c_l = REGULAR\ then$ tree$_{\text{insert}}((s_l, d_l), (d_i, s_i))$ ;
    else if $c_i = REGULAR\ then$ tree$_{\text{insert}}(s_l, d_l)$ ;
    Tree_insert$((s_l, d_l), ((s_l, d_l))$ ; $else$
    Tree_insert$(s_l, d_l)$ ;
    Tree_insert$((s_l, d_l), (d_i, s_i))$ ;  end
    if $(s_l, d_l) = (d_i, s_i)$; nest item will be assigned in server to $(d_i, s_i)$
    end while
    if $(d_i, s_i)$ last item at end of dataset then
    For every activity $A_l$ in FP-tree do
    if $f_t \geq \lambda\ then$
    add $A_l\ into$ $r$-activity-patterns;
    end if
    end for
    end if
    return $r$-activity-patterns and $f_t$ $(FP - tree)$
```

Algorithm 1: FPMTA.

```
Input: $r$-activity-patterns, $g$-activity-patterns, $\gamma$
Output: $c$-activity-patterns: to compact the real normal activity pattern
    Sorting $g$-activity-patterns in order of descending $|d_{A_u}|$ ; //$|d_{A_u}|$ it represents the quantity of activity in data set $d_A$;
    While $g$-activity-patterns' size = 0 do
    Attain first activity pattern $A_u$ in $g$-activity-patterns
    for activity pattern $A_l \in d_{A_u}\ do$
    Delete $A_l$ in $r$-activity-patterns and $g$-activity-patterns;
    for every activity pattern $A_n$in $g$-activity-patterns do
    Delete $A_l$ in $d_{A_n}$;
    end for
    end for
    delete $A_u$in $g$-activity-patterns;
    sorting $g$-activity-patterns in descending order $|d_{A_u}|$ ;
    end while; $c$ − activity − patterns = $r$ − activity − patterns
    return $c$-activity-patterns
```

Algorithm 2: Patterns Compact Algorithm (PCA).

```
Input: table-activity-dect, $d_c, t_c, D_{A_l}, T_{A_l}, \gamma$
    $minu = 1$
    $add\ d_c\ into\ T_{A_l}$ ;
    $add\ t_c\ into\ D_{A_l}$ ;
    reorganize $A_l\ with\ D_{A_l}$ and $T_{A_l}$ ;
    for $i \longleftarrow 1$ to table-activity-dect do
    if dissimilar $(A_u, A_l) < minu\ then$
    $minu = dissmilar(A_c, A_l)$ ;
    end if
    end for
    if $minu > \gamma\ then$
    label $A_l$abnormal;
    return c-activity-patterns;
```

Algorithm 3: AADTA.

which were carefully constructed for privacy area of users. However, Lu et al. ignored the background information and query privacy of the users. Although in some recent research studies [16], authors have paid attention to solve the above-mentioned issues thoroughly; however, they introduced heavy system to achieve k-anonymity. In [17], authors proposed a device free localize (DFL) technique which identifies user's location and their activities simultaneously. The wireless signals have the ability to become a sensor itself that can perceive the context information. In near future, this technique may turn the traditional wireless network into intelligent networks. However, the mechanism of this approach is not efficiently working on limited resource-restrained devices. In [18], Liu proposed a scheme for activity recognition using 2D and 3D cameras. However, video-based techniques and approaches can compromise on privacy issues. Moreover, high cost is required for video equipment. In [19], authors have discussed that users' activity in home such as bathing, cooking, and reading can be accessed by

TABLE 1: Variable detail used in Algorithms 1, 2, and 3.

Variables	Detail
λ	It represents time duration threshold.
γ	It represents variation threshold.
r-activity-patterns	This variable used to store real mined patterns.
g-activity-patterns	It represent the set $\{(A_1, r_{A_u}), \cdots, (A_i, r_{A_u}), \cdots, (A_N, r_{A_N})\}$ and where $A_i \in \mathrm{r-act-patterns}$. And η is number of activities that stored in r-activity-patterns.
r_{A_u}	This variable stores the activities that are the same to A_u.
d_i, t_i, c_i	d_i is for current triggered sensor device/node, and t_i represents the time duration probability. c_i is category of sensor device d_i (REGULAR sensor device or FUNCTIONAL sensor device).
d_l, t_l, c_l	d_l represents last triggered sensor, and t_l represents corresponding time duration probability. c_l is for corresponding type/category.
c-activity-patterns	This variable is used to store the compacted patterns from the real.
A_c, d_c, t_c	A_c represents current user' activity, d_c represents current sensor device, t_c is current time duration probability of sensor device.
D_{A_l}, T_{A_l}	These represent their mining from latest change of state-message.

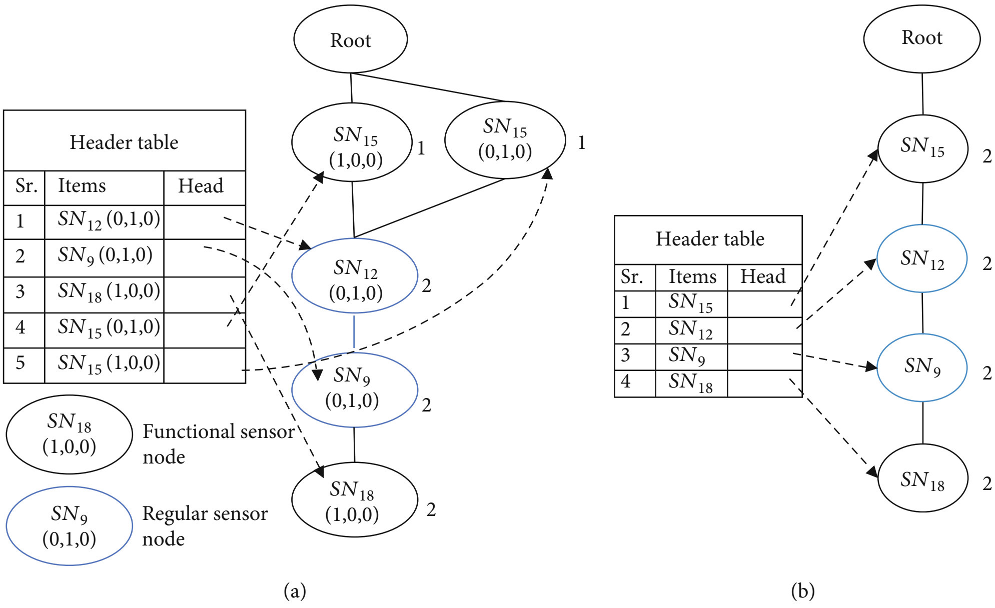


FIGURE 3: Basic difference between two kinds of frequent pattern tree (FP-tree). (a) The trajectory FP-tree and (b) the time duration-based FP-tree.

TABLE 2: Detects abnormal activity table of sensor device S_{16}.

Previous device (d_l)	Previous time-duration probability (t_l)	Particular time-duration probability (t_o)
S_{20}	(1,0,0)	(0.8, 0.2, 0)
S_{15}, S_{11}, S_{12}	(1,0,0), (0,1,0), (0,1,0)	(1,0,0)

unauthorized entities on the wireless network, even all communications are encrypted. In this approach, authors used fingerprints and time-based snooping (FATS) attacks. However, chances of privacy leakage of users' activity are very high due to limitation of this approach in [20–22]; temporal cloaking and spatial assessed time-location are directed to the main server instead of the accurate value. The main focus of these approaches is to prevent exact identification of user's location and thus improving privacy. These techniques harm the timeliness and accuracy of the responses from server, and more seriously, there are some upfront attacks that could still break user privacy. In [23], authors have proposed k-pattern clustering algorithm that classifies complex and varied user activities. This approach also used Allen's temporal relation to predict and recognize users' activities inside home. However, this method did not focus on privacy of users' activities as well as location-based privacy of users. If we observe carefully, most of the recent techniques have some pitfalls such as usage or trust on the third party or server and time-consuming huge processing overhead. In [24], authors provide new system for security institutes to monitor abnormal events. With the help of deep

Table 3: Several levels for protecting privacy of users.

