Energy and Buildings 154 (2017) 633–643

Contents lists available at ScienceDirect

Energy and Buildings

journal homepage: www.elsevier.com/locate/enbuild

Non-intrusive occupancy sensing in commercial buildings

Han Zou[a], Hao Jiang[b,*], Jianfei Yang[c], Lihua Xie[c], Costas J. Spanos[a]

[a] Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720, United States
[b] College of Electrical Engineering and Automation, Fuzhou University, Fuzhou 350108, China
[c] School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore

ARTICLE INFO

Article history:
Received 14 October 2016
Received in revised form 20 February 2017
Accepted 17 August 2017
Available online 1 September 2017

Keywords:
Occupancy sensing
Energy efficiency
WiFi

ABSTRACT

Buildings accounted for half of global electricity consumption in recent years. Accurate occupancy information could improve the energy efficiency and reduce the energy consumption in built environments. Although prior studies have explored various sensing techniques for occupancy sensing, these solutions still suffer from serious drawbacks, e.g. their estimated occupancy information is coarse, extra infrastructure is required, and the privacy of occupants is exposed. In this paper, we present the design and implementation of a novel and practical occupancy sensing system, WinOSS, which is able to provide fine-grained occupancy information thoroughly by leveraging existing commodity WiFi infrastructure along with the WiFi-enabled mobile devices carried by occupants. We have implemented WinOSS in a 1500 m^2 built environment for four weeks to validate its performance. Extensive experimental results demonstrate that WinOSS outperforms existing occupancy sensing techniques, and provides comprehensive fine-grained occupancy information (including occupancy detection, counting and tracking) in an accurate, reliable, cost-effective and non-intrusive manner.

1. Introduction

Nowadays, built environment consumes more than 50% of global electricity and is also responsible for 40% worldwide CO_2 emissions [1–3]. Commercial buildings are more energy-hungry than residential buildings due to their space layout and functionality. Lighting, heating, ventilation and air-conditioning (L-HVAC) systems are the most energy consuming components, which contribute over 70% of the total energy consumption in commercial buildings [4]. Thus, improving their energy efficiency is critical for reducing energy consumption and operation cost in commercial buildings. Prior studies have shown that huge amount of energy is wasted in unoccupied spaces since most of building management systems (BMSs) of commercial buildings operate based on static schedules [5]. On the other hand, the presence of occupants contribute to the heat gain directly to the built environment, and various occupancy activities, such as changing brightness of lights and temperature of HVAC terminal, will also affect the energy consumption significantly [6,7]. Therefore, precise occupancy information is one of the key prerequisites for energy efficient commercial buildings.

A solid body of work on occupancy sensing in the built environment has been conducted in the past decades [8–11]. The most common sensor for occupancy detection is passive infrared (PIR) sensor [8]. Being low-cost, it only provides coarse binary information (occupied or not) and fails to detect stationary occupants. Camera based occupancy sensing systems (OSSs) are able to track the movements of occupants precisely [9]. Nevertheless, its high computation cost for image processing and intrusiveness of privacy hamper it for large-scale applications. In recent years, with the wide availability of WiFi infrastructure as well as the pervasive WiFi enabled mobile devices (MDs), leveraging commercial off-the-shelf (COTS) WiFi routers along with MDs carried by occupants for occupancy sensing become possible and realizable [12]. However, existing WiFi-based OSSs require installing a dedicated App on the occupant's MDs to continuously perform active access point (AP) scanning for data acquisition, which introduces additional efforts on the user side and high battery consumption. Therefore, a practical WiFi-based OSS that is able to provide comprehensive fine-grained occupancy information is still crucially needed.

In this paper, we propose a novel **Wi**Fi-based **n**on-intrusive **O**ccupancy **S**ensing **S**ystem (**WinOSS**), which provides fine-grained occupancy information thoroughly using existing WiFi infrastructure in commercial buildings without intruding occupants. We upgrade the software of COTS WiFi APs so that they can capture the data packets in the existing WiFi traffic without introducing any extra infrastructure. Then, these APs analyze the packet

* Corresponding author.
E-mail address: jiangh@fzu.edu.cn (H. Jiang).

http://dx.doi.org/10.1016/j.enbuild.2017.08.045
0378-7788/

and extract the received signal strength (RSS) value and the MAC address of MD as identification of occupant from the packet, and send the information to a server. On the server side, we use online sequential extreme learning machine (OS-ELM) [13], as a fingerprinting-based machine learning localization algorithm to estimate the location of each MD as well as its owner (the occupant). Since the upgraded AP can obtain the real-time RSS readings of other APs as well, all the APs are natural online reference points. By leveraging the online sequential learning ability of OS-ELM, it is able to retrain the localization model when new RSS data are collected from APs. In this manner, WinOSS is capable of accurately inferring comprehensive occupancy information, including occupancy detection, counting and tracking, in complex and dynamic built environment.

We have implemented WinOSS in one entire floor (1500 m^2) of a commercial building and conducted extensive experiments to validate its performance. In general, according to the experimental results, WinOSS is able to achieve 98.85% occupancy detection accuracy, which is comparable or even better than the existing occupancy sensors such as PIR sensor. Furthermore, it is capable of inferring the occupancy level, trends and patterns precisely and timely in different indoor environments. Its average occupancy localization accuracy reaches 1.385 m. All the experimental results and performance evaluations have validated that WinOSS can provide comprehensive fine-grained occupancy information in an accurate, reliable, cost-effective and non-intrusive manner. It will definitely help BMS reduce the energy consumption and improve the energy efficiency of commercial buildings while enhancing the thermal comfort of occupants.

The rest of the paper is organized as follows. The related work is briefly reviewed in Section 2. Section 3 introduces the detailed system design of WinOSS, as well as its methodology of occupancy sensing. In Section 4, the testbed, data collection procedure, as well as the experimental procedure are elaborated. Section 5 presents and analyzes the experimental results and performance of WinOSS. We conclude this paper and discuss directions for future work in Section 6.

2. Related work

2.1. Occupancy sensing and monitoring

Various sensing technologies, such as PIR sensor [8,10], camera [9,14], radio frequency identification (RFID) [15–17], Inertial Measurement Unit (IMU) sensors [18,19], Bluetooth [20,21], ultrasound [22], WiFi [13,23,24], smart meters [25], environmental sensors [26,27], and sensor fusion [28,19,29], have been conducted to tackle the occupancy sensing problem in the past decades. In general, occupancy sensing can be categorized into three levels: occupancy detection, occupancy counting and occupancy tracking.

Occupancy detection: As the most basic level of occupancy inference, occupancy detection determines whether a predefined space is occupied or not. PIR motion sensors are one of the most commonly employed devices for occupancy detection [8]. It detects the presence of humans by measuring the variation in emitted radiation. Although these sensors are low power consuming, the output of their detection is binary and their detection accuracy is coarse-grained. Furthermore, it requires line-of-sight between the sensor and occupant. Moreover, they fail to detect relatively stationary occupants. Some prior works tried to exploit CO_2 sensors to infer whether space is occupied [30]. The large latency of the CO_2 sensor responses makes it unsuitable for real-time applications. Other environment sensors including temperature, humidity and acoustic sensors are also utilized for occupancy detection [31]. However, all of them introduce additional infrastructure costs.

Occupancy counting: The second level of occupancy inference is occupancy counting. Its objective is to determine the occupancy level (total number of occupants) in a predefined zone. The zone can be defined according to the physical layout or HVAC terminal coverage area. The occupancy level in each physical zone can be acquired from door access control systems in commercial buildings. However, its counting accuracy is not reliable when occupants come in or out together and only one of them opens the door using the door access card. While camera [9] is another information source for occupancy counting, the large computational cost for image processing and face recognition prevents it from large-scale applications. Moreover, the intrusiveness of occupant's privacy is another critical issue of using cameras.

Occupancy tracking: It is the highest level of occupancy inference. In addition to detecting and counting occupants, its goal is to estimate the location of each occupant in real-time which is similar to indoor localization. Techniques such as RFID [32], Bluetooth [33] and WiFi [34–36] have been proposed for indoor localization. RFID and Bluetooth require the occupant to carry an additional tag and install extra sensors for positioning purpose. On the other hand, WiFi has been acknowledged as the most promising alternative to GPS for indoor localization because COTS WiFi devices and infrastructures are widely available in indoor environments and most of the MDs are equipped with WiFi modules [37–40]. However, existing WiFi-based systems require a dedicated App to be installed on the occupant's MD to perform active AP scanning constantly for RSS data acquisition, which induces additional efforts on the user side as well as high battery consumption.

In summary, all the aforementioned occupancy sensing approaches still have certain drawbacks such as introducing extra hardware implementation, high cost for maintenance, large time delay, the intrusiveness of occupant's privacy and unreliable occupancy inference performance. Thus, an accurate, reliable, cost-effective and non-intrusive scheme for occupancy sensing is urgently desired.

