4-2021

Practical server-side WiFi-based indoor localization: Addressing cardinality & outlier challenges for improved occupancy estimation

Anuradha RAVI
Singapore Management University, anuradhar@smu.edu.sg

Archan MISRA
Singapore Management University, archanm@smu.edu.sg

Practical server-side WiFi-based indoor localization: Addressing cardinality & outlier challenges for improved occupancy estimation

RAVI, Anuradha*; MISRA, Archan
School of Computing and Information Systems, Singapore Management University, Singapore
* E-mail addresses: anuradhar@smu.edu.sg (A. Ravi), archanm@smu.edu.sg (A. Misra).

Published in Ad Hoc Networks, 2021 April, 115, 102443.
DOI: 10.1016/j.adhoc.2021.102443

Abstract: Server-side WiFi-based indoor localization offers a compelling approach for passive occupancy estimation (i.e., without requiring active participation by client devices, such as smartphones carried by visitors), but is known to suffer from median error of 6–8 meters. By analyzing the characteristics of an operationally-deployed, WiFi-based passive indoor location system, based on the classical RADAR algorithm, we identify and tackle 2 practical challenges for accurate individual device localization. The first challenge is the low-cardinality issue, whereby only the associated AP generates sufficiently frequent RSSI reports, causing a client to experience large localization error due to the absence of sufficient measurements from all nearby APs. The second is the outlier resolution issue, whereby clients physically located outside the fingerprinted region but attached to the WiFi network are localized erroneously to a fingerprinted landmark. To tackle the low-cardinality challenge, we identify the stationary-period of devices, and augment the device's live AP-reported RSSI readings, during such stationary periods, with the useful but 'stale' values reported by neighboring APs. To eliminate extraneous outlier devices, we apply a threshold-based filtering strategy, where the RSSI thresholds for all interior points are derived using a combination of a weighted path-loss propagation model and the Voronoi tessellation of the fingerprinting map. In addition, to overcome intermittent false positives/negatives in localization or subsequent occupancy estimation, we apply two additional techniques: (a) temporal smoothing of location estimates over a time period, and (b) identification and removal of static devices. We experimentally evaluate these combined set of techniques on 3 different indoor work, collaboration & residential spaces, and show how these techniques improve the robustness of location tracking, which subsequently translates into an approx. 80+% reduction in the overall occupancy estimation error.

Keywords: Location based services, Occupancy estimation, WLAN network measurements

1. Introduction

WiFi-based indoor localization has been a subject of research for more than two decades, with potential commercial uses of this technology covering applications such as indoor navigation [1] and occupancy sensing [2]. Various techniques such as fingerprinting [3], propagation modeling [4] and time-of-flight estimation [5] have been analyzed extensively. While state-of-the-art techniques achieve location errors less than 10 cm, they require active client participation (often requiring an App on the mobile device) and are thus not practically usable for scenarios that require location tracking of all users/devices in an indoor space. For such applications, such as occupancy counting [6] and visitor movement analytics [7], there is a need for passive server-side localization techniques, that are able to localize all WiFi-enabled devices without any active participation or App downloads by individual users. The work in this paper focuses on addressing multiple key shortcomings in such passive WiFi localization, and is motivated by our broader, ongoing work on occupancy estimation to support energy-efficient smart buildings, via control of both HVAC and LED luminaires. Such adaptive control requires accurate occupancy estimation with finer spatial resolution (e.g., within a 5×5 m meeting room). Our key goal is to reduce the error in estimating the total device-count (a proxy for human occupancy) within selected regions of a building, via appropriate enhancements to the underlying location estimation and movement tracking process.

In this work, we consider the operationally-deployed LiveLabsserver-side WiFi localization system [8], which uses signal strength (RSSI) measurements of uplink transmissions, captured by multiple WiFi APs, to periodically compute a device's current location. This system uses the classical RADAR algorithm [9], which (a) first creates a fingerprinting database, containing a mapping between a vector of RSSI values and chosen landmark locations, during the training phase, and (b) subsequently performs a nearest-neighbor, reverse-lookup on

this database to compute device locations at runtime (during the testing phase). As detailed in [10], such fingerprint-based approaches have several limitations and are capable of providing only relatively coarse-grained location accuracy, with median errors of ~ 8–10 m [8]. Note that *client-side* approaches [11,12], which can either obtain more frequent & diverse WiFi measurements via active scanning or utilize additional inertial sensors embedded in mobile devices, usually have significantly lower localization error (~ 50 cm–1 m).

We tackle three empirically-observed limitations that are particularly relevant for accurate occupancy estimation based on passive AP-side measurements:

- *Low Cardinality:* As detailed in [13], to conserve energy, modern mobile nodes (*MN*s) generate WiFi PROBE_REQUESTs fairly infrequently (especially when stationary), reducing the number of distinct APs that provide fresh RSSI measurements of a mobile device. As a consequence, location estimates are predominantly derived using readings from just one AP, resulting in median localization error rates in excess of 6–8 m.
- *Extraneous Outliers:* The localization process implicitly assumes that a mobile device is always within the fingerprinted region of interest, and thus assigns any connected device to a location within this region. However, due to the larger range of WiFi signals, there are multiple instances of MNs attaching to the WiFi network from outside the fingerprinted region (e.g., from the parking lot or lawn outside a building)–such devices are localized incorrectly and effectively inflate the occupancy estimate.
- *Errors with Transient & Static Devices:* Occupancy estimates are typically derived over slightly longer durations (e.g., 5 min) to avoid counting transient devices. When WiFi location estimates are computed more frequently, the estimated location can fluctuate 'randomly' across consecutive readings, thereby polluting the occupancy count, especially for devices that are close to the boundary of the region being monitored. Moreover, public spaces often contain non-personal static devices (e.g., wireless kiosks or WiFi-enabled cameras), which can artificially inflate the occupancy count unless they are identified and eliminated.

Key Contributions: We shall quantify the three challenges mentioned above and then present techniques to reduce the resulting localization errors. We make the following key contributions:

- *Improve Cardinality via Judicious Use of Stale Measurements*: We tackle the problem of low cardinality by augmenting the freshly-generated RSSI reports (typically only from the associated AP) with the older RSSI values previously reported by additional, non-associated APs. Via empirical studies, we discover that such reports from non-associated APs are typically the result of explicit AP scanning, which primarily occurs only when an MN actually moves and sees a sharp drop in RSSI values (in contrast to the constant random RSSI fluctuations that are present in even static MNs). We exploit this property to implicitly identify the periods when an MN is stationary, and reuse the stale, but likely still valid, RSSI reports from other APs to improve the median localization error from 6 m to 4 m.
- *Eliminating Outlier devices*: We demonstrate a method for elimination of outlier devices, i.e., those that connect to the WiFi APs from outside the fingerprinted area. To eliminate these extraneous clients, we first define the Voronoi region, in RSSI space, for each landmark. To ensure that Voronoi cells are convex polytopes even for boundary landmarks, we use a propagation model approach to estimate the RSSI values at points on the boundary of the fingerprinted area without requiring laborious additional fingerprinting. Each Voronoi region is then used to define a (min,max) *per-AP, per-landmark* threshold, such that devices localized to a specific landmark but with RSSI readings outside these thresholds are marked as outliers. We empirically demonstrate the effectiveness of this approach in eliminating such outlier devices from the occupancy estimation process.
- *Eliminating Transient Errors & Non-Personal Devices*: Motivated by empirical observations, we develop two additional post-processing techniques to improve the accuracy of occupancy estimation. First, we perform temporal smoothing, by computing the *mode* of the multiple location estimates for a device over an occupancy estimation period T. We show that such smoothing helps to improve the accuracy of location estimation for devices close to the boundary regions by 4%. Second, we develop a mechanism to identify and eliminate the devices that are observed to be remain physically static over a long period (across days), as such devices are likely non-personal devices that potentially inflate the occupancy count.
- *Real-World Deployment*: We validate our overall solution, using real-world studies, under different occupancy levels, on three different workspaces in our university campus: (a) an access-controlled academic lab space, typically occupied by a regular pool of research employees, (b) a student-centric shared, public collaboration space with a floating pool of users, and (c) common spaces of a student residential (dorm) building. We show that our refinements to the underlying location system help us reduce the occupancy estimation error to approx. 24%–30%, compared to baseline errors that vary between ~97%–120% (in the absence of these techniques).