Privacy level	Type	Description
PL1	Zone/region	In this level, user just shares that he is in university but does not allow to share where exactly he is (e.g., in which building and apartment).
PL2	Building/office/room	In this level, user shares his room/office/building but does not share the exact time (e.g., in which office at what time he is/was).
PL3	Time duration	In this level, either he shares approximate time (09.00 to 14.00) about his activity/location or accurate time, day, week, and month etc.
PL4	Activity/action	This level includes the activity/action of the user (e.g., exercise, work, taking class, and watching TV).

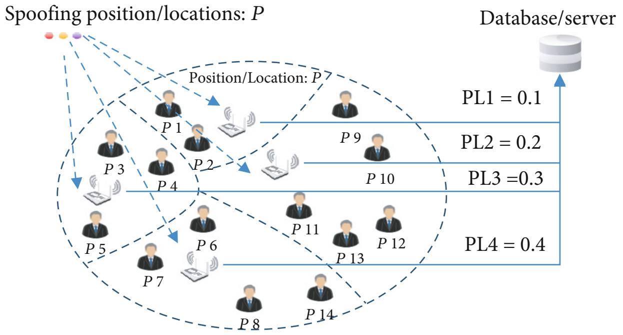


Figure 4: All privacy level that queries to the server. Work flow of PLCP.

learning, authors attained high performance of human behavior recognition by using model tests and training but his scheme does not enable user to define privacy level according to user wish. In [25], authors proposed novel idea based on genetic algorithm to resolve classification problems based on sensor data but they also ignore privacy of user based on sensor data.

Our proposed scheme is different from traditional approaches because our research emphasizes on the user's behavior pattern privacy, including behavior pattern privacy degree, multilevel privacy model, location protection mechanism, and detection algorithm.

3. Multilevel Privacy Controlling Scheme (MPCS)

In this section, we present behavior pattern privacy degree (*BehaviorPrivacyDeg)* and multilevel privacy model of proposed multilevel privacy controlling scheme in detail.

3.1. Behavior Pattern Privacy Degree. Behavior pattern privacy degree (*BehaviorPrivacyDeg*) is aimed at protecting privacy of user's activity variation in smart IoT environment which is as follows: (i) first, it extracts normal behavior pattern from the genuine data and then presents an activity pattern algorithm based on time duration that compresses and reduces the quantity of mined behavior pattern of user's activity; (ii) secondly, it records mined pattern in each device according to record keeping mechanism, and it also detects abnormal activity to protect user's behavior and pattern privacy. *BehaviorPrivacyDeg* uses three algorithms to protect the privacy of user's behavior pattern which are (i) frequent pattern mining-based time-duration algorithm (FPMTA), (ii) patterns compact algorithm (PCA), and (iii) abnormal activity detection time-duration algorithm (AADTA). *Sensors*: we divided sensors into *REGULAR* sensors and *FUNCTIONAL* sensors as per requirement of deployment to sense the data of user's locations and activities as shown in Figure 2. Firstly, set of all the deployed motion sensor devices across the smart IoT environment is represented as $D = \{s_1, s_2, s_3, s_l, \cdots, s_n\}$. User's position is represented by sing location of sensor device s_l which detects the movement of user's position/location P. Sensor devices are defined by N. As we know, all users probably have different velocity of doing activities. Therefore, the time between these sensors during user's activity is different and longer as compared to specific time segment. Activity is produced that is composed of atomic users' activities. Atomic activities $a^i_{l_a} = (s_{l_a}, t^i_{l_a})$ define the trigger of sensor device s_l where $s_{l_a} \in D$, and $t^i_{l_a}$ is trigger time of s_{l_a} in ith sampling period. Number of sampling periods is defined by N_s which represents the conditions when a person passes by s_{l_a}. Basic activity is defined as $\beta_{l_a} = \{\alpha^1_{l_a}, \alpha^2_{l_a}, \alpha^3_{l_a}, \cdots, \alpha^{N_s}_{l_a}\} = \{(s_{l_a}, t^1_{l_a}), \cdots, (s_{i_a}, t^{N_s}_{l_a})\}$ which shows basic activity, where time duration is $d_{l_a} = t^{N_s}_{l_a}, t^1_{l_a}$.

3.2. Frequent Pattern Storage, Compression, and Mining. To store, compress, and detect the abnormal activity, top priority of *BehaviorPrivacyDeg* is mining the user's normal activity pattern to protect behavior pattern.

Definition 1. Normal activity is defined as if frequency of an activity A_u which we assigned as f_u exceeds a particular threshold during appearing in the storage data; then, activity A_u is called a normal activity.

Definition 2. Abnormal activity can be defined as activity that deviates from normal activities in the collected data. In activity recognition, the temporal relationship is foundation of sequence determination [26], and it leads to error of activity recognition. We determine abnormal activity as follows, if there is any kind of activity pattern A_u which apparently seems normal but actually has deviation from normal activity, i.e., $A_{\text{vary}} \leq \gamma, Au$, is determined as abnormal activity. Mostly, supervised learning algorithms for sensor data require several labeled data; therefore, learning algorithms

```
Input. $\rho$ : the desired level of privacy, $\{\rho_i\}$: the level of privacy for user in the data, $\{(\partial_j, \delta_j)\}$: the value for the data model, $s_{ij}$: learned data
$\mathrm{PL}_{\rho_i} \longleftarrow \sum_{j=1}^{k} \partial_j \Pr(\rho_i, \partial_j, \delta_j)$
$T \longleftarrow$ search the set of $\rho_i$, so that $|\rho - \rho_i| < \epsilon$;
$C_{\mathrm{opt}} \longleftarrow \sum_{j=1}^{k} \partial_j$;
Optconf $\longleftarrow \varnothing$ ;
for every user $u_i$ having $\rho_i \in T$ do
    PL1 $\longleftarrow \sum_{j=1}^{k} \partial_j s_{ij}$ ;
    if $C_{\mathrm{opt}} > |\rho_i - \mathrm{PL1}| < \epsilon$then
        $C_{\mathrm{opt}} > |\rho_i - \mathrm{PL1}|$;
        Optconf $\longleftarrow u_i{}'s$ privacy configuration;
    end if
Return Optconf
```

Algorithm 4: Level concern privacy algorithm (LCPA).

unsupervised that saved labor and accelerate the learning speed [27].

3.3. Frequent Pattern Mining. Keeping in mind the Definition 1, we prefer to use frequent pattern mining approach [7] for user' behavior pattern privacy by mining normal activity patterns. Based on frequent pattern mining approach [7], if frequency of an element set exceeds minimum threshold λ within specific time duration, then it is considered as a normal activity. Each path from leaf node to root node and root node to leaf node is defined as pattern A_p, and the frequency is calculated as f_p which represents minimum support count in a path. We use frequent pattern tree (FP-tree) to store quantitative and crucial information about FP-tree and time duration. FP-tree is proposed to achieve the privacy level of data in smart IoT environment. In FPMTA, line 3 and line 5 represent the *insert-tree* function. The function of insert-tree set $((s_i, d_i), (s_l, d_l))$ is inserted in two steps; in first step, node (s_i, d_i) is inserted into *FP-tree* as a child node (s_l, d_l), and in second step, (s_i, d_i) *insert-tree* is to insert a node (s_i, d_i) into *FP-tree* as a child node of root node. If there (s_l, d_l) is a child node (s_p, d_p) or root node which $|s_p - d_i| \le \lambda$ and $s_p = s_i$, then counting of s_p, d_p is incremental by value 1. Suppose it is not the same, then node (s_i, d_i) is inserted into *FP-tree* as fresh child node's root node (s_l, d_l). Variables of Algorithms 1, 2, and 3 are used in Table 1.