2.2. Existing WiFi-based OSS

As presented in Section 2.1, a satisfactory OSS should not only be able to detect, count and track occupants precisely, but also meet certain performance criteria, including accuracy, reliability, cost-effectiveness and non-intrusiveness. Among the proposed sensing techniques, WiFi has certain unique merits for occupancy sensing, such as the wide availability of WiFi infrastructures in commercial buildings, and the prevalence of WiFi-enabled MDs.

Several WiFi-based Indoor Positioning Systems (IPSs), which can be used for occupancy sensing, have been proposed in recent decades [41–44]. The pioneering work of employing WiFi for indoor localization is RADAR [41]. By leveraging fingerprinting-based localization algorithm, it is able to achieve 2–3 m with a probability of 50%. Fingerprinting-based localization algorithm consists of two phases: offline training and online localization. The offline training phase involves a site survey process, in which RSSs from various APs at each calibration point (CP) and its physical coordinates form a fingerprint stored in an RSS database (a.k.a. offline radio map). During the online localization phase, the location of an MD is estimated by matching the observed RSS readings against the offline RSS fingerprint database. Horus [42] stores the RSS distribution of each AP in the fingerprint database and makes use of a probabilistic model to perform indoor localization. [45] propose a novel algorithm that is able to locate heterogeneous MDs carried by occupants with consistent high localization accuracy. Although fingerprinting-based approach is the most popular for WiFi-based IPS because it can to capture the odd RSS distribution in complex indoor environments [43,44], it still suffers from a major issue: the vulnerability to environmental dynamics, as the real-time RSS readings collected during online localization phase could deviate from those stored in the offline radio map due to the variation of temperature, humidity, occupancy distribution and multipath effects. Serious localization errors will be introduced if the radio map is not updated accordingly.

With the satisfactory localization performance of WiFi-based IPSs, some researchers have proposed to make use of them for occupancy sensing [46,12]. For instance, a WiFi-based occupancy detection and counting system that can achieve zone-level accuracy is proposed in [46]. [12] leverage Authentication, Authorization and Accounting (AAA) WiFi logs from WiFi APs to infer occupancy. With this data, the authors claim that 17.8% of energy consumption of HVAC system can be saved. A WiFi-based OSS that integrates WiFi signals with calendar schedules of occupants is proposed [47]. Sensor fusion of WiFi with smartphone equipped sensors, including magnetometers accelerometers and gyroscopes, is also proposed to improve the occupancy tracking performance in [48]. However, these existing WiFi-based OSSs require occupants to install a dedicated App on their MDs for RSS data acquisition, which definitely introduce additional efforts on the user side and high battery consumption. Furthermore, the sampling rate of data acquisition is usually low due to the hardware limitation of MDs. Moreover, these WiFi-based OSSs are not able to detect occupants who carry iOS based MDs because Apple Inc. has recently suspended any RSS open API for active AP scanning. In a nutshell, issues from both localization algorithm and system perspectives have hindered existing WiFi-based OSSs for large-scale implementation in commercial buildings.

3. System design

3.1. System overview

In order to make the OSS practical for large-scale implementation, WinOSS is designed to satisfy the following criteria: (1) ***comprehensive***: the system should be able to conduct occupancy sensing in all aspects exhaustively, including occupancy detection, counting and tracking; (2) ***agile***: the system should be capable of inferring occupancy information effectively; (3) ***non-intrusive***: the occupancy sensing mechanism aims to perverse the privacy of occupants and should not require the active participation from them.

According to these design criteria, we design a WiFi-based Non-intrusive Occupancy Sensing System (WinOSS), an intelligent wireless system that is able to overhear the RSS data packages in the existing WiFi traffic in real-time without any intrusiveness on occupant side and precisely detect, count and track occupants successfully. The following sections will introduce the system architecture of WinOSS, and the methodology of occupancy sensing by WinOSS.

3.2. WinOSS

In general, WinOSS is built on a wireless distribution system (WDS) and can be directly implemented on most of the COST WiFi routers that support OpenWrt [49]. It is able to create a WiFi LAN and provide Internet services for occupants as a basic function of WDS. More importantly, WinOSS enables COTS WiFi routers to overhear the data packets transmitted between each MD and routers, and precisely retrieve the RSS values and corresponding MAC addresses as identifiers of occupants, and forward this information to a back-end server for occupancy sensing in a non-intrusive manner.

Fig. 1 presents the system architecture of WinOSS. The main components include COTS WiFi APs, a back-end server, and

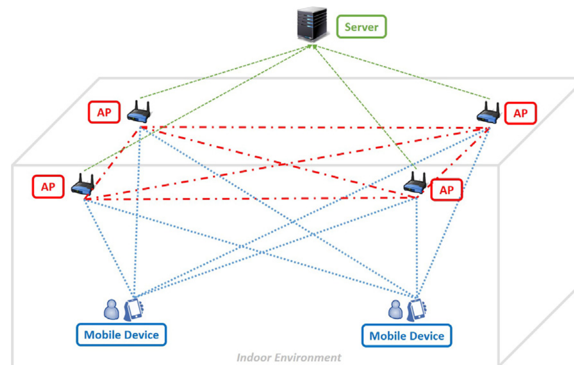


Fig. 1. System architecture of WinOSS.

occupants and their MDs. All the APs perform the following major tasks: capture the 802.11*n* data packets in the network, extract relevant information from the packets, arrange it in a particular format and forward them to the back-end server. The back-end server is responsible for parsing the data sent from APs and building up the online RSS fingerprint database for localization. We upgrade the firmware of APs with OpenWrt and add a designed software based on Lipcap to sniff the existing WiFi traffic, and enable the APs to capture as well as analyze the data packets. Compared to the existing data acquisition method which relies on the MD to perform active scanning at a low sampling rate due to its hardware limitation, these APs are able to overhear and collect the packets at a maximum rate around 100 packets per second. Furthermore, since WinOSS opportunistically captures the data packets from existing WiFi traffic generated by various existing Apps on MDs, such as data stream from watching videos, periodic email fetching and push notification services, it poses no additional burden on the battery life of the MD.

For each AP, in addition to capturing the data packets sent and received by each MD, it is also capable of capturing the packets of other APs since they generate wireless traffic as they operate. Therefore, the RSS measurements at these APs can be viewed as the signal strength of nearby MDs, and these measurements are also affected by the same environmental dynamics and change over time. In this manner, all the APs can be used as natural online reference points for RSS radio map adaptation to overcome the vulnerability issue to environmental dynamics, because we are able to obtain their real-time RSS readings and physical coordinates. Fig. 2

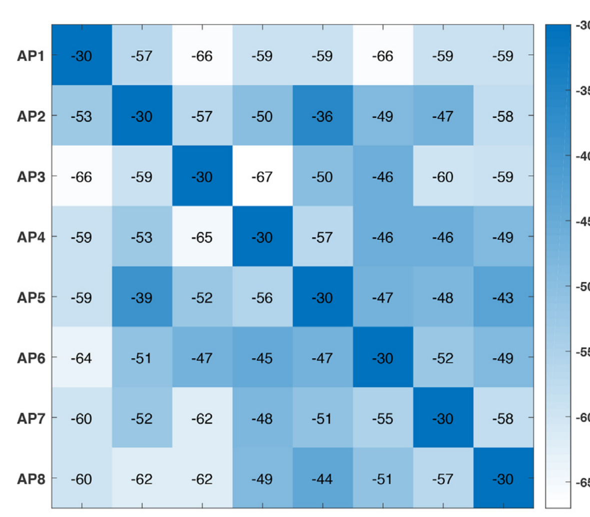


Fig. 2. Visualization of pairwise RSS matrix of 8 APs (dBm).

demonstrates the visualized pairwise RSS of 8 APs. In principle, each AP cannot sense the signal strength of itself. Therefore, we calibrate the average RSS of two APs placed side-by-side and assign −30 dBm as the self-sensed RSS to complete the pairwise RSS matrix of APs as shown in Fig. 2.

3.3. Occupancy sensing by WinOSS

In this section, we present the methodology of using the RSS data collected by WinOSS to estimate the location of MD as well as its user (the occupant).

As mentioned in Section 2.2, the fingerprinting-based algorithm has been extensively adopted in WiFi-based IPS due to its ability to capture the signal variations more accurately than other algorithms in complex indoor environments [44,43]. We adopt online sequential extreme learning machine (OS-ELM), for occupancy tracking. As an adaptive localization algorithm, OS-ELM is proposed in [24], which is able to provide good generalization performance at an extremely fast learning speed [50], and has an online sequential learning ability [51] that does not require retraining when new RSS data are collected. Serving as a fingerprinting-based machine learning localization algorithm, OS-ELM contains three phases: offline initialization training phase, online sequential learning phase and localization phase.