We believe that our work is both *generalizable* and *practically useful*: a) the challenges of low cardinality & outlier elimination apply not just to RADAR but any other passive localization technique (e.g., based on propagation models), and (b) indoor occupancy analytics is projected to be a USD $9B market by 2025.[1] The rest of the paper is organized as follows. Section 2 surveys relevant past work. Section 3 discusses the experimental setup and data collection methods across the 3 distinct spaces. Section 4 describes the problems identified in the server-side indoor localization system. Section 5 gives the proposed solutions to solve the challenges identified. We discuss the results of our methodology in Section 6. Finally, Section 7 concludes the paper.

2. Related work

In this section we present the existing approaches towards WiFi based indoor localization, as well as describe their associated challenges. The WiFi fingerprinting-based approach for indoor localization was introduced classically via techniques such as RADAR [9] & Horus [14] that utilize coarse-grained RSSI information. In more recent years, an extremely large variety of approaches (e.g., PinLoc [4], Pilot [15]) have used additional physical-layer information (e.g., the phase & amplitude of different sub-carrier frequencies) to enrich the fingerprint and improve location estimation accuracy.

Researchers have also extensively investigated the sensitivity of RSSI-based fingerprinting strategies to various parameters. Li et al. [16] studied how the average localization error was inversely proportional to the number of landmarks. Kaemarungsi et al. [17] showed that RSSI values in indoor environments are prone to fluctuation due to changes in ambient conditions and *shadowing* effects caused by the presence of humans. Mazuelas et al. [18] proposed a robust indoor positioning system by dynamically calibrating the propagation model using the obtained real-time RSSI values. Talvitie et al. [19] discussed the impact of missing fingerprints and compared techniques for interpolating from sparse fingerprint values, while Kafrawy et al. [20] developed indoor WiFi propagation models for localizing vehicles and humans. To overcome the RSSI variations due to client heterogeneity, Kjaergaard [21] proposed a hyperbolic fingerprinting method that uses ratios of signal strengths (instead of absolute values) for localization. Wang et al. [22]

[1] https://www.marketwatch.com/press-release/indoor-location-by-positioning-systems-market-size-to-surpass-447-cagr-up-to-2025-2019-04-29.

showed that temporal correlation, over a longer observation period, can improve the accuracy of RSSI fingerprinting-based indoor localization.

As an alternative to active WiFi-based localization techniques, researchers have also investigated the specific challenges with server-side, *passive* WiFi localization. For instance, Li et al. [23] exploit the absolute time-of-arrival (ToA) of WiFi signals, measured at multiple WiFi sniffers. In a different approach, Li et al. [24] utilize a nonlinear regression model to relate the CIR information from PHY-layer with the propagation distance and propose a novel trilateration algorithm to localize devices. Sun et al. [25] also demonstrated a passive approach, based on the sniffing of PROBE_REQUEST frame transmitted by mobile clients, that uses a log-distance path loss model to perform trilateration-based location estimation. While the paper reports median localization error of ~4–5 m, the experimental studies are fairly limited (involving individuals walking in straight line segments) and also involved the use of custom sniffing hardware. The LiveLabs server-side localization system [8], uses real-time measurements from a commercially deployed Aruba WiFi network, to perform RADAR-based localization of devices: while median errors in this system were ~6–8 m, the error distribution was seen to have a long tail (95th percentile error = 15 m). Such commercial infrastructure typically offers less frequent and coarser measurement reports than custom sniffers, directly impacting the accuracy achievable via a passive sensing approach. Via a careful analysis of such a server-side localization system, Jaisinghani et al. [13] established the severity of the *cardinality problem*, whereby multiple APs provide RSSI measurements only infrequently, with the problem being more acute in the 5.5 GHz band. In an alternative approach, ArrayTrack [26] derived accurate location estimates (~20 cm median error) by triangulating across AoA estimates from multiple cooperating APs.

Researchers have also suggested more sophisticated passive localization techniques, which do not even require the individual to carry a WiFi-enabled mobile device. Such *passive localization* approaches include radio tomography imaging (e.g., [27]), which constructs a 3-D 'image" of an environment based on concurrent fine-grained RSSI measurements made by multiple sophisticated receivers, or the use of ultra-wideband (UWB) technology [28], which allows passive tracking of moving objects even through obstacles such as walls. These approaches, however, all require custom infrastructure deployment.

Overall, all of these approaches implicitly assume the availability of a set of RSSI measurements from multiple APs or receivers, likely obtained through *active scans issued by an MN*–as has been noted, **state-of-the-art WiFi clients rarely perform such active scans**. These methods also focus on improving an MN's location accuracy within the fingerprinting area, instead of explicitly tackling the problem of filtering out MNs that attach to an AP but are outside the fingerprinted region.

3. Experimental setup and data collection

In this work, we utilize, as a baseline, the LiveLabs indoor localization system [8] deployed in our university campus, based on the existing Aruba-provided WiFi infrastructure [29]. We test our proposed system for occupancy tracking in three distinct spaces:

(a) An access-controlled academic research lab (roughly 400 m^2 in area) that is primarily occupied by a set of full-time research and administrative staff. Fig. 1 illustrates the floor plan of the research labs, with the ith fingerprinted landmark locations marked as L_i. The figure also illustrates the notion of extraneous or outlier devices, marked as "E" in the floor plan, which are outside the research lab area but can get included in the occupancy estimate if they get erroneously mapped to one of the designated landmarks within the research space. This space is covered by 5 distinct APs, all located within the research space.

(b) Another informal, collaborative workspace (roughly 1000 m^2 in area), in a different building, that is accessible to all university students and staff, and is typically used in a 24X7 fashion. Fig. 2 illustrates the various landmark coordinates and the distinct sections into which this collaborative space are divided; this area is covered by 4 different APs (AP01, AP02, AP03 and AP04). We restrict our study to sections S1, S2, S3, S6 and the 'lobby' area, as the sections S4 and S5 are currently seen to be largely unoccupied during daily use. The list of outlier devices will then consist of devices located in the lobby area (as this is outside the boundaries of the collaborative space), as well as other devices (e.g., on other floors) that get connected to one of the four APs mentioned above.

(c) To demonstrate that our approach is robust and independent of vendor-specific features, we also analyze data collected by a Cisco-Meraki [30] based WiFi infrastructure deployed in a residential building that houses university students. We restrict our study to the common spaces utilized by the students on a 24X7 basis. Fig. 3 illustrates the floor plan for one such common space — the extraneous devices as marked as "E", and the legitimate devices as "D", in the floor plan. For privacy reasons, we were unable to fingerprint the other non-communal spaces of the building. Moreover, due to the ongoing viral pandemic, the premises did not have its usual quota of residents, making it impossible for us to conduct long-term studies on "real-world" occupancy behavior. Accordingly, we primarily utilize this building to additionally test our outlier detection mechanisms, i.e., the ability to filter out the student devices that connect to the WiFi network from outside the designated common spaces.

3.1. Localization process

For the Aruba infrastructure (which forms the dominant part of our analysis), each AP in the infrastructure generates real-time RTLS[2] data every 5 s, which are directed to our localization servers. For the Cisco infrastructure deployed in the residential building, the localization system receives RSSI feeds from the Meraki cloud [30]. The localization process consists of two phases, as described below:

Offline (Fingerprinting) Phase: The fingerprinting process involves choosing distinct landmarks (annotated as $L\_63 - L\_74$ in Fig. 1 and $3-6$ *and* $15-34$ in Fig. 2)–our designated landmarks (3–5 m apart) are delineated based on the overhead sprinklers, which are typically deployed in a grid-like fashion with a separation of ~3 m. We then manually place devices **(configured with custom software to perform active scanning continuously)** at each of these landmarks, and record the RSSI readings. For the Aruba network, these readings are reported once very 5 s as part of RTLS packets, by the corresponding APs over a total period of 5 min. For the Cisco network, as the data received from Meraki Cloud arrives aperiodically and in random order, the system waits until it receives new reports from all APs before proceeding on to localization, resulting in location update periods of $\approx 1 - 1.5$ mins. The values received over the 5 min interval are then averaged to create the RSSI fingerprint for each landmark. To further improve the accuracy of fingerprinting maps, we conducted the fingerprint exercise on 5 different days over different crowd densities, averaging the readings received across different days. Table 1 summarizes the key data fields received in such RTLS reports, for both the Aruba and Cisco deployments. Note that the channel and age information is absent from the Cisco data. As the Cisco infrastructure was seen to generate updated readings from all the APs, independent of association status, this system does not require the use of our 'stale data augmentation' mechanism (to be discussed in Section 5.1).