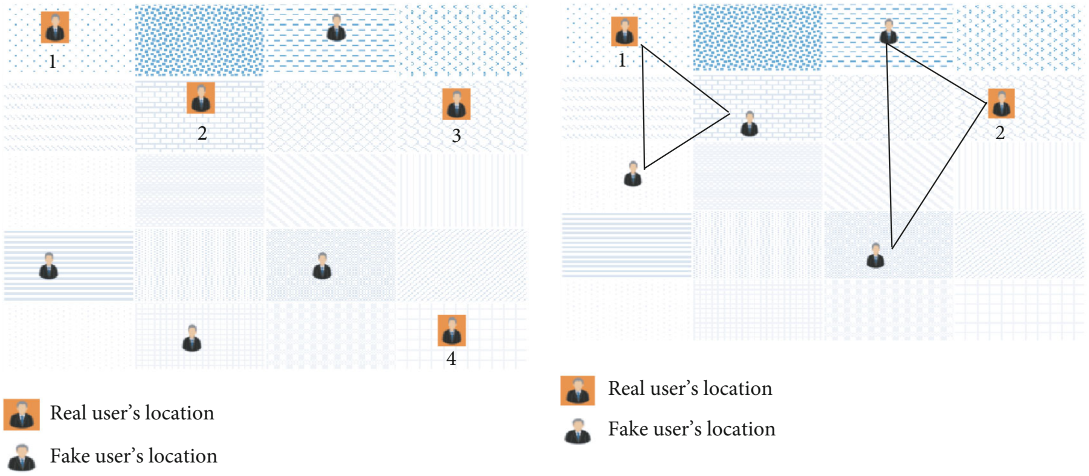
To compress and compact the mined frequent activity pattern of user's behavior, we introduced a PCA. Furthermore, *BehaviorPrivacyDeg* of MPCS introduced abnormal activity detection-based time-duration algorithm (AADTA) to protect the privacy of user's behavior pattern by detecting abnormal activity. AADTA contains sensor device ID d_i and sensor category and table of activity detection that is named as table-activity-dect. Activity table of sensor device S_{16} as shown in Figure 3 is described in Table 2. Mined patterns are stored in relevant room sensor devices separately as per proposed storage method. *Previous sensors* stored the ID in normal pattern field before triggering sensor S_{16}. Time-duration probability is stored by previous time-duration probability corresponding with previous sensors S_{16}.

3.4. Multilevel Privacy Design Model. The term privacy conveys various concepts such as privacy of activities, location, time duration, and decisional privacy. The form of privacy discussed in this section is user's behavior pattern privacy based on activities. We divided user's behavior pattern privacy into four levels termed as privacy level-1 (PL1), privacy level-2 (PL2), and so on as discussed in Table 3 and Figure 4. Let $\mathrm{PM} = \{\mathrm{PL}_1, \mathrm{PL}_2, \mathrm{PL}_3, \mathrm{PL}_4\}$ be the set of privacy model, including four privacy levels. The ability of multilevel privacy model is to deal privacy of user's behavior pattern in smart IoT environment. Multilevel privacy model is comprised of (i) privacy level configuration protocol (PLCP) and (ii) activity design model.

3.5. Privacy-Level Configuration Protocol (PLCP). PLCP is designed to manage privacy of users by controlling privacy levels and transmit data among sensors. In order to avail any service from server, users have to share some information of their privacy level with the server through limited resource sensors as shown in Figure 2. Privacy of user will be changed with the selection of privacy level. Term u_i is for user, and term ρ_i is used for privacy-level concern. At the level PL_i, the average number of hidden data for all user is defined as $\mathrm{PL}_{\rho_i} = \sum_{j=1}^{k} \partial_j \Pr(\rho_i, \partial_j, \delta_j)$ where term δ_j is used as how sensitive the data is perceived by user and ∂_j is used as weight for the data. The $\Pr = (\rho_i, \partial_j, \delta_j)$ bt the value of user's privacy concern. We defined this measure for privacy rating at privacy configuration level PL_{ρ_i}. For the user u_i, the actual weighted number of hidden data $\sum_{j=1}^{k} \partial_j s_{ij}$ is privacy rating at level of PL_{ρ_i}. PLCP uses level concern algorithm and privacy level index mechanism.

$$\sum_{j=1}^{k} \partial_j s_{ij} = \sum_{j=1}^{k} \partial_j \Pr(\rho_i, \partial_j, \delta_j). \qquad (1)$$

(1) *Level Concern Privacy Algorithm (LCPA).* LCPA provides a way for finding the optimal privacy



(a) Real and fake users' location

(b) Fake location with bigger and smaller cloaking region (CR)

Figure 5: Layout of our basic approach in smart environment.

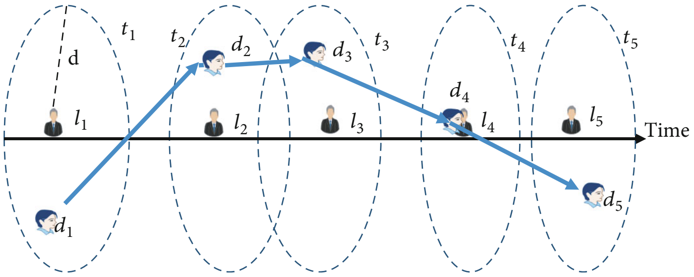


Figure 6: The scenario of user's locations inside smart environment.

configuration for a desired level of privacy concern. A new user can stipulate his/her level of privacy concern ρ_i based on the relative value in the range (1 to 4). LCPA assumes the data item models $\Pr = (\rho_i, \partial_j, \delta_j), j = 1 \cdots ..k$. Also, privacy configuration for each user has been calculated as ρ_i, $i = 1{,}2{,}3{,}4$. LCPA first calculates the privacy level PL_{ρ_i} for user level of privacy concern ρ_i with the data item model and then searches whether the user has the similar level of privacy concern $|\rho - \rho_i| < \epsilon$, where $\epsilon > 0$ is very small value according to LCPA.

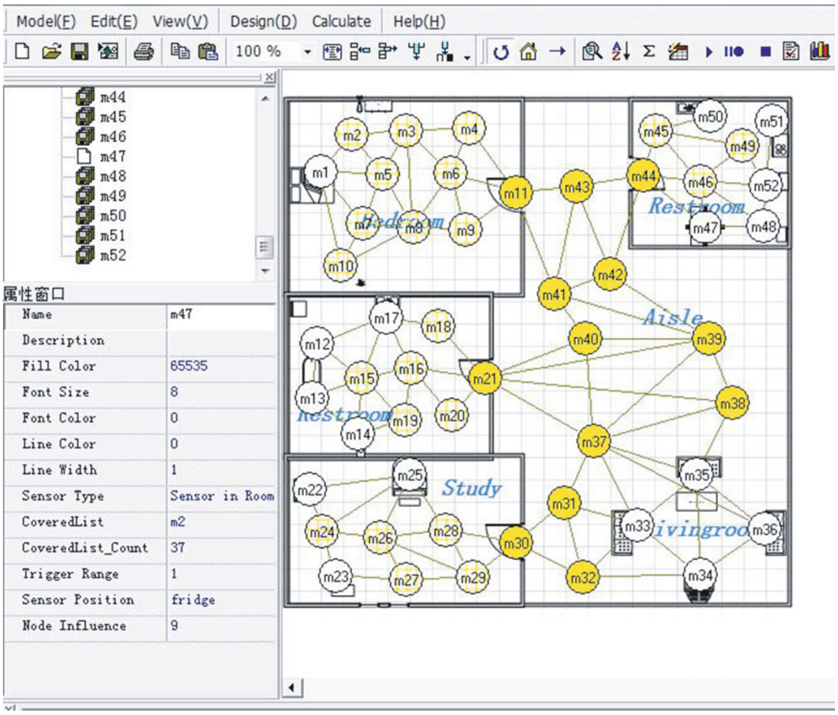
(2) *Privacy Level Index Mechanism*. We introduced a new privacy level index mechanism which is used to assign index for each level. Let us assume that privacy level has been PL1-assigned index 0.1, PL2 has been assigned 0.2, PL3 has been assigned 0.3, and PL4 has been assigned index 0.4. User can use these indices to set their privacy level according to their concern in our smart IoT environment. At the same time, different user has different privacy levels and these levels are used by our proposed MPCS to protect the user's behavior pattern privacy.