3.3.1. Offline initialization training phase

The objective of the initialization training phase is to train an initial model for localization. Assume that there are P APs installed in an indoor environment and a sum of D_0 WiFi RSS fingerprints are collected by WinOSS at numerous calibration points (CPs). These RSS fingerprints and their physical coordinates are adopted as training inputs and training targets respectively to build up the initial OS-ELM model. Each sample can be represented as $(\mathbf{s}_m, \mathbf{t}_m) \in \mathbb{R}^{D_0} \times \mathbb{R}^2$, where the training input $\mathbf{s}_m = [RSS_m^1, RSS_m^2, \ldots, RSS_m^P]$ is a vector of RSSs received from P APs, and training target $\mathbf{t}_m$ is the 2-D physical coordinates of the CP. Assume that a single-hidden layer feedforward neural network (SLFN) with L hidden nodes can approximate these D_0 training samples with zero error, or equivalently

$$f_L(\mathbf{s}_m) = \sum_{l=1}^{L} \beta_l G(\mathbf{a}_l, b_l, \mathbf{s}_m) = \mathbf{t}_m, \quad m = 1, 2, \ldots, D_0, \tag{1}$$

where $\mathbf{a}_l$ and b_l are the learning parameters of the hidden nodes, β_l is the output weight, and $G(\mathbf{a}_l, b_l, \mathbf{s}_m)$ is the activation function which yields the output of the l-th hidden node with respect to the input $\mathbf{s}_m$. There are three steps to build up the initial OS-ELM model:

Step 1: Randomly assign the input parameters: input weights $\mathbf{a}_i$ and bias b_i, $i = 1, \ldots, L$.
Step 2: Calculate the initial hidden layer output matrix $\mathbf{H}_0$

$$\begin{aligned}\mathbf{H}_0 &= \begin{bmatrix} G(\mathbf{a}_1, b_1, \mathbf{s}_1) & \ldots & G(\mathbf{a}_L, b_L, \mathbf{s}_1) \\ \vdots & \ldots & \vdots \\ G(\mathbf{a}_1, b_1, \mathbf{s}_{D_0}) & \ldots & G(\mathbf{a}_L, b_L, \mathbf{s}_{D_0}) \end{bmatrix}_{D_0 \times L} \\ &= \left[\mathbf{h}(\mathbf{x}_1)^T, \mathbf{h}(\mathbf{x}_2)^T, \ldots, \mathbf{h}(\mathbf{x}_{D_0})^T \right]^T_{D_0 \times L}.\end{aligned}$$

Step 3: Estimate the initial output weight $\beta^{(0)}$. According to the analysis in [43], the Hardlim function is more suitable than others and hence chosen as the activation function for OS-ELM modeling in this work. To estimate $\beta^{(0)}$ is equivalent to solving a least squares problem as follows:

$$\begin{aligned} \min_{\boldsymbol{\xi}, \boldsymbol{\beta}^{(0)} \in \mathbb{R}^{L\times 2}} \quad & J = \sum_{m=1}^{D_0} \xi_m \\ \text{s.t.} \quad & \xi_m = \|\mathbf{h}(\mathrm{x}_m)\boldsymbol{\beta}^{(0)} - \mathbf{t}_m\|^2 \quad m = 1, 2, \ldots, D_0, \end{aligned} \tag{2}$$

and noticing that the Moor-Penrose generalized inverse of $\mathbf{H}^{\dagger} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T$ [50]. Therefore, the optimal solution is given by $\beta^{(0)} = \mathbf{P}_0\mathbf{H}_0^T\mathbf{T}_0$, where $\mathbf{P}_0 = (\mathbf{H}_0^T\mathbf{H}_0)^{-1}$ and $\mathbf{K}_0 = \mathbf{P}_0^{-1} = \mathbf{H}_0^T\mathbf{H}_0$.
Step 4: Set $k = 0$, where k is a parameter indicating the number of chunks of data that are presented to the network.

3.3.2. Online sequential learning phase

The main purpose of the online sequential learning phase of OS-ELM is to realize consistent accurate occupancy tracking over various environmental dynamics [13]. Since WinOSS enables COTS APs to become online reference points, we can always obtain the online pairwise RSS data among APs in real-time. These RSS readings and the corresponding physical coordinates of APs are adopted as new training samples D_{AP}^{k+1} for the $(k+1)$-th OS-ELM online sequential learning to further adapt the environmental dynamics. The revised OS-ELM model will be obtained by the following steps:

Step 1: Calculate the partial hidden layer output matrix $\mathbf{H}_{k+1}$ =

$$\begin{bmatrix} G(\mathbf{a}_1, b_1, \mathbf{s}_1^{k+1}) & \ldots & G(\mathbf{a}_L, b_L, \mathbf{s}_1^{k+1}) \\ \vdots & \ldots & \vdots \\ G(\mathbf{a}_1, b_1, \mathbf{s}_{D_{AP}}^{k+1}) & \ldots & G(\mathbf{a}_L, b_L, \mathbf{s}_{D_{AP}}^{k+1}) \end{bmatrix}_{D_{AP}^{k+1} \times L} \tag{3}$$

Step 2: Calculate the output weight $\beta^{(k+1)}$. We have

$$\mathbf{K}_{k+1} = \mathbf{K}_k + \mathbf{H}_{k+1}^T\mathbf{H}_{k+1} \tag{4}$$

$$\beta^{(k+1)} = \beta^{(k)} + \mathbf{K}_{k+1}^{-1}\mathbf{H}_{k+1}^T(\mathbf{T}_{k+1} - \mathbf{H}_{k+1}\beta^{(k)}), \tag{5}$$

and in order to avoid inverting matrices, such as $\mathbf{K}_{k+1}^{-1}$ in Eq. (5) in the recursive process, the Woodbury formula [52] is applied to transform the equations to $\mathbf{K}_{k+1}^{-1} = \mathbf{K}_k^{-1} - \mathbf{K}_k^{-1}\mathbf{H}_{k+1}^T(\mathbf{I} + \mathbf{H}_{k+1}\mathbf{K}_k^{-1}\mathbf{H}_{k+1}^T)^{-1}\mathbf{H}_{k+1}\mathbf{K}_k^{-1}$. Since $\mathbf{P}_{k+1} = \mathbf{K}_{k+1}^{-1}$, $\mathbf{P}_{k+1} = \mathbf{P}_k - \mathbf{P}_k\mathbf{H}_{k+1}^T(\mathbf{I} + \mathbf{H}_{k+1}\mathbf{P}_k\mathbf{H}_{k+1}^T)^{-1}\mathbf{H}_{k+1}\mathbf{P}_k$. Therefore, the output weight $\beta^{(k+1)}$ is calculated as follows

$$\beta^{(k+1)} = \beta^{(k)} + \mathbf{P}_{k+1}\mathbf{H}_{k+1}^T(\mathbf{T}_{k+1} - \mathbf{H}_{k+1}\beta^{(k)}) \tag{6}$$

Step 3: Set $k = k + 1$ for the next online sequential learning phase.

3.3.3. Localization phase

Before any WiFi RSS readings are collected by WinOSS, this initial OS-ELM model is utilized to estimate the location of each occupant. When the pairwise RSS data among APs D_{AP} are obtained by WinOSS, they are integrated into the initial OS-ELM model to generate a revised OS-ELM model as illustrated in Section 3.3.2. Afterwards, the location of each occupant is estimated by feeding the RSS readings into the latest revised OS-ELM model. In this manner, WinOSS is able to precisely track multiple occupants simultaneously in dynamic indoor environments, as long as they turn on the WiFi function of their MDs which is a common user habit for most occupants.

Fig. 3. Layout of the testbed.

4. Implementation

4.1. System setup

To validate the occupancy sensing capability of WinOSS, we implemented and tested WinOSS in the entire headquarter of Berkeley Education Alliance for Research in Singapore (BEARS), which is a 1500 m^2 multi-functional office and lab located in the 11th floor of CREATE Tower, National University of Singapore. As depicted in Fig. 3, the area is composed of two large open space offices, seven personal offices, three conference rooms, one living space and one dry laboratory. We deployed 22 TP-LINK N750 router, as APs to construct the sensing network of WinOSS. The firmware of these routers was upgraded to OpenWrt with our software for RSS data collection. To train the initial OS-ELM localization model, RSS samples of a Google Nexus 6 were collected at 80 CPs over the entire area during the offline initialization training phase. 100 RSS samples were collected at each location. Since we knew the physical coordinates of all the routers, the real-time pairwise RSS readings among routers as well as their coordinates were adopted as new training samples for OS-ELM online sequential learning to update the localization model. As presented in Section 3.3, WinOSS estimates the locations of occupants by feeding the RSS readings of their MD into the latest revised OS-ELM localization model.