[2] *Real Time Location Services*, a standard format for supporting continuous location tracking.

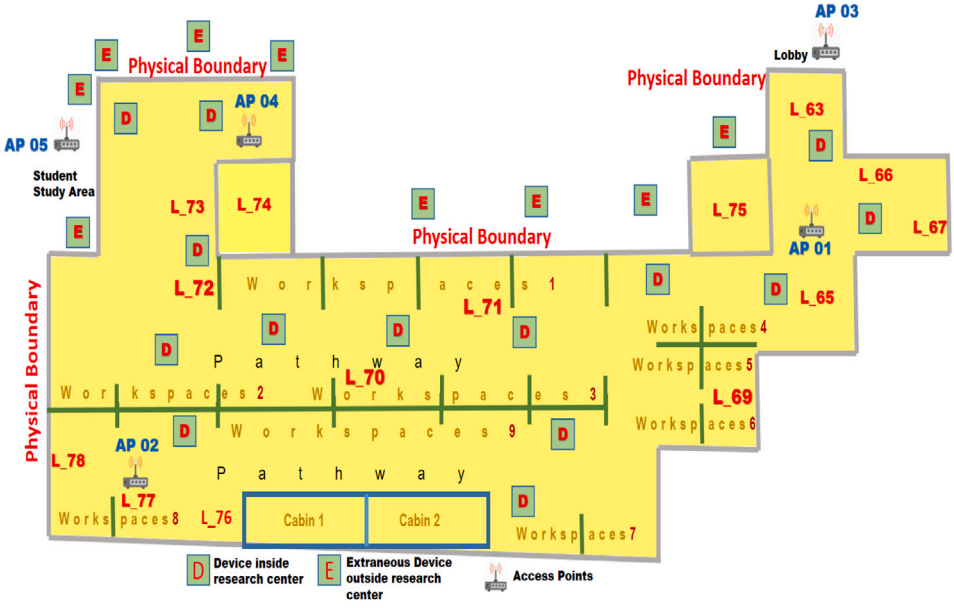


Fig. 1. Floor plan of building 1 (research center).

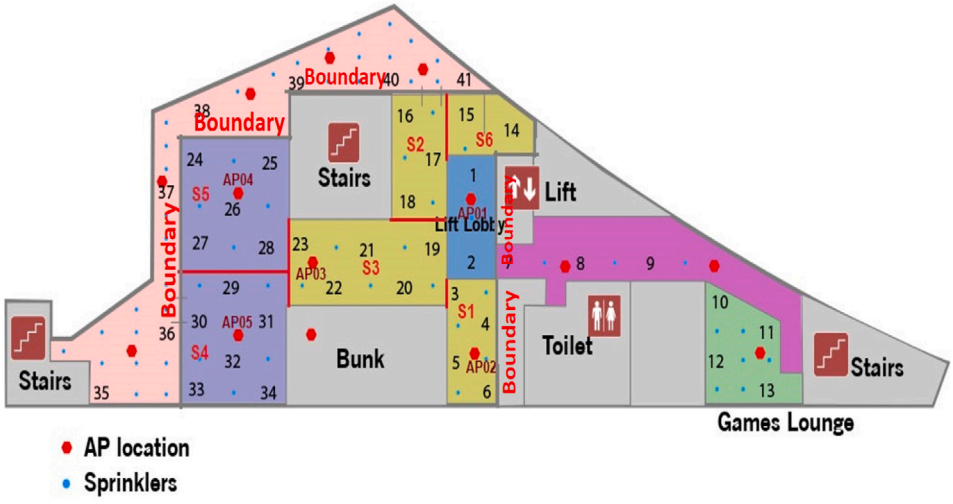


Fig. 2. Floor plan of building 2 (collaborative workspace).

Table 1
AP data as received from Aruba and CISCO-Meraki infrastructures.

Field	Description	Aruba	Cisco
Timestamp	AP Epoch time (Milliseconds)	Yes	Yes
Client MAC	SHA1 of original MAC address	Yes	Yes
Age	Elapsed seconds since AP received a device's frames	Yes	No
Channel	2.4/5 GHz band on which device was seen by AP	Yes	No
AP MAC	APs MAC Address	Yes	Yes
Associate status	Association of a device with particular AP	Yes	Yes
RSSI	Signal strength for a device as reported by AP	Yes	Yes

The fingerprint map consists of a mapping between each landmark and the vector of averaged RSSI values reported by the APs for a device placed at that particular landmark, as given in Eq. (1). Here, L_n represents the landmark number and the N-dimensional (N = total number of APs) vector $\langle AP_n, RSSI_n\rangle$ gives the list of APs along with the averaged RSSI value reported by the corresponding AP.

$$\langle L_1 : \langle AP_1, RSSI_1\rangle \cdots \langle AP_n, RSSI_n\rangle \cdots$$
$$L_n : \langle AP_1, RSSI_1\rangle \cdots \langle AP_n, RSSI_n\rangle\rangle \quad (1)$$

Online Phase: In the online phase, the APs individual report their RTLS feeds (with the fields listed in Table 1) to our localization system. After aggregating the entries for a single device (MN_k), the RSSI distance (in signal strength space) is computed for each individual MN, as indicated in Eq. (2), for each landmark. AP^k_{Rn} represents the real time RSSI value received by an AP for the client and AP^l_{Fn} represents the fingerprinted RSSI value recorded for the lth landmark. Subsequently, the landmark that is the *nearest neighbor* in the RSSI space, i.e., the landmark with the smallest value of $Dist(l,k)$, is chosen as the best estimate of the MN's location. Note that the Aruba infrastructure explicitly provides an "age" field indicating the time elapsed since the last measurement by the AP — in prior work [13], this field has been used to filter out the *stale* readings, based on the belief that such correspond to some past location of the client.

$$Dist(l,k) = \sqrt{(AP^k_{R1} - AP_{F1^l})^2 + \cdots + (AP^k_{Rn} - AP_{Fn^l})^2} \quad (2)$$

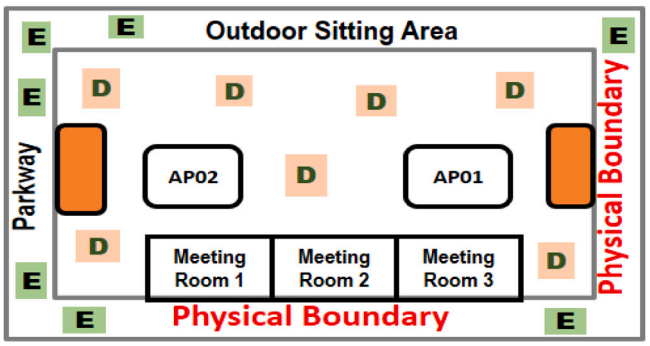


Fig. 3. Floor plan of building 3 (common space — residential building).

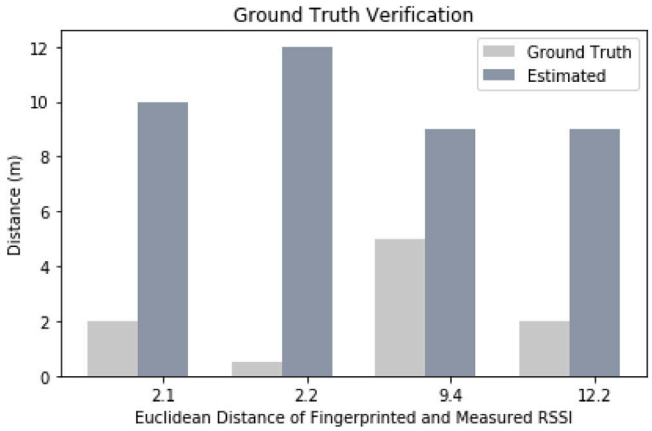


Fig. 4. True estimation error vs. Best-case error for different stationary locations (research lab).