Each privacy level has data set as discussed above. Sometimes the user is more conscious about information of his location and sometimes about information of his time duration etc. Figure 4 illustrates how PLCP works. Assuming that one user follows the PL1 in Figure 4 when sensor is triggered, it first executes the LCPA to control and manage privacy of user's behavior pattern. The target of adversary is to access sensitive information of a user. We focused on two types of adversaries: (i) active adversary, any entity is an active adversary if he can access the untrusted sensor nodes. (ii) A passive adversary, which can eavesdrop on a communication channel between compromised nodes to track other user's sensitive data. We consider gateway and sensor nodes as active adversaries.

3.6. Location-Based Privacy Algorithm (LPA). Privacy levels 1 and 2 include user's location, and in order to protect user's location, we used concept of entropy. Entropy is used to measure the degree of k-anonymity. To calculate entropy, each location has probability of being queried q_i and probability donated by b_i is 1. To identify the individual's entropy E in users, set is defined as

$$E = -\sum_{i=1}^{k} b_i \cdot \log 2b_i. \tag{2}$$

Our goal is to attain the maximum entropy, which can be achieved when all possible positions/locations P have the same probability $1/P$ where the maximum entropy will be $E_{\max} = \log_2 P$. Server can assume real location with high probability as $1/P - P_d$, where P_d represents the number of



Name	m47
Description	
Fill Color	65535
Font Size	8
Font Color	0
Line Color	0
Line Width	1
Sensor Type	Sensor in Room
CoveredList	m2
CoveredList_Count	37
Trigger Range	1
Sensor Position	fridge
Node Influence	9

Figure 7: Motion sensor-based smart environment setup for simulation experiment.

fake locations and server will calculate it based on probabilities of their low query. The query probability is higher than others in locations 1 to 3 and on the basis of information.

It means that $P - P_d = 3$,and entropy will drop considerably from $\log_2 P$ to $\log_2 (P - P_d)$. We enhance privacy of users in two phases: (i) first, we try to select fake locations of users with the same query probabilities; (ii) second, if there are more than one user, the fake location spread is as far as possible. Suppose the user's location map is segregated into equal size cells $n \times n$ as shown in Figure 5. Each cell of the map has its own enquiry probability that is based on previous query history as follows:

$$q_i = \frac{\text{number of enqueries in each cell } i}{\text{number of enqueries in full map}}, i = 1, 2, 3, \cdots\cdots\cdots .n^2, \quad (3)$$

where

$$\sum_{i=1}^{n^2} q_i = 1. \quad (4)$$

To provide a degree of k-anonymity, in addition to real locations, we need to conclude the other $k-1$ cells to assign the fake locations. The user selects the P cells right before and P cells right after real location from sorted list as $2P$ users. Therefore, user make N set of cells, and in every set, one cell belongs to real user's location and the others are randomly selected from $2P$ users. The $m^{\text{th}} (m \in [1, n)$ set is represented as $R_j = [r_{j1}, r_{j2}, \cdots, r_{ji}, \cdots, r_{jk}]$. The normalized query probability of the involved cells which is based on real query probabilities of the selected cells can be presented as s_{j1}, $s_{j2}, \cdots, s_{ji}, \cdots, s_{jk}$ and calculated by summing it to 1.

$$s_{j1} = \frac{q_{bi}}{\sum_{l=1}^{k} q_{bl}}, i = 1, 2, \cdots\cdots, k. \quad (5)$$

To effectively achieve k-anonymity of P location, we need to create an optimal set. The level of privacy is guaranteed by using the entropy metric that is extensively used to measure privacy of users. We compute entropy for specifically selected set R_j as follows:

$$E_j = -\sum_{i=1}^{k} r_{ji.}.\ \log 2\, r_{ji.} \quad (6)$$

Finally, the LPA achieved the set with effective and highest degree of entropy.

$$R = \operatorname{argmax} E_j. \quad (7)$$

To measure the cloak region (CR), distances between pair of fake locations are calculated and the sum of distances can be utilized to measure the CR which is $\sum_{i=j} d\,(r_i, r_j)$ where $d\,(r_i, r_j)$ represents the distance between rows/cells r_i and r_j. In Figure 6, l_1 represents real location of user and d_1 is selected as a fake location of the user, since it is considered farthest location from l_1. Furthermore, suppose there are two choices for assigning third fake locations d_2 and d_3. We select it based on the sum of distance between

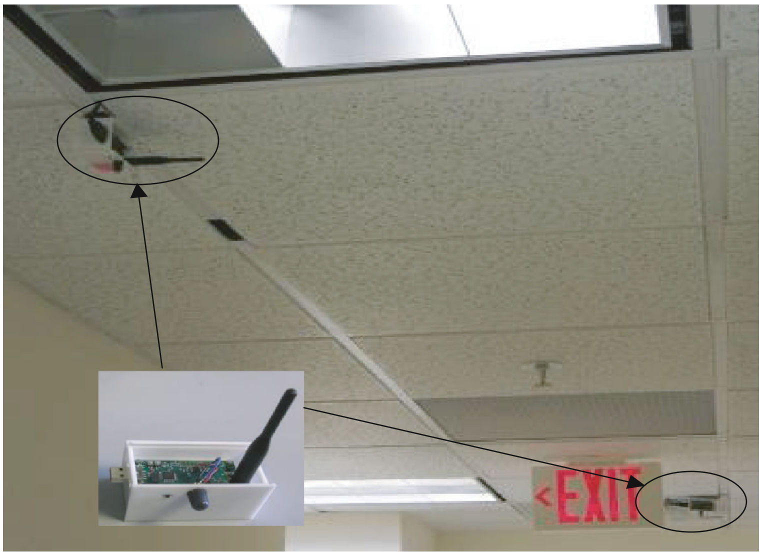


Figure 8: Example of the smart IoT environment and deployment of motion sensor devices are shown.

pairs of fake user's locations. We have to select either of them because $d_2 l_1 + d_2 d_1 = d_3 l_1 + d_3 d_1$. In this scenario, $d_2 l_1 . d_2 d_1 = d_3 l_1 . d_3 d_1$; hence, we select d_2 as a fake location. Let $R = [r_1, r_2, \cdots, r_k]$ represents the set of fake and real user's location. Multiobjective optimization problem (MOP) is described as

$$\max \left\{ -\sum_{i=1}^{k} b_i \cdot \log_2 b_i, \prod_{i \neq j} d\left(r_i, r_j\right) \right\}, \tag{8}$$

where $r_i, r_j \in R$, k_i, and k_j represent the query probabilities of the r_i and r_j, respectively. Our first priority is to confuse the adversary so that adversary cannot target the specific location of user. This objective can be represented as follows:

$$R = \operatorname{argmax} \left(-\sum_{i=1}^{k} b_i \cdot \log_2 b_i \right), \tag{9}$$

That is basic condition to achieve the higher entropy by using a set of fake locations. Optimal combination of P locations is as follows:

$$R = \operatorname{argmax} \prod_{i \neq j} d\left(r_i, r_j\right). \tag{10}$$

Time Duration. The time duration T_d is divided into three parts: small, medium, and big; thus, fuzzy logic [28] is used to calculate the time duration, and fuzzy inference system (FIS) [29] is adopted to measure the probability of T_d being small ($P_{ş-T_d}$), medium ($P_{ɱ-T_d}$), and big ($P_{ƀ-T_d}$). Basic activity is defined as β_{l_a}, and sensor device is defined as s_{di} so as a result $\beta = (T_{d_a}, s_{i_a})$ is redefined as $\beta = (s_{i_a}, (P_{ş-T_d}, P_{ɱ-T_d}, P_{ƀ-T_d})$. In this paper, small time duration range is 0 to $\tilde{d}_{t3}$, and medium time duration range is from $\tilde{d}_{t1}$ to $\tilde{d}_{t4}$, and big time duration range is from $\tilde{d}_{t2}$, where $0 < \tilde{d}_{t1} < \tilde{d}_{t2} < \tilde{d}_{t3} < \tilde{d}_{t4}$. Each s_{i_a} stores $t\_\text{rule}^i = \{d^i_{t1}, \tilde{d}^i_{t2}, \tilde{d}^i_{t3}, \tilde{d}^i_{t4}\}$, and t_rule^i is fixed according to location and monitoring zone of sensor device s_{id}. The mean of maximum scheme is appropriate for our method. Assumed activity A_1 as an example and we set $\tilde{d}_{t1} = 5$, $\tilde{d}_{t2} = 20$, $\tilde{d}_{t3} = 35$, and $\tilde{d}_{t4} = 50$. After using fuzzy logic, term $(s_8, 15)$ can be defined as $(s_8, (0, 1, 0))$.