4.2. Experimental study

We chose the four chambers in the dry laboratory (labeled as Lab 1–4 as shown in Fig. 3), and the living space and one open space office (labeled as Office 1–4 as shown in Fig. 3) to evaluate the occupancy detection performance of WinOSS. The size of each chamber in the dry laboratory is 4 m × 6 m. Since these chambers have fully controlled air handling and sufficient thermal insulation, researchers usually conduct controlled experiments to evaluate the thermal comfort of occupants in distinct scenarios. The living space is a common feature of large commercial office buildings because it facilitates communications and increases interaction between employees. Furthermore, it could also facilitate organizations to save on the cost of HVAC, lighting and maintenance.

We also conducted a 4-week experiment in the pre-defined 8 zones to evaluate the accuracy of occupancy counting by WinOSS. 32 researchers participated in this experiment and turned on the WiFi module of their smartphones consistently. Meanwhile, WinOSS only scanned these smartphones as identifiers of occupants during the experimental period. The accurate ground truth of occupancy level in the lab was obtained by logging zonal occupancy levels based on image frames captured by a CCTV system in our dry laboratory. Four undergraduates were engaged to manually record the occupancy level in Office 1–4 in an event-trigger manner, i.e., counting and marking targets down when zonal population changed.

In addition to the evaluation of occupancy detection and counting performance of WinOSS, we also validated its localization accuracy since precise occupancy tracking would enhance a wide variety of applications, such as navigation, augmented reality, location-aware pervasive computing, targeted advertising, social networking, and location-aware BMS control [53]. To make a thorough evaluation of the occupancy tracking performance of WinOSS, 32 testing points (4 in each zone) were randomly selected and 100 samples were collected at each testing point around 1.5 min.

5. Discussion

In this section, we present and analyze the experimental results of the three levels of occupancy sensing performance of WinOSS, namely occupancy detection, counting and tracking.

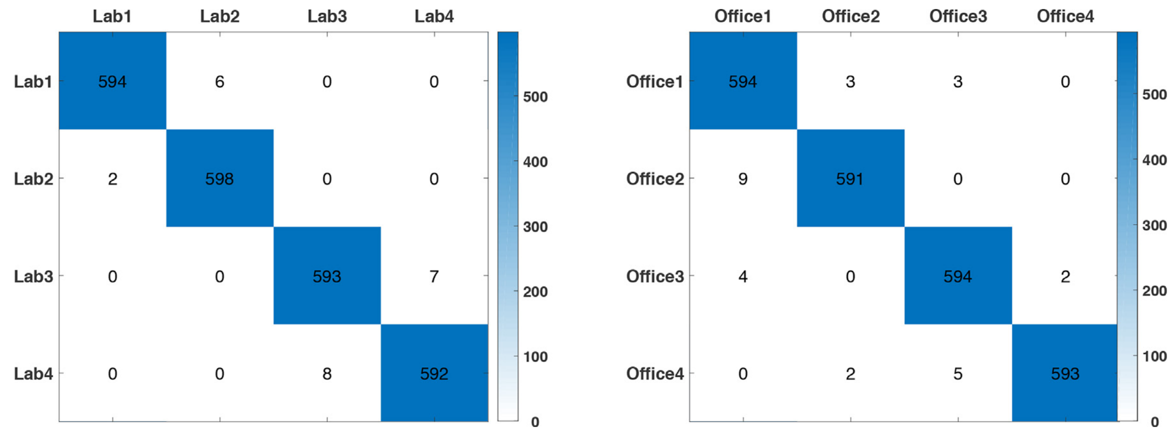


Fig. 4. Confusion matrix of occupancy detection accuracy of WinOSS in lab and office area.

5.1. Occupancy detection accuracy

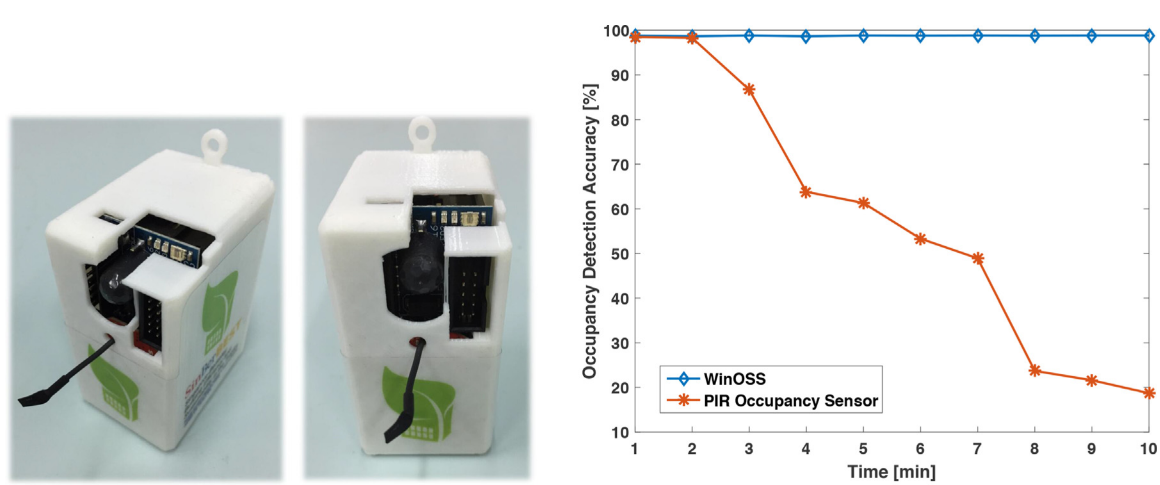
To evaluate the occupancy detection accuracy of WinOSS, one occupant walked around each of eight zones for 10 min (detection rate: per second). According to the experimental results, WinOSS is able to achieve 98.85% detection accuracy on average, which is comparable to or even better than the performance of existing occupancy sensors such as PIR motion sensor. Fig. 4 depicts the detection confusion matrix of WinOSS in the lab area and office area. Its detection accuracy in office area (98.66%) is slightly inferior to the lab area (99.04%). The reason is that the walls in the lab area are built from concrete while the four zones in office area are divided virtually. Therefore, when the occupant is walking close to the boundary of the virtual zone in the office area, he or she could be misclassified to the nearby zones. This issue can be overcome by performing a more fine-grained offline calibration at the boundary area of each zone to improve the detection accuracy.

In addition to testing the detection performance when the occupant is moving, we also validated its detection accuracy when the occupant is relatively stationary since existing occupancy sensor, such as PIR motion sensor, usually fails to capture stationary occupants. In this experiment, one occupant stayed in each zone and remained stationary for 10 min. As shown in Fig. 5, One Building-in-Briefcase (BiB) sensor [26], which is equipped with an AMN41121 PIR motion detector, was placed beside a ceiling lamp in each zone as an occupancy sensor to compare with WinOSS. Fig. 5 demonstrates the comparison of average detection accuracy in 8 zones between BiB sensor and WinOSS during a 10-min experiment. As shown in Fig. 5, WinOSS is able to detect the occupant consistently even he is relatively stationary. This is because WinOSS estimates the location of each occupant every second regardless the activity of the occupant. On the other hand, the PIR motion detector failed to detect the presence of a stationary occupant after 3 min because its detection mechanism requires continuous motion of occupants to function effectively.

In summary, WinOSS is able to detect the presence of occupant accurately and consistently regardless the activity of occupant (dynamic or static), by just leveraging the existing WiFi infrastructure.

5.2. Occupancy counting accuracy

The previous section has illustrated that WinOSS is capable of detecting the presence of occupant precisely. In this section, we further investigate the occupancy counting performance of WinOSS. As introduced in Section 4.1, 32 researchers participated in a 4-week experiment. To quantify its occupancy counting accuracy comprehensively, three performance metrics are adopted: (1) Normalized root mean square deviation (**NRMSD**): NRMSD is leveraged to evaluate the overall accuracy of WinOSS. Given a time series T



(a) (b)

Fig. 5. (a) Building-in-Briefcase (BiB) sensor (equipped with an AMN41121 PIR motion detector). (b) Occupancy detection accuracy comparison between WinOSS and PIR occupancy sensor.

Table 1
Occupancy counting accuracy of WinOSS.

Zone ID	Lab 1	Lab 2	Lab 3	Lab 4	Office 1	Office 2	Office 3	Office 4	Overall
NRMSD	0.037	0.106	0.079	0.099	0.145	0.020	0.170	0.111	0.096
Mean	0.015	0.040	0.025	0.035	0.196	0.560	0.386	0.312	0.196
STD	0.185	0.317	0.235	0.296	0.700	1.111	0.946	0.833	0.578

NRMSD: normalized root mean square deviation; Mean: expectation of the occupancy counting error; STD: standard deviation of the occupancy counting error.

of estimated occupancy level $s_{1:T}$ and ground truth data $g_{1:T}$, the NRMSD between $s_{1:T}$ and $g_{1:T}$ is defined as

$$NRMSD(s_{1:T}, g_{1:T}) = \frac{\|s_{1:T} - g_{1:T}\|/\sqrt{T}}{\max(s_{1:T}) - \min(s_{1:T})}$$

where $\|\cdot\|$ is the Euclidean norm, $max(\cdot)$ and $min(\cdot)$ represent the maximum and minimum value of the time series T. The other two metrics are (2) **mean** (expectation of the occupancy counting error) and (3) **STD** (standard deviation of the occupancy counting error).