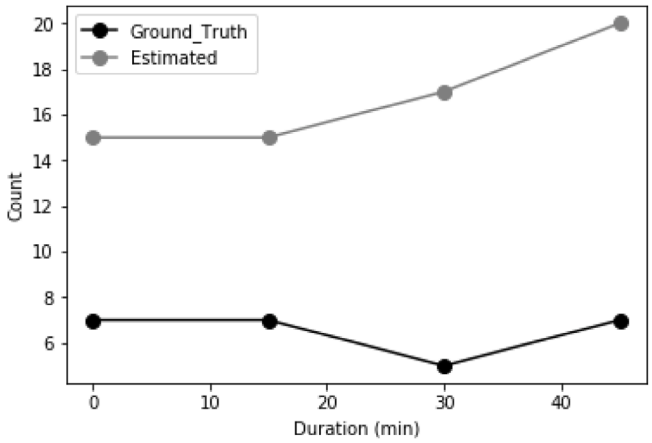


Fig. 5. Occupancy count (residential building): Expected vs. Estimated.

3.2. Groundtruth verification

The above mentioned localization process results in an average localization error of 8–10 m, which we empirically evaluated by measuring the distance error observed between the ground-truth location of the device and the location estimated by the localization system. We then used the server-side RSSI readings to first compute the estimated landmark location. For a given AP_i, let $RSSI_i(F, L)$ denote the fingerprinted RSSI value (at landmark L) and $RSSI_i(R)$ represent the measured RSSI value for the test client. Intuitively, the closer the MN's location to a specific landmark L, the smaller should be the difference between $RSSI(R)$ and $RSSI_i(F, L)$.

By varying the MN's location, we can obtain different values for this difference between the MN and its closest landmark L_{near} (i.e., the landmark to which the MN should ideally be localized): $RSSI - Diff(L_{near}) = |RSSI_i(F, L_{near}) - RSSI(R)|$. Fig. 4 reports the physical distance error reported between the ground-truth and the estimated landmark location vs $RSSI - Diff_{(near)}$, for different locations of the MN. Via multiple experiments, we observe that, contrary to our expectations, the MN's localization error is usually much higher than the minimum possible error (i.e., the distance to the nearest landmark). In effect, this occurs because the MN's RSSI distance to a farther (erroneous landmark) turns out to be smaller than $RSSI - Diff(L_{near})$. For example, when the measured RSSI error measured between the client's RSSI values and that of its nearest landmark is *2.2*, the physical distance error is expected to be $\leq 3m$; however, the observed distance error is much higher ($11.8m$). This implies that MN is incorrectly assigned to a landmark that is quite distinct and far from L_{near}.

In addition, using the baseline localization system, we also observed the total device count within each workspace. Fig. 5 plots the observed vs. ground truth occupancy count of individual devices within the Residential building space, over an observation duration of 45 min. We observe that the estimated occupancy is always considerably higher than the ground truth. We shall further analyze the reasons behind such estimation error in Section 4.

4. Detailing the identified problems

The analysis performed on the data collected from multiple locations reveal several problems, of which we focus on tackling three that are shown to have a significant impact on the accuracy and robustness of the localization system.

4.1. Low cardinality

When attempting to connect to a WiFi network, a mobile device broadcasts PROBE_REQUEST messages, typically across multiple channels. These messages are received by all APs in its vicinity. Typically, the AP with the highest signal strength is chosen for association, with all data frames subsequently unicast to this associated AP. Client devices transmit PROBE_REQUESTS either in the 2.4 GHz or the 5 GHz band, but based on the configuration set by the network administrator, the APs either reply with the probe response in the same band as requested by the device, or in the preferred 5 GHz band. (In modern networks, the 5.5 GHz is selected preferentially, due to the larger number of orthogonal channels which helps minimize cross-device interference.) For devices which do not support the 5 GHz band, the APs send probe response in 2.4 GHz space [30].

From our experimental data, we observed that the "associated" AP provides updated RSSI measurements in each consecutive RTLS update (i.e., every 5 s for the Aruba network). In contrast, the other (non-associated) APs report fresh RSSI values only during the initial probing phase, when the MN performs active scanning across all channels, but subsequently does not generate any fresh RSSI reports [13] while the MN is engaged in data communication with its associated AP. This observation is corroborated by the Aruba specifications [29], which state that the APs associated with a client device report the RSSI values for clients based on the data frames, while the ones that are not associated report the RSSI values purely based on measurements of PROBE_REQUEST frames. Prior work [31] has also observed that mobile devices tend to send more PROBE_REQUESTs in the 2.4 GHz band (compared to 5.5 GHz), and that the transmission frequency of PROBE_REQUESTS tend to decrease (to conserve energy consumption) when an MN is in the vicinity of "known" SSID networks.

Due to the above-mentioned reasons, we tend to receive regularly-refreshed updated RSSI values only from the associated AP, while the other APs simply report the stale readings (observed during the

Table 2
Data reported for client-ABC in the research center.

Timestamp	Age	AP MAC	RSSI	Associated status	Client MAC
533	33	AP01	−74	0	ABC
535	2	AP02	−60	BSSID of Associated AP	ABC
536	33	AP03	−65	0	ABC
536	34	AP05	−61	0	ABC

Table 3
Data reported for client-ABC in the residential building.

Access points	SNR (Client-Reported)	SNR (Fingerprinted)
AP01	12	30.4
AP02	24	10.6
AP03	9	37.6

infrequent probe phase). We report one such example in Table 2, which shows the RSSI values reported by all the APs for a particular client. The age parameter is low only for the associated AP (AP02), while the other APs ($status = 0$) report the same stale RSSI values, but with increasing age. State-of-the-art localization systems simply filter out and discard such stale RSSI values; such elimination effectively causes the localization process to utilize readings from only one AP, which results in an increased estimation error.

4.2. Outlier location

Most APs have a fairly long range — in our university, we can hear APs that are 60 m away. Accordingly in Fig. 1, the range of AP01 and AP03 can extend well beyond the boundaries of the research lab, even percolating to the next building. More specifically, the range is even higher for the 2.4 GHz band, due to its lower attenuation properties. For devices located within the fingerprinted building, this extended range provides a benefit — it effectively increases the cardinality of RSSI reports, thereby reducing the localization error. However, this increased range also comes with a drawback for accurate occupancy estimation: devices can attach to the WiFi network from locations outside the fingerprinted region (e.g., from outside the building) and can then erroneously be localized within the fingerprinted area. Such errors arise because current localization techniques do not incorporate any explicit outlier elimination logic, and thus can cause such clients to be also included in occupancy estimates.

As an illustration of this phenomenon, Table 3 presents an example of a device, reported by 3 different APs. The table plots both the true SNR (higher the SNR, stronger the signal) value reported for the MN (column *Client-Reported*), as well as the fingerprinted SNR values corresponding to the landmark L (column $Fingerprinted$) to which the MN is currently localized. As observed, the RSSI values reported by AP01 and AP03 are far less compared to the fingerprinted values for the landmark, while AP02's reported RSSI value is very high. On visually inspecting these values, we can conclude that this is likely an extraneous device, located outside the residential building's fingerprinted areas. However, as this outside area is not fingerprinted, the localization algorithm has no mechanism to map this device to such external landmarks, and will end up localizing the device to *one of the landmarks within the residential space*.[3] In turn, this will inflate the occupancy count for devices. In Section 5.2.1, we shall show how we can tackle this problem by effectively defining a permissible range within which RSSI readings must lie to be accepted as legitimate clients.

[3] One can conceivably argue that, in this specific case, we can address the problem by fingerprinting the space outside the residential building. However, this is not a tenable approach — for example, even devices on the streets and parks outside can connect to this WiFi network, and it is not practical to attempt to fingerprint all such areas.