Table 4: Experiment result of performance test.

Number of pattern	Average time of execution
6	77.5 m/s
12	80.4 m/s
18	86.3 m/s
24	93.9 m/s

Table 5: Detecting abnormalities.

Parameters	Algorithm detecting_ activity trajectory	
Size of model	95	95
Trajectory_abnormality	73	73
Disc_abnormal_trajectory	75	91
Size of model	54	54
Time_duration_abnormality	38	38
Disc_time_duration_abnormality	35	Fail

Activity Design Model. In this section, we described the concept of activity variation. Activity variation can be defined as small difference between two the same activities because the same pattern of activities cannot be repeated exactly in the same way. Activity variation consists of trajectory variation and duration variation which is used to measure this small difference.

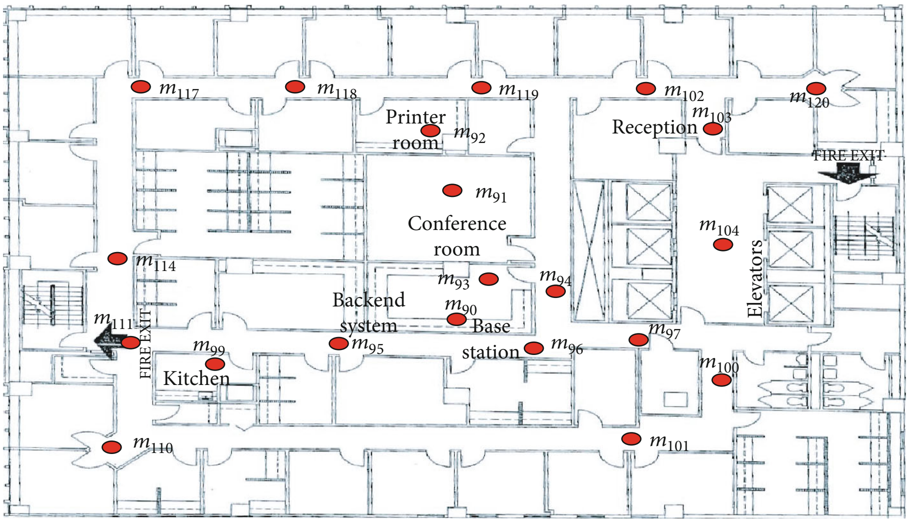
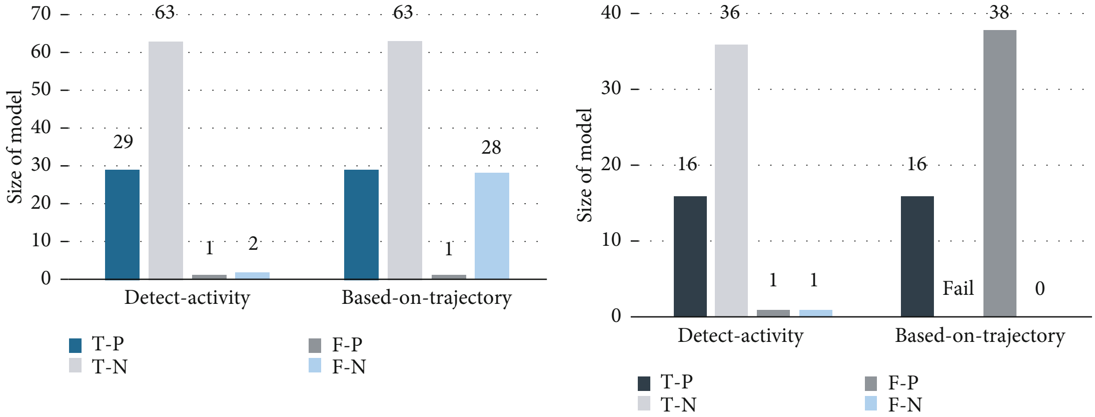


Figure 9: Deployment layout of motion sensor devices network in smart IoT environment. The position of sensor and its ID are shown.



(a) Detection abnormal trajectory

(b) Detection abnormal time duration

Figure 10: Experiment results during simulation test.

Table 6: Detail of stored patterns in sensors.

Sensor device ID	Total patterns	Sensor device ID	Total patterns	Sensor device ID	Total patterns
d_1	2	d_{13}	17	d_{25}	17
d_2	4	d_{14}	14	d_{26}	19
d_3	7	d_{15}	13	d_{27}	14
d_4	9	d_{16}	9	d_{28}	13
d_5	12	d_{17}	12	d_{29}	13
d_6	15	d_{18}	7	d_{30}	11
d_7	17	d_{19}	8	d_{31}	9
d_8	13	d_{20}	10	d_{32}	7
d_9	15	d_{21}	13	d_{33}	5
d_{10}	16	d_{22}	11	d_{34}	3
d_{11}	11	d_{23}	12	d_{35}	2
d_{12}	15	d_{24}	15	d_{36}	7

(a) *Trajectory Variation*. The term trajectory variation is defined as T_v. Activities A_0 and A_1 as shown in Figure 5 take as an example $T_{\text{va1}} = s_{18} \longrightarrow s_{20} \longrightarrow s_{20} \longrightarrow s_{21}$ but $T_{\text{va2}} = s_{18} \longrightarrow s_{10} \longrightarrow s_{20} \longrightarrow s_{21}$, and this represents the same activity but with a small difference in trajectory. This trajectory variation is measured by ᴍ_variation, and the difference between two trajectories T_{vn} and T_{vm} is calculated as

$$T_{\text{variation}}\left(T_{vn}, T_{vm}\right) = \text{Minu}\left(\left|T_{vn} - T_{vm}\right|, \left|T_{vm} - T_{vn}\right|\right) + \left\|T_{vn}\right| - \left|T_{vm}\right\| + \text{order}\left(T_{vn}, T_{vm}\right) = \mathrm{M}_{\text{variation}}. \tag{11}$$

$|T_{vn} - T_{vm}|$ represents the total number of s_{di} which $s_{di} \in T_{vn}$ and $s_{di} \notin T_{vm}$. $\|T_{vn}| - |T_{vm}\|$ explain the length between T_{vn} and T_{vm}. Order (T_{vn}, T_{vm}) computes the difference in sequence between T_{vn} and T_{vm} [26].

```
Input: real location $L_{\text{real}}$, sets of $N$ and $P$, probabilities of query in $q_i$.
Output: set of fake-locations
  1. All cells sort on based probabilities of their query
  2. Select fake $2P$ of users among which $P$ user is right before $L_{\text{real}}$ and $P$ user right after $L_{\text{real}}$in stored list.
  3. for $(m = 1; j \leq N; m++)$ do
  4. develop a set $R_j$ which consist of $L_{\text{real}}$and $P - 1$, additional cells are randomly chosen from users $2P$;
  5. Calculates the normalized probability $s_{ji}$ for every cell $r_{ji}$ in the set.
  6. $E_j \longleftarrow -\sum_{i=1}^{k} b_{ji} \cdot \log_2 b_{ji}$ ;
  7. End
  8. Output max $E_j$;
```

Algorithm 5: Location-based privacy algorithm

(b) *Time Duration Variation*. As discussed above, activities A_0 and A_1 as shown in Figure 5 are not the same activities due to the difference of time duration in s_{19}. However, another activity $A_y = \{(s_{10}, 11), (s_{18}, 25), (s_{19}, 13), (s_{18}, 15)\}$ is not same with A_1, and difference of time duration is small. Term T_{dv} is used for time duration variation. Therefore, the variation between the duration of two activities can be calculated as

$$\begin{aligned}\text{Dissimilarity}\left(A_x, A_y\right) &= \frac{T_{dv} - \text{deviation}\left(T_{vx}, T_{vy}, D_{A_x}, D_{A_y}\right)}{\left(\left|T_{vn}, T_{vm}\right|\right)/2} \\ &\quad + \frac{T_v - \text{deviation}\left(A_x, A_y\right)}{\left(\left|T_{vn}, T_{vm}\right|\right)/2} \\ &= A_{\text{dissmilarity.}}\end{aligned} \tag{12}$$

Activity variation of PL4 is calculated by equations (10) and (12), where ɳ is the duration threshold. The variation threshold is defined as Γ to measure the similarity, and if $A_{\text{simlarity}} \leq \Gamma$, then A_x is considered as similar to A_y.