Table 1 summarizes the three occupancy counting performance metrics of WinOSS in the 8 pre-defined zones during the entire experiment. The overall NRMSD of WinOSS is only 0.096, which is outstanding and competitive compared with traditional occupancy counting system. The small mean estimation error (0.196) and STD (0.578) justify its accurateness and robustness.

Fig. 6 demonstrates the occupancy counting comparison of WinOSS and the ground truth during a period of four weeks. Comparing with the ground truth, WinOSS is able to capture the general occupancy trends accurately. It clearly presents the discrepancy of occupancy level between weekdays and weekends, as well as working hours and off-duty hours on weekdays. On weekdays, most occupants come to the office around 9 am. There is a decrease in occupancy level around noon indicating that occupants usually go out for lunch at that time. Majority of occupants leave the office around 6 pm and the office usually remains unoccupied from 8 pm until 8 am on next weekday. It is worth noticing that WinOSS can infer the incidents of occupancy variations effectively. For instance, as shown in Fig. 6, unlike other weekdays, WinOSS indicates that the occupancy level increases at noon on every Thursday because our research center organizes a gathering lunch at that time every week and all the researchers come to the open space for lunch and discussion. Furthermore, it also successfully detects the low occupancy level on Monday of Week 4 (September 12th, 2016) because it is a public holiday in Singapore. On weekends, according to the data collected by WinOSS, although the occupancy is reasonably low during most of the time, some researchers still come to the office during weekends. Thus, they need to turn on the light and air conditioning manually because the default setting of HVAC system is turned off according to the predetermined static schedule.

As demonstrated in Fig. 6, the occupancy level displays similar periodic patterns every week but still there are certain differences. Static scheduling for lighting and HVAC control is definitely not the optimal strategy for BMS in commercial buildings. On the other hand, according to our observations, the real-time variation of occupancy level captured and inferred by WinOSS provides extremely vital information to realize occupancy adaptive HVAC and lighting control and indicate the great potential for sustainable energy savings.

We also analyze the detailed occupancy counting performance of WinOSS in various types of indoor environments. For instance, Figs. 7 and 8 illustrate the occupancy trends on one Friday in each lab zone and office zone respectively. As shown in Fig. 7, most of the labs are occupied by fewer than 4 researchers and the certain amount of time those labs are unoccupied. The reason is that researchers are conducting controlled experiments to test the performance of HVAC or air quality in these chambers. Therefore, the occupancy trigger based lighting control is more appropriate in lab area than static scheduled based one. On the other hand, as shown in Fig. 8, the occupancy level in the office area, especially in the open space (Office 1 + Office 2) is more variable. Based on this insight obtained from WinOSS, the BMS for the open space should be occupancy trigger based since occupants are gathering in this area for various activities such as group discussions, lunch parties, and seminars.

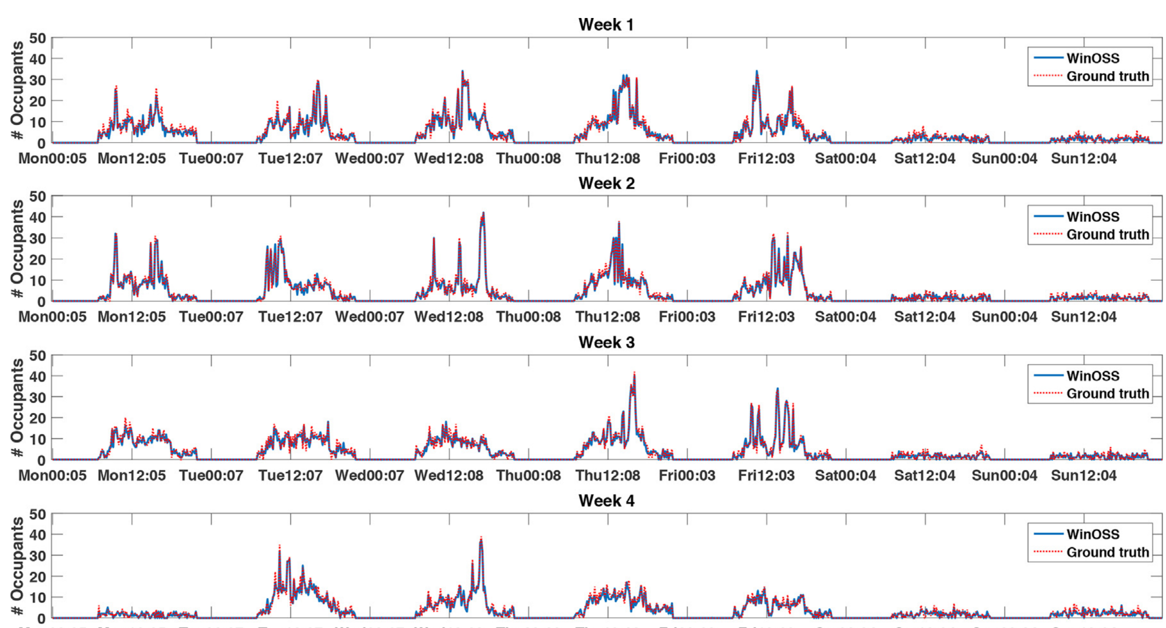


Fig. 6. Comparison of occupancy level inferred by WinOSS and the ground truth.

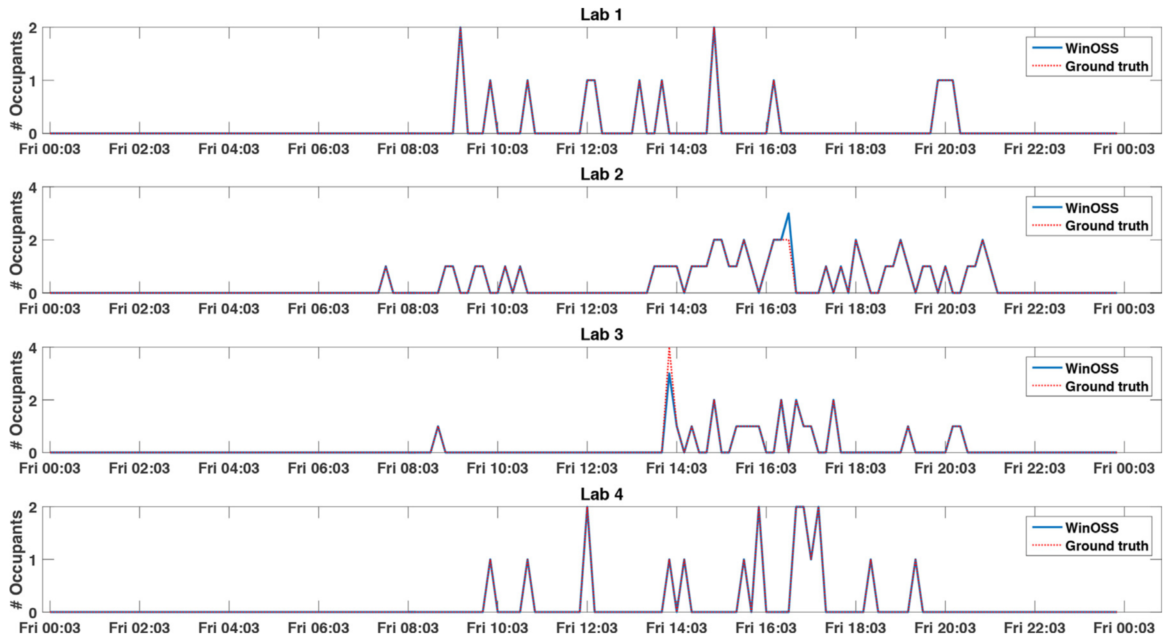


Fig. 7. Occupancy level inferred by WinOSS in Lab area.

In summary, the experiment results and analysis have validated that WinOSS is capable of inferring the occupancy level, trends and patterns in an accurate and effective manner in common indoor environments.

5.3. Occupancy tracking accuracy

Sections 5.1 and 5.2 have fully demonstrated the outstanding occupancy detection and counting performance of WinOSS. As introduced in Section 3.3, WinOSS is also able to precisely estimate the location of each occupant to realize occupancy tracking in real-time by just employing existing WiFi infrastructure. To measure the localization accuracy, we define the location estimation error e to be the distance between the real location coordinates (x_0, y_0) and the system estimated location coordinates (x, y), i.e., $e = \sqrt{(x - x_0)^2 + (y - y_0)^2}$.