Table 4
Time-varying estimated location (Landmark) vs. RSSI values for a stationary client (Research Lab).

Client RSSI	Estimated landmark	Closest landmark (Ground-Truth)
AP01 : −50	L_64	L_64
AP01 : −52	L_64	L_64
AP01 : −56	L_65	L_64
AP01 : −64	(Outlier)	L_64

Table 5
Fingerprint similarity across multiple (3 different) landmarks.

Landmark	Fingerprint values
4	AP01: −57; AP02: −45
5	AP01: −58; AP02: −41
6	AP01: −60; AP02: −43

4.3. Localization errors

The large errors observed in Fig. 4 are due to not just the problem of low cardinality, but also due to the significant temporal fluctuation in the reported RSSI values, even for stationary clients. As an illustrative example, Table 4 enumerates a selected set of RSSI values, and the resulting estimated landmark location, for a stationary client, compared to its closest actual landmark, over a 5-min observation period. (Each entry is based on the average of RSSI values computed over a 20 s interval.) We observe that the estimated landmark itself fluctuates, between not just two legitimate landmarks but also, occasionally, to lie outside the fingerprinted area itself. Such fluctuations are due to effects such as shadow fading and transient movement of passersby, and can cause false positives/negatives in the location estimation process.

In addition, at certain locations, distinct landmarks may have fingerprint values that are very close to one another in the RSSI space, implying that even minor fluctuations in the RSSI readings can cause the estimated location to fluctuate across such landmarks. Table 5 demonstrates one such example with three different landmarks (4, 5 and 6 as observed in Fig. 2) that are mutually at least 3 m apart in the Collaborative space. We can see that the RSSI tuples for all three landmarks are very similar. In Section 5.3, we shall see how the incorporation of a temporal smoothing approach helps to tackle these time-varying localization errors.

5. Proposed solution

In this section, we present our proposed solutions for the problems defined in the previous section.

5.1. Low cardinality

Prior studies [32,33] have shown that the decline in PROBE_REQUEST, and the consequent drop in AP cardinality, occurs for stationary clients; in contrast, when a client moves (away from the range of an AP), it effectively initiates a probing phase, which helps the APs in the MN's vicinity to obtain revised RSSI estimates. Accordingly, we hypothesize that the reporting of the MN by a new AP is a marker of significant movement by an MN. Our proposed solution exploits this phenomenon — we effectively first classify if a device is in the *moving/probe vs. stationary* stage, by observing whether there are new APs (other than the currently associated AP) that have generated new RSSI values. If so, this is likely due to the explicit PROBE_REQUEST scans initiated by a moving MN; in this case, we use only the *recent* AP reports, discarding all *stale* reports (those with timestamps older than 15 secs). However, if there are no new reporting APs, then we conclude that the MN is still *stationary* (or has moved only by a small distance) and then include even the older (stale) RSSI readings, as those readings

Algorithm 1 Stationary Pattern Identification

```
for R : ⟨RTLS⟩ do
  if R.AP ≠ ArchivalTable.AP then
    Add RTLS_AP to ⟨AP_List⟩ for Location Estimation
    NewFound=1 //Here, New Found indicates a new AP association → client is moving.
  end if
end for
for R : RTLS do
  if ArchivalTable_AP = R.AP then
    if R.Age ≤ Archival_Table_AP.age then
      Add RTLS AP to ⟨AP_List⟩ for Location Estimation
    else if RTLSAge ≥ ArchivalTableAge ς NewFound ≠ 1 then
      Add R record to ⟨AP_List⟩ for Location Estimation //Include stale AP data.
    else if R.Age ≥ ArchivalTableAge ς NewFound = 1 then
      Do Not Add R to ⟨AP_List⟩ for Location Estimation //new AP found → ignore stale AP data.
    end if
  end if
end for
```

are likely to be persistent for a stationary device. Algorithm 1 outlines the relevant pseudocode.

To justify this insight into MN behavior, we present empirical result on the observed active scanning behavior and RSSI fluctuation as an MN is progressively moved farther away from its currently-associated AP. Fig. 6(a) represents the observed changes in RSSI reports generated by one or more APs, as an MN is gradually moved farther (over a range of 1.5–22 m) from its associated AP (AP01), with the MN staying stationary at each location for ~10 mins. We observed that the MN performs active scans (which in turn generate new RSSI reports from a nearby, but non-associated, AP02) when it moves to distances 9.6 and 16.4 m way. Even at these distances, the MN continues to maintain its existing association with AP01, although its RSSI values have progressively dropped to ~−70 dB. The figure confirms that new RTLS reports from non-associated APs occur very infrequently, and are generated only when an AP moves sufficiently away from its currently-associated AP, leading to low RSSI values and active scans.

Fig. 6(a) also shows that, as anticipated, the RSSI value continues to fluctuate even at a single location. To provide some basic stability to our localization approach, we compute the mean of each AP's MN-specific RSSI value over a period of 20 secs — i.e., roughly using 4 consecutive RTLS reports. Fig. 6(b) provides a box-plot of such 20-sec averaged RSSI values at different distances from AP01. We can see that this short-term averaging provides significant stability to the RSSI estimates, which now exhibit the desired properties of *monotonic decrease* and *non-overlapping readings* as the distance from AP01 increases.

5.2. Outlier elimination

To tackle the outlier problem, our approach is to eventually define an *acceptable RSSI range*, on a per-landmark basis, for each AP associated with that landmark. Once such a range is defined, we can then eliminate outlier MNs by first determining their predicted location (landmark), using the conventional RSSI-nearest neighbor (**NN**) approach, and then checking if the actual RSSI value lies within this landmark's acceptable range.

To create a boundary, we first divide each of the landmarks using an $N-dimensional$ (N = number of APs) Voronoi tessellation in the signal-space. Fig. 7 shows such an example (Tessellation is shown for two different sections (Sections 1 and 2) of Fig. 2 in SNR) for $N = 2$ – i.e., with two APs. (In practice, for each landmark, we restrict ourselves to a 2-dimensional tessellation, involving the two (*dominant*) APs with the strongest signal strength in the fingerprint DB. This was seen to provide sufficient practical discrimination and is computationally simple.) The tessellation represents the variation observed by the dominant APs, across different sections of the same collaboration space. We also assume that we know the distance of representative *boundary points* of the fingerprinted region from each of the APs. Outlier elimination then consists of the following steps:

5.2.1. Boundary estimation

The first step involves estimating the likely RSSI values at each of those boundary points. (Note that those points may not have been manually fingerprinted.) To estimate this, we utilize a path-loss propagation model as given in Eq. (3)

$$P_{RSSI} = \beta - n.d + X_{\alpha}, \tag{3}$$

where P_{RSSI} is the RSSI strength, β is the transmitted power and antenna gains, n gives the path loss constant, d defines the distance and X_{α} is the shadow fading defined by the Gaussian random variable with zero mean. We first apply a regressor to the known AP-landmark distances and RSSI(landmark) readings to learn the optimal model parameters. Fig. 8 gives the straight line fit for the model for a given (landmark, AP) combination. The shadow fading coefficient is estimated by the average prediction error, according to $X_{\alpha} = \frac{\sum_{i=1}^{m}(P_{e_i} - P_{o_i})^2}{m}$, where P_{e_i} denotes the predicted value, P_{o_i} the observed value and m is the number of observed values.

Subsequently, we use the regressor to predict (without any additional fingerprinting) the RSSI readings at the representative boundary points, and then use the *average* of these values to denote the global minimum **per-AP** signal strength (α^{G}_{min}) that an MN located *anywhere* within the fingerprinting region should have. Similarly, we assume a minimum distance d_{min} (computed as the height of the floor, such that any legitimate point should be $> d_{min}$ away from the AP) and compute the global per-AP maximum permissible RSSI value α^{G}_{max} (using the regressor). This step, of using a propagation model to estimate the RSSI values at boundary points, is important for ensuring that the Voronoi cells lying on the boundary are *bounded* (i.e., form a convex polytope).