4. Experiments

4.1. Simulation-Based Experiment. As a simulation model with ground facts, we used smart IoT environment simulator tool to simulate the sensor device-based smart IoT environment, and information was installed manually instead of real setup smart IoT environment. Simulation smart IoT environment is basically divided into three main parts which are as follows.

(1) *Motions Sensor Devices*. We installed more than 100 sensor devices to sense data of location-based users' activity for simulation in smart IoT environment which is shown in Figures 7 and 8. In Figure 7, sensors, which are colored with yellow, are deployed in hallways and elevators. Light yellow sensors are deployed within the rooms, office, and conference rooms. White color sensors are installed in living room, study room, and restrooms.

Table 7: Triggered sensors (known versus unknown).

Task setting	Known	Unknown	ADL
105-115	9	11	73%
115-106	12	15	77%
104-121	10	11	80.10%
112A-109B	4	6	62%
101-119	6	8	72%
Average ADL			73%

(2) *Smart IoT Environment's Trajectory*. We designed more than 15 normal trajectories which have average length of 13. These trajectories reflect typical condition about user's activities.

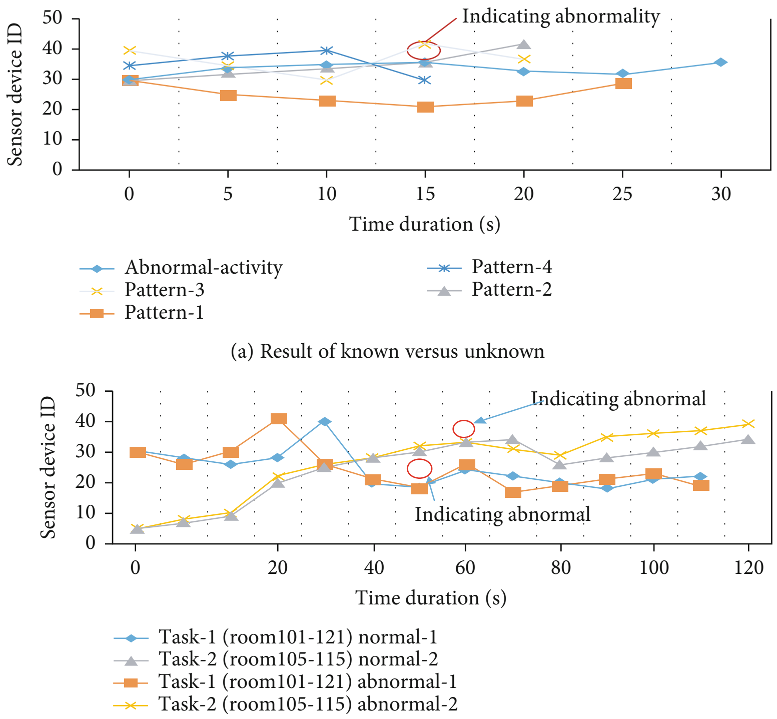
(3) *Time Duration.* As per deployment locations of sensor devices ($d_i^\prime s$ locations) and basic features, three types of t_rule are defined to respond the concerned sensor devices.

(i) In t_rule1, firstly, {2 d, 4 d, 6 d, 8 d} is designed for those sensor devices which are utilized for detecting passing (such as in lobby and hallway).

(ii) In t_rule2, {1 s, 3 s, 5 s, 9 s, 11 s, 13 s} is designed for such sensor devices which are deployed in areas where users may stay for few minutes (such as in washroom and kitchen).

(iii) In t_rule3, {0.4 h, 1.5 h, 2 h, 5 h, 7 h, 9 h} is designed for sensor devices which are located in the area where users will stay for rather long time such as office, study room, and bedroom.

Meanwhile, their time duration of staying is t_d^i, and corresponding *table-activity-dect* are set and assigned with appropriate value manually. The simulation detection system has completed the operations of the LPA, FPMTA, PCA, and the AADTA.



(a) Result of known versus unknown

(b) Result of task normal versus abnormal

Figure 11: Experimental result of tasks.

Real-Time Location. The parameter average distance and location are designed to calculate the real-time location' property. ADL is measured as follows:

$$\mathrm{ADT} = \sum_{i=1}^{n} \frac{L_{\mathrm{dis}}}{|A_u|}. \tag{13}$$

L_{dis} represents the trigger sensor devices during decision-making, and A_u represents the length of A_u. Experiment results showed that ADT of detecting activity is 75.5% which is good as compared to centralized detecting algorithm.

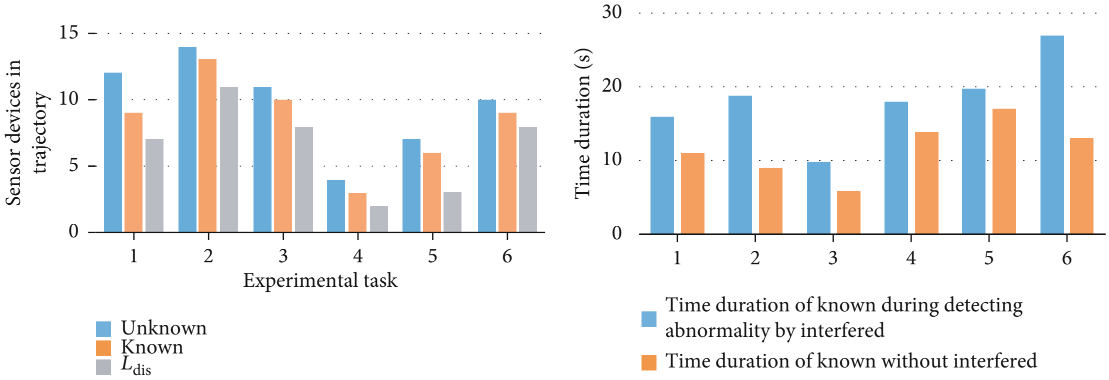
4.2. Lab-Time Experiments. In this section, we conducted real experiment.

4.2.1. Detecting Activity's Feasibility. In smart IoT environment, each sensor device will use AADTA for execution process. In AADTA, the time complexity is $o(t^i)$ and it showed that the time complexity of AADTA is $o(t^i)$. We used TelosW sensor devices for real-time experiment because TelosW has memory size of 1 MB, and it meets the computing capacity of detecting activity. If the average size of stored patterns is 10 at TelosW sensor device, then it means total 7489 patterns $((\text{bytes})(1024 * 1024)/(\text{bytes})((4+4+4+2) * 10) = 7489)$ can be stored on one sensor device in smart IoT environment. It clearly showed that feasibility of sensors' capacity for storage of patterns is enough. Average time of execution of number of patterns is shown in Table 4.