Table 2 summarizes the localization accuracy in each zone. In general, by leveraging the OS-ELM localization algorithm, WinOSS is able to achieve 1.385 m localization accuracy on average without any intrusiveness on user side or deploying extra infrastructure for occupancy tracking. The high localization accuracy in the lab (1.376 m) and the office (1.394 m) validates that WinOSS can provide precise and reliable occupancy tracking service in different types of built environments. Fig. 9 demonstrates the cumulative percentile of localization error in 8 zones. It is worth noticing that the 75% percentile error of WinOSS is only 1.800 m which is good enough for occupancy based light control and HVAC control. The detailed localization error on each testing point (including 16 points in lab area and 16 points in office area) is demonstrated in Fig. 10.

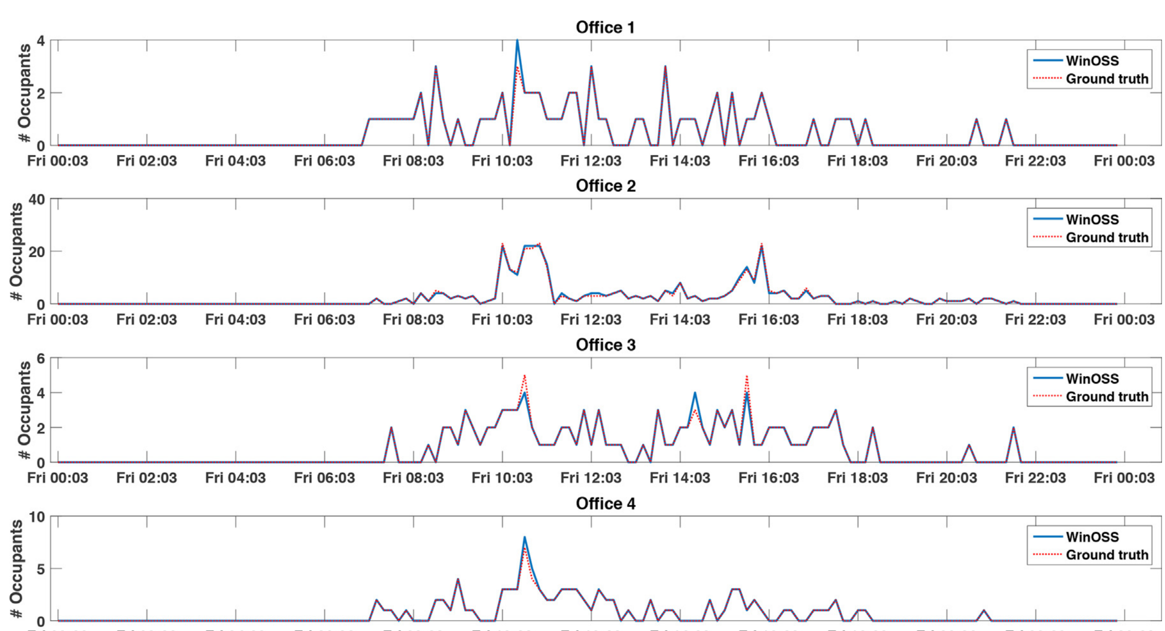


Fig. 8. Occupancy level inferred by WinOSS in Office area.

Table 2
localization accuracy in Lab and Office area (m).

Accuracy (Mean ± STD)	Zone 1	Zone 2	Zone 3	Zone 4	Overall
Lab area	1.339 ± 0.728	1.516 ± 0.614	1.476 ± 0.796	1.173 ± 0.667	1.376 ± 0.719
Office area	1.630 ± 0.694	1.383 ± 0.670	1.258 ± 0.605	1.305 ± 0.714	1.394 ± 0.686
Total					1.385±0.703

Lab area: the four lab zones as shown in Fig. 3; Office area: the four office zones as shown in Fig. 3; Total: average localization error of Lab and Office areas.

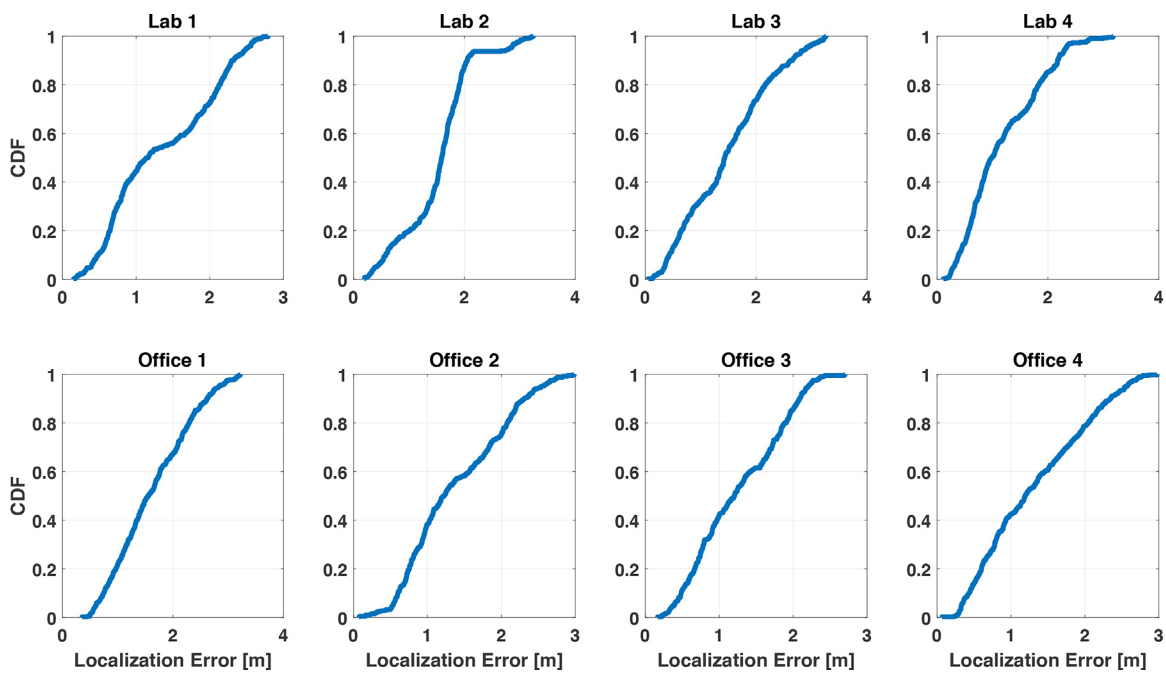


Fig. 9. CDF of localization error in each zone of Lab and Office area.

As shown in this figure, the localization performance of WinOSS is more consistent in the office area than the lab area because the layout of the lab is more complicated which introduces more multi-path effect on signal propagation.

In summary, WinOSS can achieve high occupancy tracking accuracy by just employing the COTS WiFi routers along with MDs carried by the occupants, as well as the OS-ELM localization algorithm. Since it is able to track the location of each occupant

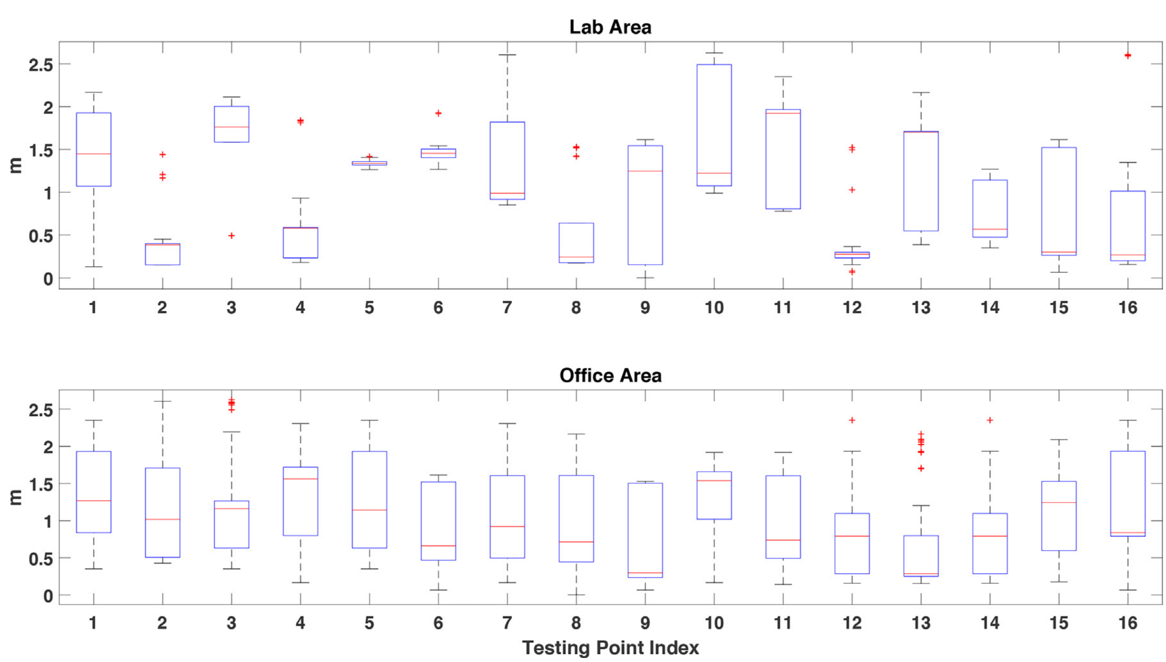


Fig. 10. Localization accuracy at each testing point in Lab and Office area.