5.2.2. Legitimate range estimation

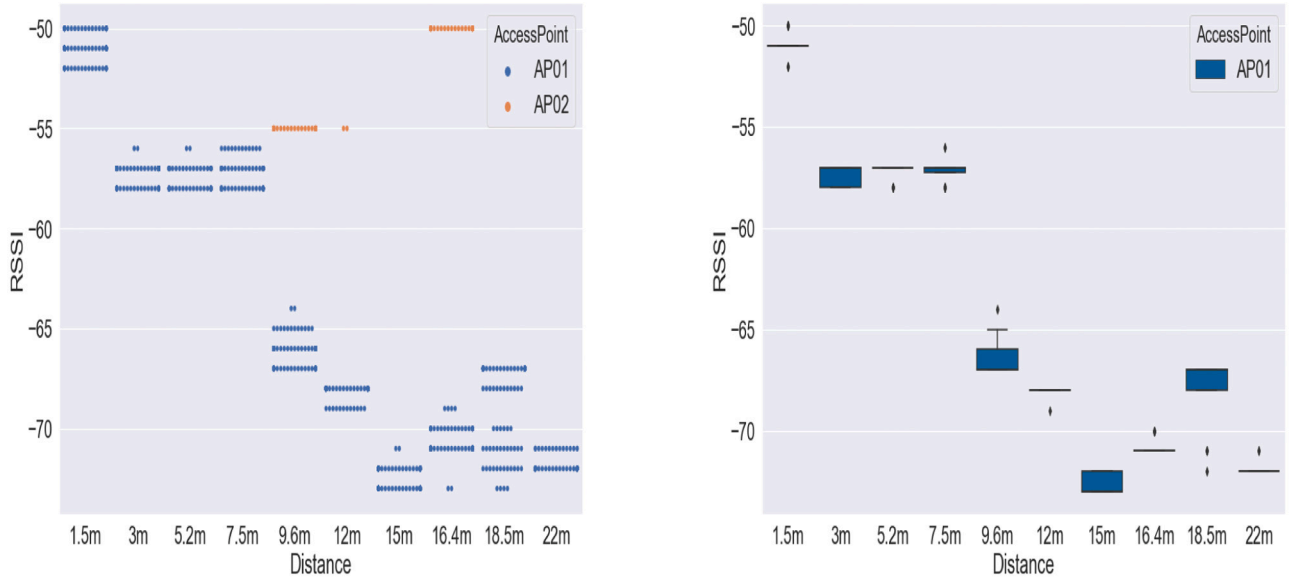
We next address the question: how do we define the real Voronoi region (in the signal space) for each landmark? To tackle this question, we need to define a range of legitimate RSSI values associated with each **(landmark, AP)** combination. In particular, the legitimate RSSI values, for AP_i, for an MN that has been mapped to landmark l are those that both (a) lie within the global range $(\alpha^{G}_{min}, \alpha^{G}_{max})$, *AND* (b) lie within the Voronoi region of landmark l. We compute the set of points (in the RSSI space) that satisfy both these criteria and accordingly define an additional, per-landmark set of thresholds for each AP: $\{\alpha^{l}_{min}, \alpha^{l}_{max}\}$. At the implementation level, we employ the Bowyer–Watson algorithm [34], which uses the per-AP fingerprinted RSSI values to create the corresponding Delaunay triangles (dual of the Voronoi cells). Fig. 9 depicts the range estimation of each landmark.

5.2.3. Final outlier logic

Given the resulting Voronoi regions and a predicted landmark l for a test client, we determine its location estimate to be *legitimate* only if (i) if its RSSI readings satisfy the global thresholds $(\alpha^{G}_{min}, \alpha^{G}_{max})$ for every reporting AP, and (ii) if the RSSI readings are within the permitted Voronoi space of the landmark's two strongest APs (denoted by $AP1(l), AP2(l)$)–i.e., it satisfies the constraints given in Eq. (4):

$$\alpha^{l}_{min}(AP1(l)) \le P_{\mathrm{RSSI}_{AP1}} \le \alpha^{l}_{max}(\mathrm{AP1(l)}) \;\&\; \alpha^{l}_{min}(AP2(l)) \le P_{\mathrm{RSSI}_{AP2}} \le \alpha^{l}_{max}(\mathrm{AP2(l)}). \tag{4}$$

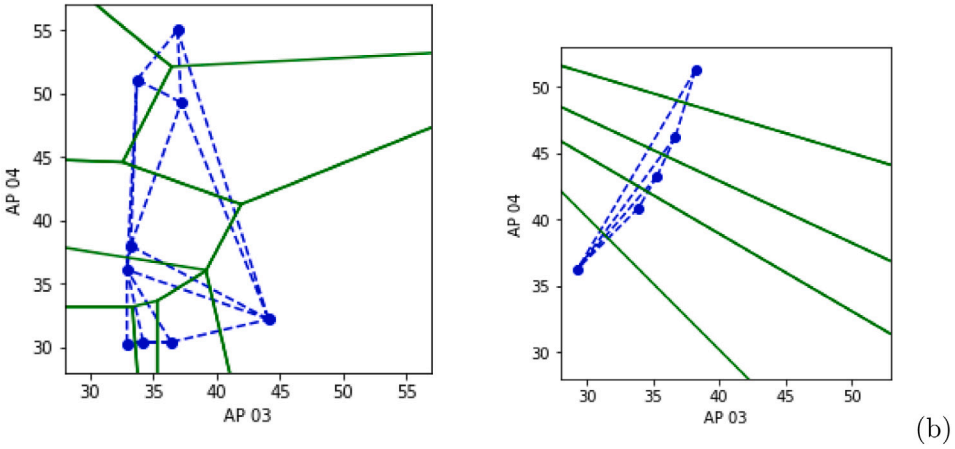
Note that if a location is declared to be illegitimate, the client is then mapped to the subsequent (second-most likely) location; this process continues iteratively until a suitable and legitimate candidate location is found or until all landmarks are exhausted (in which case the MN's location is declared as *indeterminate*).



(a) Fluctuations in RSSI vs. Distance from AP01 (associated AP)

(b) 20sec Avg. RSSI vs. Distance from AP01 (Associated AP)

Fig. 6. RSSI & Scanning behavior vs. AP-MN distance.



(a) Section 1 of collaboration Space

Section 2 of collaboration Space (b)

Fig. 7. Voronoi tessellation — collaborative workspace.

5.3. Location estimation: Temporal smoothing

We next describe our temporal-smoothing process, which helps tackle the intermittent or transient localization errors that occur due to 'random' fluctuations in instantaneous RSSI values. We first capture the entire set of estimated landmarks computed over the observation period $T = 2.5$ mins — i.e., the set of $\frac{150}{5} = 30$ landmarks computed once every 5 s. We then take the *MODE* of all such landmarks to yield one representative location, MN_{Loc}, for that MN over the specific period T. We shall see that this smoothing process is quite simple, but highly effective. Algorithm 2 details the temporal smoothing procedure.

5.4. Eliminating static devices

As a final step for improving the occupancy count, we introduce the mechanism for static device elimination — i.e., filtering out devices that are constantly present in the environment being monitored

Algorithm 2 Temporal Smoothing

{60 records retrieved for 2.5 mins interval for each client.}
for $MN : \langle Client\_List \rangle$ **do**
  for $T_i : T_{1..60}$ **do**
    if $Location\_Output_i = FOUND$ **then**
      $LOC_{1..N} = Location\_Output_i$ {Retrieve the set of landmarks estimated for the client}
    end if
  end for
  $MN_{Loc} = MODE(Location\_Output_i)$ {MN_{Loc} is the selected landmark location among all the locations observed for the client in 2.5 min.}
end for

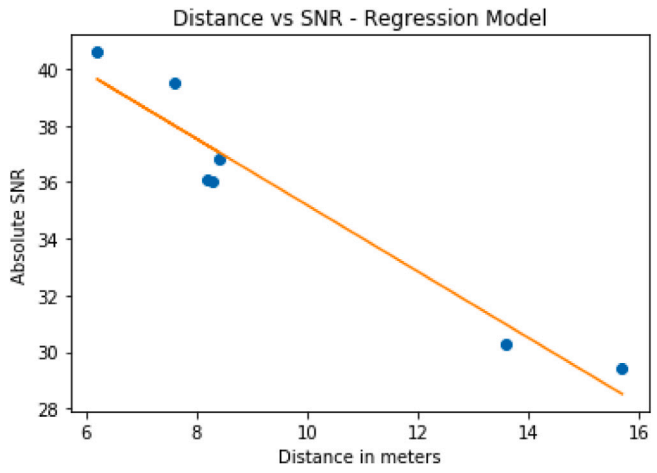


Fig. 8. SNR vs. distance relation for residential building.