Detecting Abnormalities. Transition probabilities of each sensor in smart IoT environment are represented by $T\_\mathrm{pro}^i = \{t\_p\_r^i, t\_p\_u^i, t\_p\_l^i, t\_p\_d^i\}$. This transaction probability is set to calculate the possibility of which near sensor device will be triggered for next. Considering the deployed sensor devices as shown in Figure 7, if a user triggers m_{27}, the user must trigger $\{m_{26}, m_{22}, m_{25}, m_{24}\}$ as transaction $m_{27} \longrightarrow m_{26}, m_{27} \longrightarrow m_{22}, m_{27} \longrightarrow m_{25}, m_{27} \longrightarrow m_{24}$. If we set $T\_\mathrm{pro}^i = \{0.2, 0.1, 0.3, 0.4\}$ then user will like to select the trajectory $m_{27} \longrightarrow m_{25}$. Moreover, it is also possible that user may choose to do the remaining three trajectories. Here, users are allowed to randomly choose any trajectory from 15 designed trajectories. In other words, users can choose any route depending on the T_pro^i and user can also change his route. We calculated 95 trajectories after repeating 95 times, and only 4 of them are the same as we have designed. 75 abnormal trajectories are detected by our algorithm-based trajectory method [7] and labeled 91 abnormalities, but in real, just 73 abnormalities are produced as shown in Table 5. We use two important keys during simulation experiment when time duration is taking into consideration. Firstly, we use a^i for average speed where i represents the sensor device ID. Average speed represents the approach corresponding with every interlinked device-pair but we set up various speeds during simulation in each sensor device to manage the average speed. Secondly, we assign various speeds with index v^i representing the variance of a^i. When user is passing through sensor device d_i and a^i randomly selects from 0.4 m/s to 1.2 m/s, v^i randomly selects from 0.11 m/s to 0.32 m/s. Time duration t_d^i is altered manually. After repeating and executing 40 times, 40 trajectories are produced with uniform time duration. 30 abnormalities are generated, and our algorithm detected 29 abnormalities by using trajectory-based approach [7].

Table 8: Time-duration number of devices triggered in normal versus abnormal.

Task setting		Senor devices			Time duration (s)	Abnormal-detection-location	ADL
105-115	9	9	48.51	69.1	7	58%	
115-106	12	12	80.2	85.43	8	68.4%	
104-121	10	10	52.31	65.89	7	78.9%	
112A-109B	4	4	12.10	24.23	4	70%	
101-119	6	6	16.75	25.13	3	80%	
					Average ADL	71%	



(a) Experiment of known versus unknown

(b) Time duration during interfered versus normal

Figure 12: Experiment results of our proposed scheme.

4.2.2. Results. The experimental setup to validate our algorithms is based at Chongqing University Campus A, China. During these experiments, we choose two groups of students who have volunteered to participate. Students in group 1 were aware with the environment layout, and students of group 2 were not familiar with environment. Sensor devices were deployed in the building as shown in Figure 9. Red colored sensors in Figure 9 represent the motion sensors. TelosW sensor devices were deployed, and position of sensor in building is shown in Figure 8. Five tasks were performed in two experiments. In each task, participant needs to start from specific position and reaches destination through designed workplace. To achieve the fair result, the specified rooms and position were randomly chosen. The results are shown in Figure 10(b). After extracting 662 activates, we stored related information in each node by LCPA, LPA, PCA, and AADTA, and Table 6 shows the complete details.

Knowing and Unknowing. Students of group 1 were aware about the layout of designed setup, and they completed all six tasks without any prompting. Trajectories of group 1 are traced to detect abnormal activity at real time by using Algorithm 5 (AADTA). Students of group 1 involved in the same task are different from unaware participants of group 2 as shown in Table 7. In other words, unaware participants develop uncommon trajectories which were significantly different from pattern generated by aware group. After 14.5 seconds, it is clearly shown that it repels previous possibility of pattern 2 and it mismatches with other patterns shown in Figure 11(a). Therefore, such kind of activity is labeled with abnormal activity, and user's behavior pattern privacy can be protected by detecting such abnormal activity.

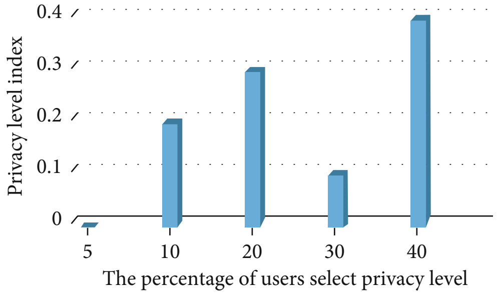
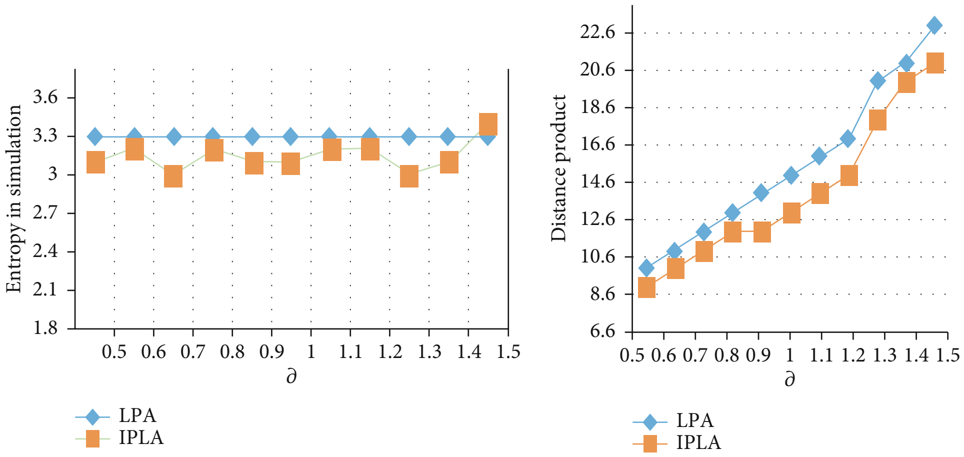


Figure 13: Result of experiment about privacy level of various users.

Normal Versus Abnormal. In the second experiment, it is required from participants of group 1 to stimulate a condition which we can label as abnormal condition. To create real abnormal situation, like as tumble, is hard to stimulate. Therefore, to generate abnormal phenomena, some disturbance such as by calling to a participant randomly while the task is being performed, are added in the experiment. After applying disturbed method, our algorithm detects abnormal activity at real-time occurrence without waiting for task's completion as shown in Figure 11(b). So our algorithm detects activity at real time instead of central computing in which abnormality is detected after completion of whole process of activity, and it enhanced the real-time performance. Hence, we found that our scheme protects privacy of user's behavior pattern by detecting abnormal activity at real time without waiting for completion of the process. Table 8 shows the result of abnormal activity detection by our proposed *BehaviorPrivacyDeg*. The



(a) ∂ versus entropy

(b) ∂ versus distance product

Figure 14: Effect of ∂ entropy and product of distance.

transition of participants shown in Figure 9 is $d_{26} \longrightarrow d_{24} \longrightarrow d_{22} \longrightarrow d_{34} \longrightarrow d_{40} \longrightarrow d_{21} \longrightarrow d_{19} \longrightarrow d_{17} \longrightarrow d_{13} \longrightarrow d_{10} \longrightarrow d_{16} \longrightarrow d_{21} \longrightarrow d_{14}$. When abnormality is detected at sensor device d_{19} by interfering the participants, the trajectory remains the same but time duration is significantly changed. Results in Figures 12(a) and 12(b) show the suitability and effectiveness of our scheme.

User Privacy-Level Concern Index. In this section, as discussed in privacy-level design model section, experiment result of our proposed MPCS showed that user's behavior pattern privacy is changed with the changing of privacy level. Privacy levels are configured by using index value on server. In Figure 13, index value showed that most users have much concerned about their activity privacy in smart IoT environment. After this, result revealed that users are more concerned about that area/zone and only 10 percent users are worried about their location. Hence, users can control their privacy level according to their concern by using our proposed MPCS.

Location Privacy. To protect the location of user in smart IoT environment, our proposed LPA achieved privacy of user's location by considering entropy and cloak region (CR). Users are required to share some level of personal information for getting services from server via installed sensor devices which are also called access point (AP).

We used a parameter ∂ to obtain partial information. In our experiments, we used 120 sensor devices which sense data and $\partial = 1.5$ represents the user familiarity about query probability over 75 APs. The effect of ∂ on entropy and product of distance are represented in results of our proposed LPA which is shown in Figures 14(a) and 14(b). In our simulation, $P = 15$, $r = 500$m, and change ∂ is from 0.5 to 1.5. The result revealed that location privacy algorithm (LPA) is better and has achieved the set target. The assessments of results showed that performance of LPA is better.