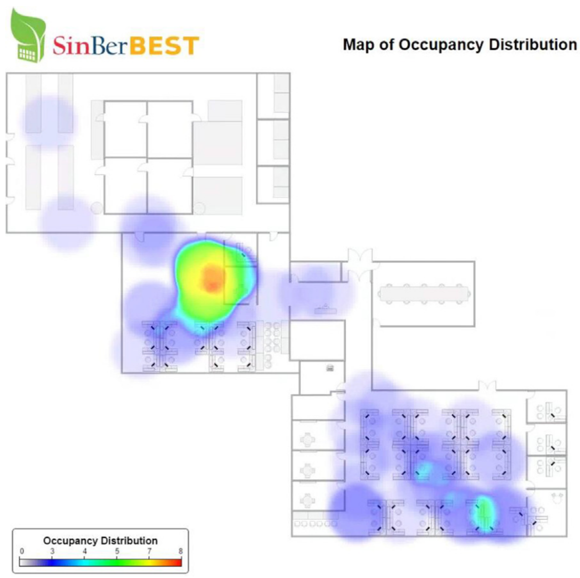


Fig. 11. Occupancy distribution captured by WinOSS.

simultaneously, the occupancy distribution in buildings can be easily obtained as well. Fig. 11 demonstrates the occupancy distribution of 1500 m^2 office area captured by WinOSS to show its feasibility of building-wide implementation.

5.4. Limitations of WinOSS

The experimental results and analysis presented in the aforementioned three sections have validated that WinOSS is capable of providing fine-grained occupancy information comprehensively, including detection, counting and tracking, in commercial buildings. In this section, we discuss the limitations of WinOSS as well as the potential solutions.

In WinOSS, we assume that occupants carry their personal MDs when they are inside the buildings, and the positions of their MDs are considered as their locations. However, there are some special circumstances. For instance, sometimes occupants forget to bring their MDs or the WiFi module of their MDs are accidentally turned off when they are inside buildings. In this case, we plan to develop a web-based interface to interact with them through their laptops or PCs in order to check and confirm with their presence. Furthermore, occupants may also forget to carry their MDs when they go to the toilet or conference rooms. In this situation, by analyzing the historical activity patterns of each occupant captured by WinOSS, we could predict and infer their activities and then confirm with them through the interface to ensure correct occupancy information are obtained and feedback to BMS. WiFi-based device-free occupancy sensing system [54] can be integrated with WinOSS to compensate the performance when occupants forget to carry their MDs. Other special case is when one individual carries multiple MDs, such as one phone and one tablet, in buildings and WinOSS will assume that two occupants are detected. To overcome this issue, firstly we will associate the MAC addresses of each occupant's personal MDs with them and store this data on the server. Then, if the estimated positions of multiple personal MDs, which link to an identical occupant, are the same, only one will be considered as his or her real-time location instead of counting two distinct individuals.

Another potential issue is that some MDs adopt sleep mode to improve MD's battery life which turn off WiFi if occupants do not actively use their MDs. To address this issue, we plan to develop an APP running on MDs that could send packets to near APs periodically so it could wake up the MDs in sleep mode when occupants are inside the building. In this manner, WinOSS is able to detect their presence all the time because MDs continuously generate WiFi traffic.

6. Conclusion and future work

In this paper, we proposed, WinOSS, a novel WiFi-based non-intrusive occupancy sensing system that provides comprehensive fine-grained occupancy information using existing WiFi infrastructure along with the WiFi-enabled MDs carried by the occupants. By analyzing the WiFi traffic between MDs and COTS APs, and leveraging the online sequential learning ability of OS-ELM, WinOSS is able to precisely and effectively infer occupancy information, including occupancy detection, counting and tracking, in complex and dynamic built environment. We have implemented and tested WinOSS in a real-world office area and experimental results demonstrate that: (1) It is able to achieve 98.85% occupancy detection accuracy even when occupants stay stationary; (2) It is capable of inferring the occupancy level with 0.096 NRMSD in various types of indoor environments; (3) It can provide 1.385 m occupancy tracking accuracy without introducing any intrusiveness on user side or deploying extra infrastructure. In summary, the comprehensive occupancy information provided by WinOSS can contribute toward substantial energy saving of commercial buildings.

For the future work, we plan to integrate WinOSS with BMS, such as lighting control system and HVAC system, to realize occupancy based adaptive control and personalized thermal comfort for each occupant.

Acknowledgments

This research is partially funded by the Republic of Singapore National Research Foundation (NRF) through a grant to the Berkeley Education Alliance for Research in Singapore (BEARS) for the Singapore-Berkeley Building Efficiency and Sustainability in the Tropics (SinBerBEST) Program. BEARS has been established by the University of California, Berkeley as a center for intellectual excellence in research and education in Singapore. The research is also partially funded by Republic of Singapore NRF under grant NRF2013EWT-EIRP04-012, the National Natural Science Foundation of China (Nos. 61703105 and 61703106) the Natural Science Foundation of Fujian Province of China (No. 2017J01500), the Youth Science Foundation of Fujian Province of China (No. JZ160415), the Qishan Talent Support Program in Fuzhou University (No. XRC-1623), and the Research Foundation of Fuzhou University (No. XRC-17011).

References

[1] M. LaFrance, et al., Technology Roadmap: Energy Efficient Building Envelopes, 2013.
[2] A. Allouhi, Y. El Fouih, T. Kousksou, A. Jamil, Y. Zeraouli, Y. Mourad, Energy consumption and efficiency in buildings: current status and future trends, J. Clean. Prod. 109 (2015) 118–130.
[3] J. Yang, M. Santamouris, S.E. Lee, Review of occupancy sensing systems and occupancy modeling methodologies for the application in institutional buildings, Energy Build. 121 (2016) 344–349.
[4] L. Klein, J.-y. Kwak, G. Kavulya, F. Jazizadeh, B. Becerik-Gerber, P. Varakantham, M. Tambe, Coordinating occupant behavior for building energy and comfort management using multi-agent systems, Autom. Constr. 22 (2012) 525–536.
[5] Y. Zhou, Statistical Learning for Sparse Sensing and Agile Operation, 2017.
[6] Y. Zhou, Z. Kang, L. Zhang, C. Spanos, Causal analysis for non-stationary time series in sensor-rich smart buildings, in: 2013 IEEE International Conference on Automation Science and Engineering (CASE), IEEE, 2013, pp. 593–598.
[7] Y. Zhou, B. Jin, C.J. Spanos, Learning convex piecewise linear machine for data-driven optimal control, in: 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), IEEE, 2015, pp. 966–972.
[8] Y. Agarwal, B. Balaji, S. Dutta, R.K. Gupta, T. Weng, Duty-cycling buildings aggressively: the next frontier in HVAC control, in: 2011 10th International

Conference on Information Processing in Sensor Networks (IPSN), IEEE, 2011, pp. 246–257.

[9] V.L. Erickson, M.Á. Carreira-Perpinán, A.E. Cerpa, Occupancy modeling and prediction for building energy management, ACM Trans. Sens. Netw. (TOSN) 10 (3) (2014) 42.

[10] T. Labeodan, K. Aduda, W. Zeiler, F. Hoving, Experimental evaluation of the performance of chair sensors in an office space for occupancy detection and occupancy-driven control, Energy Build. 111 (2016) 195–206.

[11] D. Lymberopoulos, J. Liu, X. Yang, R.R. Choudhury, V. Handziski, S. Sen, A realistic evaluation and comparison of indoor location technologies: experiences and lessons learned, in: Proceedings of the 14th International Conference on Information Processing in Sensor Networks, ACM, 2015, pp. 178–189.

[12] B. Balaji, J. Xu, A. Nwokafor, R. Gupta, Y. Agarwal, Sentinel: occupancy based HVAC actuation using existing WiFi infrastructure within commercial buildings, in: Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems, ACM, 2013, p. 17.

[13] H. Zou, X. Lu, H. Jiang, L. Xie, A fast and precise indoor localization algorithm based on an online sequential extreme learning machine, Sensors 15 (1) (2015) 1804–1824.

[14] T. Labeodan, W. Zeiler, G. Boxem, Y. Zhao, Occupancy measurement in commercial office buildings for demand-driven control applications – a survey and detection system evaluation, Energy Build. 93 (2015) 303–314.

[15] H. Zou, L. Xie, Q.-S. Jia, H. Wang, An integrative weighted path loss and extreme learning machine approach to RFID based indoor positioning, in: 2013 International Conference on Indoor Positioning and Indoor Navigation (IPIN), IEEE, 2013, pp. 1–5.