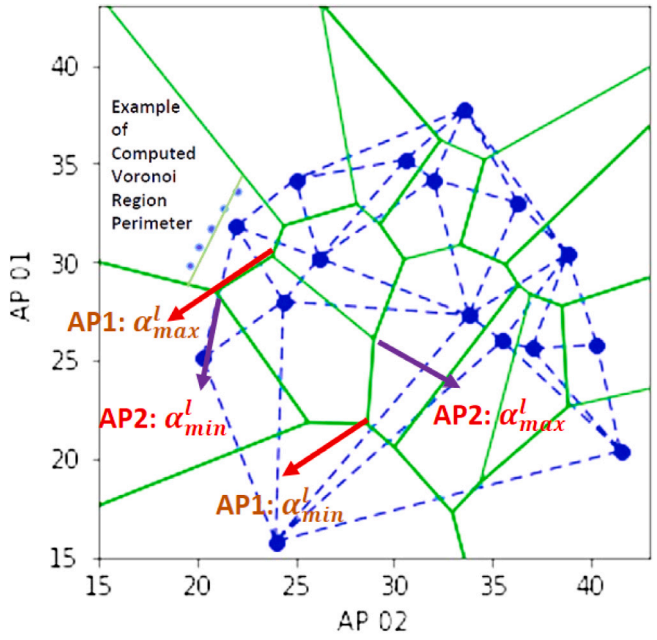


Fig. 9. Voronoi tessellation — residential building.

but that exhibit no movement all. This is based on the empirical observation that such spaces often have a set of WiFi-connected, but non-personal devices, representing objects as diverse as Smart TVs, network-connected cameras and network-connected check-in kiosks. In the absence of effective filtering, such devices can contribute to device count and inflate the total occupancy estimate.

Accordingly, we propose a novel heuristic for identifying and filtering such static devices. The high-level idea is quite straightforward: simply identify devices that are *constantly & continuously* observed at one single location (landmark). However, this apparently-simple problem turned out to be a bit more involved due to several empirically observed artifacts, which make it difficult to distinguish between static and transient devices: (a) some devices are active only occasionally during a day (e.g., activating their WiFi interface only once every 2 h), and also only intermittently across days (e.g., only 3 days per week); (b) even if the device is static, its location estimate fluctuates constantly, due to changes in the wireless propagation environment. Empirical observation showed that, while instantaneous device locations fluctuated, the set of $top-K$ estimated landmarks for such a static device remained unchanged (across all days for which the device was observed) 98% of the time. Moreover, we observed that performing this analysis using an observation period of 1 week provided robust identification — this longer period minimized the likelihood of false positives in the identification of static devices, while also reliably capturing intermittently transmitting devices (we observed that certain static devices generated WiFi signals only 42.8% (3 days) of the week). The formal definition of the filtering logic is detailed in Algorithm 3 and consists of the following key steps:

- We observe the change in location pattern of devices across an entire week; only devices that appear in the location logs on $\geq N$ days in a week are potential static device candidates. (This approach filters out a larger set of transient devices.)
- On a given single day, we observe the estimated set of locations Y (Y is a set of top-k ($k = 3$) locations) for a single client across X slots, with each slot comprising of T hours ($X * T = 24$). In practice, we set $X = 6$–i.e., consider 6 different 4-h slots per day. We then determine (Eq. (5)), for each such slot, a single representative location by computing *MODE* of all the estimated locations (set Y), for each such time period T. Our filtering logic checks for the change in Y across all the slots $X_1 ... X_6$. If there is no change observed (i.e., if the modal location of the device is observed to remain constant across multiple time periods in a day), the client is determined as static for that particular day.

$$
\begin{aligned}
Y_{T1} &= Location_1 \dots Location_{N_k} \\
Y_{T2} &= Location_1 \dots Location_{N_m} \\
&\dots \\
Y_{TN} &= Location_1 \dots Location_{N_p} \\
Y_{X_i} &= MODE(Y_{T1}, Y_{T2}, \dots Y_{TN})
\end{aligned} \tag{5}
$$

- If a device satisfies the above criteria, we additionally consider possible changes in its estimated location across the entire week. Our filtering logic observes the set Y (landmark locations for the client estimated on a single day) for each day (represented as : Y_{Day_1} for Day_1 and Y_{Day_N} for Day_N) the client was seen in the WiFi records. The set of estimated locations for each day (given by $Y_{Day_1} - > Y_{X_{1..6Day_1}}$) is the locations observed across all the time slots in a given day. From empirical observations, we propose that the set of modal locations ($Y_{X_{1..6}}$) for a device should remain constant across multiple days of the week OR at least be a subset of the location observed across all the days of the week, i.e., $Y_{Day_i} \subseteq Y_{Day_j} \subseteq Y_{Day_N}$. Here, Y_{Day_i} represents the day on which maximum landmark locations were estimated for the client.

If all of the above criteria are met, that device is identified and marked as a static device. Note that the algorithm can be parametrized in terms of a few variables (k, X, etc.) and can thus be customized to possible differences in behavioral artifacts, across different locations, by choosing different values for these parameters.

6. Results and discussion

In this section, we quantify the real-world performance of our improved AP cardinality, outlier detection and temporal smoothing methods, and show how they help improve the accuracy of occupancy estimation. To demonstrate the robustness of our techniques, we present results across the 3 different environments/spaces mentioned before. The experiments were conducted at different times of the day, with varying crowd levels and also using a variety of heterogeneous mobile devices.

Algorithm 3 Static Device Filtering Logic

for Each Day (j) in the Week **do**
{For each slot in the day, find distinct clients and retrieve their estimated locations. In our case, XSlot is 6 (Each slot is of 4 h, covering 24-h in a day)}
for Each slot in XSlot **do**
for All Distinct Clients in slot **do**
$\langle Client_{slot} : Y_{L1...LN\,Count}\rangle = Top_k\,Estimated\,Location$ {Store Top-K estimated location for a client along with their count for each slot}
end for
end for
for All distinct clients C in $\langle Client_{slot}\rangle$ **do**
{If client present in All slots, pick the highest count for each slot and compare with all other slots}
if C in $slot_{1...6}$ **then**
$C_{Y_{L1...LN}} : MAX\_COUNT(Y_{L1...LN\,1})$
for i in $slot_{1...6}$ **do**
{If the landmark list is same for all slots, consider it as static device for the day. j represents the day number}
if $C_{Y_{L1...LN\,1}} = MAX\_COUNT(Y_{L1...LN\,i})$ **then**
$\langle Static_{Day-j} : C_{Y_{L1...LN}}\rangle$
{ $< Static_{Day-j}$ represents the set of static devices for Day-j}
end if
end for
end if
end for{For all static devices in the day}
for All $\langle Static_{Day-j} : C_{Y_{L1...LN}}\rangle$ **do**
if $C_{Y_{L1...LN}}$ is same for all j **then**
Admit C as Static Device.
end if
end for
end for
{The above logic applies for both bands 5 GHz and 2.4 GHz. The same representation has been simplified for presentation.}

6.1. Improvement in cardinality

We discuss our observations on cardinality improvement primarily across the two different buildings (i.e., the research lab and collaboration workspace), where we performed the bulk of our studies. We compare our modified algorithm (which includes so-called *stale* readings during the stationary periods of an MN) against the baseline approach, which ignores such stale readings.