5. Conclusion

In this paper, we have proposed an effective multilevel privacy controlling scheme based on behavior pattern privacy degree and multilevel privacy design model. To protect the privacy of user's behavior pattern, we introduced *Behavior-PrivacyDeg* based on FPMTA, PCA, and AADTA. *Behavior-PrivacyDeg* focuses to mine, compress, store, and compute activities of user's behavior pattern by using proposed mining, compression algorithms, and storage mechanism. To detect abnormality and to protect the activity, we use the AADTA. Privacy levels are used for controlling method to protect users' behavior pattern. LCPA is used to configure the privacy level of users according to their concern and priority. PLA protects the privacy of user's location. PLA used entropy and cloak region (CR) to ensure privacy of location by spreading fake locations as far as possible. The experiments revealed the performance and feasibility of proposed MPCS. The scheme we proposed could provide a basis for behavior pattern privacy, LBS research, having the practical and theoretical significance on the study of trajectory anonymity, and location-based privacy preserving in smart IoT environment.

Data Availability

There is no data associated with the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of the article.

Authors' Contributions

A.K. and M.M.A.K devised the methodology and acquired funding. A.K. and M.A.J. carried out the formal analysis and data curation. A.K. and M.U.F. wrote the original draft, reviewed the writing, and edited the manuscript. A.A. and C.W. proofread the manuscript before its final submission. A.K, M.M.A.K, and M.A.J. contributed equally to this work.

Acknowledgments

The authors would like to thank the anonymous reviewers for a careful reading of this article and for all their comments, which led to a number of improvements in the article. This work is supported by the National Natural Science Foundation of China under grant No. 61672115, Guangzhou Government Project under grant No. 62104301, and Chongqing Social Undertakings and Livelihood Security Science and Technology Innovation Project Special Program (No. cstc2015jcyjBX0124 and No. cstc2017shmsA30003).

References

[1] P. Kumar, A. Braeken, A. Gurtov, J. Iinatti, and P. H. Ha, "Anonymous secure framework in connected smart home environments," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 968–979, 2017.

[2] K. Shin, X. Ju, Z. Chen, and X. Hu, "Privacy protection for users of location-based services," *IEEE Wireless Communications*, vol. 19, no. 1, pp. 30–39, 2012.

[3] N. Kaaniche and M. Laurent, "Data security and privacy preservation in cloud storage environments based on cryptographic mechanisms," vol. 111, Computer Communications Elsevier, 2017.

[4] L. Calderoni, P. Palmieri, and D. Maio, "Location privacy without mutual trust: the spatial Bloom filter," in *Computer communications*, vol. 68, pp. 4–16, Elsevier, 2015.

[5] R. Roshan and A. K. Ray, "Challenges and risk to implement IOT in smart homes: an Indian perspective," *International Journal of Computer Applications*, vol. 153, no. 3, pp. 16–19, 2016.

[6] P. Kumar, A. Gurtov, J. Iinatti, M. Ylianttilla, and M. Sain, "Lightweight and secure session-key establishment scheme in smart home environments," *IEEE Sensors Journal*, vol. 16, no. 1, pp. 254–264, 2016.

[7] B. Chikhaoui, S. Wang, and H. Pigot, "A frequent pattern mining approach for ADLS recognition in smart environments," in *Proceedings of the 25th IEEE International Conference on Advanced Information Networking and Applications (AINA '11)*, pp. 248–255, Biopolis, Singapore, March 2011.

[8] S. T. Peddinti, A. Dsouza, and N. Saxena, "Cover locations: availing location-based services without revealing the location," in *Proceedings of the 10th Annual ACM Workshop on Privacy in the Electronic Society - WPES '11*, New York, NY, USA, 2011.

[9] M. F. Mokbel, C.-Y. Chow, and W. G. Aref, "The new casper: query processing for location services without compromising privacy," in *Proceedings of the 32nd International Conference on Very Large Data Bases*, Seoul, Korea, 2006.

[10] B. Niu, X. Zhu, W. Li, and H. Li, "Epcloak: an efficient and privacy preserving spatial cloaking scheme for lbss," in *2014 IEEE 11th International Conference on Mobile Ad Hoc and Sensor Systems*, Philadelphia, PA, USA, 2014.

[11] G. Dini and P. Perazzo, "Uniform obfuscation for location privacy," in *Data and Applications Security and Privacy XXVI*, Springer, 2012.

[12] C. Y. Chow, M. F. Mokbel, and W. G. Aref, "Casper∗: query processing for location services without compromising privacy," *ACM Transactions on Database Systems*, vol. 34, no. 4, pp. 1–48, 2009.

[13] K. Fawaz and K. G. Shin, "Location privacy protection for smartphone users," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, New York, NY, USA, 2014.

[14] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Enhancing privacy through caching in location-based services," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, Hong Kong, China, 2015.

[15] H. Lu, C. S. Jensen, and M. L. Yiu, "Pad: privacy-area aware, dummy based location privacy in mobile services," in *Proceedings of the Seventh ACM International Workshop on Data Engineering for Wireless and Mobile Access - MobiDE '08*, New York, NY, USA, 2008.

[16] L. Fenghua, W. Sheng, N. Ben, and L. H. H. Yuanyuan, "Time obfuscation-based privacy-preserving scheme for location-based services," *Workshop on Physical-Layer Security: Rise, Fall and Rise Again Trilogy Toward Securing Data Networks*, 2016.

[17] X. Zhang, J. Wang, Q. Gao, X. Ma, and H. Wang, "Device-free wireless localization and activity recognition with deep learning," in *2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, pp. 1–5, Sydney, Australia, March 2016.

[18] Z. Liu, "Human activity recognition with 2D and 3D cameras," in *Progress in Pattern Recognition, Image Analysis, Computer-Vision, and Applications*, L. Alvarez, M. Mejail, L. Gomez, and J. Jacobo, Eds., vol. 7441 of Lecture Notes in Computer Science, p. 37, Springer, Berlin, Germany, 2012.

[19] V. Srinivasan, J. Stankovic, and K. Whitehouse, "Protecting your daily in-home activity information from a wireless snooping attack," in *Proceedings of the 10th International Conference on Ubiquitous Computing - UbiComp '08*, pp. 202–211, Seoul, South Korea, 2008.

[20] M. Gruteser and D. Grunwald, "Anonymous usage of location based services through spatial and temporal cloaking," in *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services*, pp. 31–42, New York, NY, USA, 2014.

[21] H. Ngo and J. Kim, "Location privacy via differential private perturbation of cloaking area," in *28th Computer Security Foundations Symposium*, Verona, Italy, 2015.

[22] G. Natesan and J. Liu, "An adaptive learning model for k- anonymity location privacy protection," in *39th Annual International Computers, Software and Applications Conference*, Taichung, Taiwan, 2015.

[23] T. Serge, B. Mickala, and Y. Younghwan, "User activity recognition in smart homes using pattern clustering applied to temporal ANN algorithm," *Sensors*, vol. 15, no. 5, pp. 11953–11971, 2015.

[24] J. Lu and W. Qi Yan, "Comparative Evaluations of Human Behavior Recognition Using Deep Learning," in *Multimedia Cyber Security Book*, p. 14, IGI Global Publisher of Timely Knowledge, 2020.

[25] M. A. K. Quaid and A. Jalal, "Wearable Sensors Based Human Behavioral Pattern Recognition Using Statistical Features and Reweighted Genetic Algorithm," in *Multimedia Tools and Applications*, Springer, 2020.

[26] W. Chengliang, Z. Qian, P. Yayun, D. Debraj, and S. Wen-Zhan, "Distributed abnormal activities detection in smart environments," *International Journal of Distributed Sensor Networks*, vol. 10, Article ID 283197, 2014.

[27] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, San Francisco, CA, USA, 3rd edition, 2011.

[28] M. Rose, M. Delgado, A. Vila, H. Hagras, and A. Bilgin, "A fuzzy logic approach for learning daily human activities in an ambient intelligent environment," in *Proceedings of the IEEE International Conference on bFuzzy Systems (FUZZ-IEEE '12)*, pp. 1–8, Brisbane, QLD, Australia, June 2012.

[29] A. Provotar and A. Lapko, "Fuzzy inference systems and their applications," *Cybernetics and Systems Analysis*, vol. 49, no. 4, pp. 517–525, 2013.