[16] K. Weekly, H. Zou, L. Xie, Q.-S. Jia, A.M. Bayen, Indoor occupant positioning system using active RFID deployment and particle filters, in: 2014 IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS), IEEE, 2014, pp. 35–42.

[17] H. Zou, L. Xie, Q.-S. Jia, H. Wang, Platform and algorithm development for a RFID-based indoor positioning system, Unmanned Syst. 2 (3) (2014) 279–291.

[18] B. Huang, G. Qi, X. Yang, L. Zhao, H. Zou, Exploiting cyclic features of walking for pedestrian dead reckoning with unconstrained smartphones, in: Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp), ACM, 2016, pp. 374–385.

[19] H. Zou, Z. Chen, H. Jiang, L. Xie, C. Spanos, Accurate indoor localization and tracking using mobile phone inertial sensors, WiFi and iBeacon, in: 2017 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL), IEEE, 2017, pp. 1–4.

[20] Z. Chen, Q. Zhu, Y.C. Soh, Smartphone inertial sensor-based indoor localization and tracking with iBeacon corrections, IEEE Trans. Ind. Inform. 12 (4) (2016) 1540–1549.

[21] H. Zou, H. Jiang, Y. Luo, J. Zhu, X. Lu, L. Xie, Bluedetect: an iBeacon-enabled scheme for accurate and energy-efficient indoor–outdoor detection and seamless location-based service, Sensors 16 (2) (2016) 268.

[22] R. Jia, M. Jin, Z. Chen, C. Spanos, Soundloc: accurate room-level indoor localization using acoustic signatures, in: IEEE International Conference on Automation Science and Engineering (IEEE CASE 2015), 2015, pp. 186–193.

[23] X. Lu, H. Wen, H. Zou, H. Jiang, L. Xie, N. Trigoni, Robust occupancy inference with commodity WiFi, in: 2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), IEEE, 2016, pp. 1–8.

[24] H. Zou, H. Jiang, X. Lu, L. Xie, An online sequential extreme learning machine approach to WiFi based indoor positioning, in: 2014 IEEE World Forum on Internet of Things (WF-IoT), IEEE, 2014, pp. 111–116.

[25] M. Jin, R. Jia, C. Spanos, Virtual occupancy sensing: using smart meters to indicate your presence, IEEE Trans. Mob. Comput. 99 (2017) 1.

[26] K. Weekly, M. Jin, H. Zou, C. Hsu, A. Bayen, C. Spanos, Building-in-Briefcase (BIB), 2014 arXiv:1409.1660.

[27] M. Jin, H. Zou, K. Weekly, R. Jia, A.M. Bayen, C.J. Spanos, Environmental sensing by wearable device for indoor activity and location estimation, in: Industrial Electronics Society, IECON 2014 – 40th Annual Conference of the IEEE, IEEE, 2014, pp. 5369–5375.

[28] Z. Chen, H. Zou, H. Jiang, Q. Zhu, Y.C. Soh, L. Xie, Fusion of WiFi, smartphone sensors and landmarks using the Kalman filter for indoor localization, Sensors 15 (1) (2015) 715–732.

[29] R. Jia, M. Jin, H. Zou, Y. Yesilata, L. Xie, C. Spanos, Mapsentinel: can the knowledge of space use improve indoor tracking further? Sensors 16 (4) (2016) 472.

[30] B. Dong, B. Andrews, K.P. Lam, M. Höynck, R. Zhang, Y.-S. Chiou, D. Benitez, An information technology enabled sustainability test-bed (itest) for occupancy detection through an environmental sensing network, Energy Build. 42 (7) (2010) 1038–1046.

[31] L.M. Candanedo, V. Feldheim, Accurate occupancy detection of an office room from light, temperature, humidity and CO_2 measurements using statistical learning models, Energy Build. 112 (2016) 28–39.

[32] H. Zou, H. Wang, L. Xie, Q.-S. Jia, An RFID indoor positioning system by using weighted path loss and extreme learning machine, in: 2013 IEEE 1st International Conference on Cyber-Physical Systems, Networks, and Applications (CPSNA), IEEE, 2013, pp. 66–71.

[33] G. Conte, M. De Marchi, A.A. Nacci, V. Rana, D. Sciuto, Bluesentinel: a first approach using iBeacon for an energy efficient occupancy detection system, in: 1st ACM International Conference on Embedded Systems For Energy-Efficient Buildings (BuildSys), 2014.

[34] H. Zou, Y. Zhou, H. Jiang, B. Huang, L. Xie, C. Spanos, Adaptive localization in dynamic indoor environments by transfer kernel learning, in: Wireless Communications and Networking Conference (WCNC), 2017 IEEE, IEEE, 2017, pp. 1–6.

[35] Z. Qiu, H. Zou, H. Jiang, L. Xie, Y. Hong, Consensus-based parallel extreme learning machine for indoor localization, in: 2016 IEEE Global Communications Conference (GLOBECOM), IEEE, 2016, pp. 1–6.

[36] H. Zou, M. Jin, H. Jiang, L. Xie, C. Spanos, Winips: WiFi-based non-intrusive ips for online radio map construction, in: 2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), IEEE, 2016, pp. 1081–1082.

[37] X. Lu, Y. Long, H. Zou, Y. Chengpu, L. Xie, Robust extreme learning machine for regression problems with its application to WiFi based indoor positioning system, in: 2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP), IEEE, 2014, pp. 1–6.

[38] H. Zou, Y. Luo, X. Lu, H. Jiang, L. Xie, A mutual information based online access point selection strategy for WiFi indoor localization, in: 2015 IEEE International Conference on Automation Science and Engineering (CASE), IEEE, 2015, pp. 180–185.

[39] X. Lu, C. Yu, H. Zou, H. Jiang, L. Xie, Extreme learning machine with dead zone and its application to WiFi based indoor positioning, in: 2014 13th International Conference on Control Automation Robotics & Vision (ICARCV), IEEE, 2014, pp. 625–630.

[40] H. Zou, Y. Zhou, H. Jiang, B. Huang, L. Xie, C. Spanos, A transfer kernel learning based strategy for adaptive localization in dynamic indoor environments: poster, in: Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking (MobiCom), ACM, 2016, pp. 462–464.

[41] P. Bahl, V.N. Padmanabhan, Radar: an in-building RF-based user location and tracking system, in: INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings, vol. 2, IEEE, 2000, pp. 775–784.

[42] M. Youssef, A. Agrawala, The Horus WLAN location determination system, in: Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services, ACM, 2005, pp. 205–218.

[43] H. Zou, B. Huang, X. Lu, H. Jiang, L. Xie, A robust indoor positioning system based on the procrustes analysis and weighted extreme learning machine, IEEE Trans. Wirel. Commun. 15 (2) (2016) 1252–1266.

[44] N. Alsindi, Z. Chaloupka, N. AlKhanbashi, J. Aweya, An empirical evaluation of a probabilistic rf signature for WLAN location fingerprinting, IEEE Trans. Wirel. Commun. 13 (6) (2014) 3257–3268.

[45] H. Zou, B. Huang, X. Lu, H. Jiang, L. Xie, Standardizing location fingerprints across heterogeneous mobile devices for indoor localization, in: 2016 IEEE Wireless Communications and Networking Conference (WCNC), IEEE, 2016, pp. 1–6.

[46] G. Fierro, O. Rehmane, A. Krioukov, D. Culler, Zone-level occupancy counting with existing infrastructure, in: Proceedings of the Fourth ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings, ACM, 2012, pp. 205–206.

[47] S.K. Ghai, L.V. Thanayankizil, D.P. Seetharam, D. Chakraborty, Occupancy detection in commercial buildings using opportunistic context sources, in: 2012 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), IEEE, 2012, pp. 463–466.

[48] K. Nyarko, C. Wright-Brown, Cloud based passive building occupancy characterization for attack and disaster response, in: 2013 IEEE International Conference on Technologies for Homeland Security (HST), IEEE, 2013, pp. 748–753.

[49] OpenWrt, https://openwrt.org/.

[50] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (1) (2006) 489–501.

[51] Y. Zhou, R. Arghandeh, C.J. Spanos, Online learning of contextual hidden markov models for temporal-spatial data analysis, in: 2016 IEEE 55th Conference on Decision and Control (CDC), IEEE, 2016, pp. 6335–6341.

[52] G.H. Golub, C.F. Van Loan, Matrix Computations, vol. 3, JHU Press, 2012.

[53] P. Lazik, A. Rowe, Indoor pseudo-ranging of mobile devices using ultrasonic chirps, in: Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems, ACM, 2012, pp. 99–112.

[54] H. Zou, Y. Zhou, J. Yang, W. Gu, L. Xie, C. Spanos, Freedetector: Device-free occupancy detection with commodity WiFi, in: Sensing, Communication and Networking (SECON Workshops), 2017 IEEE International Conference on, IEEE, 2017, pp. 1–5.