Fig. 10 plots the location estimation error (the distance of the predicted location from the MN's ground truth) for 4 different clients (C1–C4), placed at different locations in the university research lab. We see that the appropriately curated inclusion of stale RSSI data helps to reduce the estimation error significantly, to less than 4 m in at least 75% of these cases. The smaller improvement for C2 was due to the observed movement of multiple visitors through the area during the study, which affected the underlying radio environment.

We also present the results on the location estimation error observed in the collaboration workspace. We test the robustness of our approach on heterogeneous mobile devices to observe their behavior while they are stationary, and also as they are moved between the lift lobby, section-S1, section-S2 and section-S6 (see Fig. 2). A device was placed within each section close to a measured landmark for a duration of 20–25 mins to observe their stationary behavior and then moved to the next section. It is to be noted that all these sections are covered by 3 different APs (AP01, AP02 and AP03), with AP01 and AP02 being the dominant APs. This gave us an opportunity to understand the behavior of devices when there is movement across an area covered by multiple APs. In contrast to the university research lab setting, where the two APs that covered the lab were separated by a greater distance (approx 22 m apart), the APs (AP01 and AP02) in the collaboration workspace area were placed closer together (approx 10 m apart).

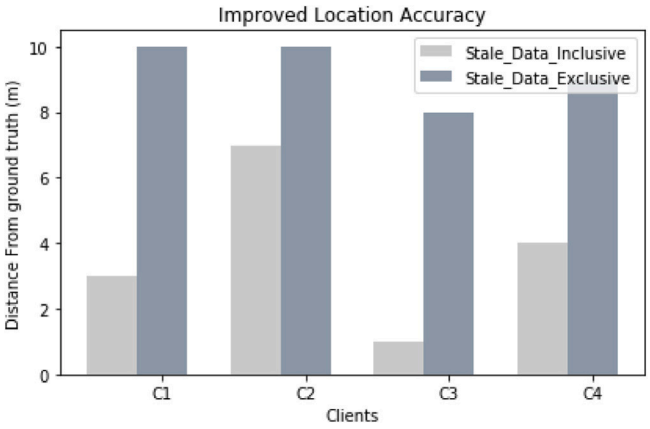


Fig. 10. Location accuracy (with & without 'Stale' readings (research space)).

Table 6
Localization Error vs. AP Cardinality.

Time	Location error (Without Stale)	Location error (With Stale)
T1 (RSSI: −58)	12 m (Cardinality-1)	4 m (Cardinality-3)
T2 (RSSI: −55)	8 m (Cardinality-1)	2.5 m (Cardinality-3)
T3 (RSSI: −51)	3 m (Cardinality-1)	3 m (Cardinality-3)
T4 (RSSI: −53)	6 m (Cardinality-2)	6 m (Cardinality-3)

Fig. 11 plots the localization error (aggregated across different sections) for 5 different devices. Clients C1–C4 were moved between each section, while the client C5 was kept stationary and was not moved between the sections. We observe that the average localization error using the baseline method (which filters stale readings) was 4–8 m, with our augmented approach reducing this error to 2–5 m.

For clients C1, C2 and C4, the average error with and without the stale readings is roughly the same (even though the std. deviation for the error is higher for C1). This was due to the higher cardinality value of AP reports (average cardinality was $\geq$2), with at least two APs reporting fresh readings for these clients 70% of the time. For client C5, which remained stationary throughout the 1.5 h time period, this AP cardinality is much lower; consequently, the use of stale readings results in a significant reduction in the estimation error. To further illustrate this point, Table 6 lists the localization error vs. different cardinality readings observed for the client C5. We see that, in addition to AP cardinality, the strength of the RSSI readings also influence the localization error. At time instants T3 and T4, the client is accurately localized to the expected landmark (3 m/6 m apart) even with cardinality=1, due to the stronger received signals.

Dominance & Accuracy of Stay Episodes Our proposed method is especially effective in tackling the cardinality problem during periods when the MN is stationary. To quantify the importance of improved localization during such stationary episodes, we analyzed the motion traces of all clients, in the Research Lab, over an entire day. We found that clients spent, on average, **92%** of the day in a "stationary state", with a mean stay duration of 372 min (and std. deviation. of 180 min). Moreover, to evaluate the possibility of our algorithm making false "stationary" inference, we experimented with multiple client devices that were (a) either completely stationary, or (b) made small movements (within 1–2 landmarks), over a 60 min duration. We noted that the clients were classified as "stationary" (for the purposes of Algorithm 1) in 100% of all such cases: while small movements resulted in changes in the RSSI value reported by the associated AP, they do not actually cause MNs to generate explicit PROBE_REQUESTs.

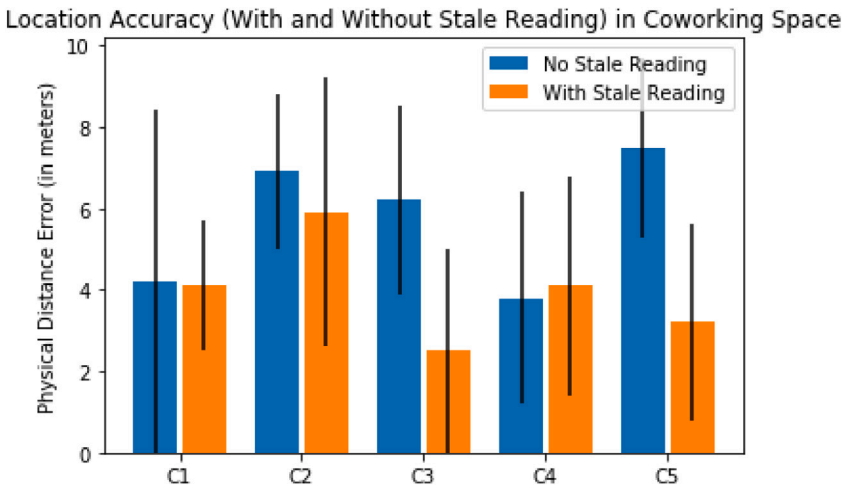


Fig. 11. Localization error (with & without 'Stale' AP readings (collaborative space)).

Table 7
Localization Accuracy vs. Cardinality.

Cardinality	1	2	3
Accuracy	15%	85%	92%

Table 8
Data reported for test client in the residential dorm.

Access points	Client SNR	Without threshold	With threshold
AP01	11	Included	$\alpha^{G}_{min}(1) = 13.5$; Excluded
AP02	17	Included	$\alpha^{G}_{min}(2) = 20.3$; Excluded
AP03	15	Included	$\alpha^{G}_{min}(3) = 19.5$; Excluded

To further demonstrate the benefit of improved cardinality, Table 7 lists the location estimation accuracy vs. the cardinality of the corresponding RSSI reports (for the Research Lab). A location estimate is deemed to be accurate if the estimated location (landmark) is identical to the ground truth (the landmark nearest the MN's actual location.) Clearly, the augmentation of cardinality provides dramatic benefits, increasing the localization accuracy to ~80%–90% (in contrast to accuracy values less than 20% when cardinality=1).

6.2. Location outlier elimination

We next study the performance of our outlier elimination logic across the 3 different spaces.

As explained earlier, the outlier algorithm discards those location estimates that lie outside a permitted signal strength range defined for each (AP, landmark) combination. As an illustration of the effectiveness of this strategy, Table 8 lists the SNR values for a particular client that was reported at a landmark in the residential dorm, even though it was *actually placed at a point outside the fingerprinted region*. The table lists the SNR value for the different APs. As observed, the client readings are eliminated in all three cases, as the SNR values are seen to be outside the AP-specific *global* thresholds.

Robustness of Outlier Detection: To test the ability of our technique to accurately separate outliers from legitimate estimates, we conducted studies where an MN was placed at multiple distinct locations, under *varying crowdedness levels*, all of which are close to the boundary, but (a) *within* the fingerprinted region (i.e., within the research lab and sections {S1, S2, S6} of the collaborative space), and (b) *outside* the fingerprinted region (i.e., in the {student lounge, lobby} areas of the research lab and Lift Lobby for the collaborative space). Table 9 lists the outlier detection results, for under both conditions for the research lab, and shows that our outlier detection is robust (overall accuracy = 82%), with both low false-positive and false-negative rates.

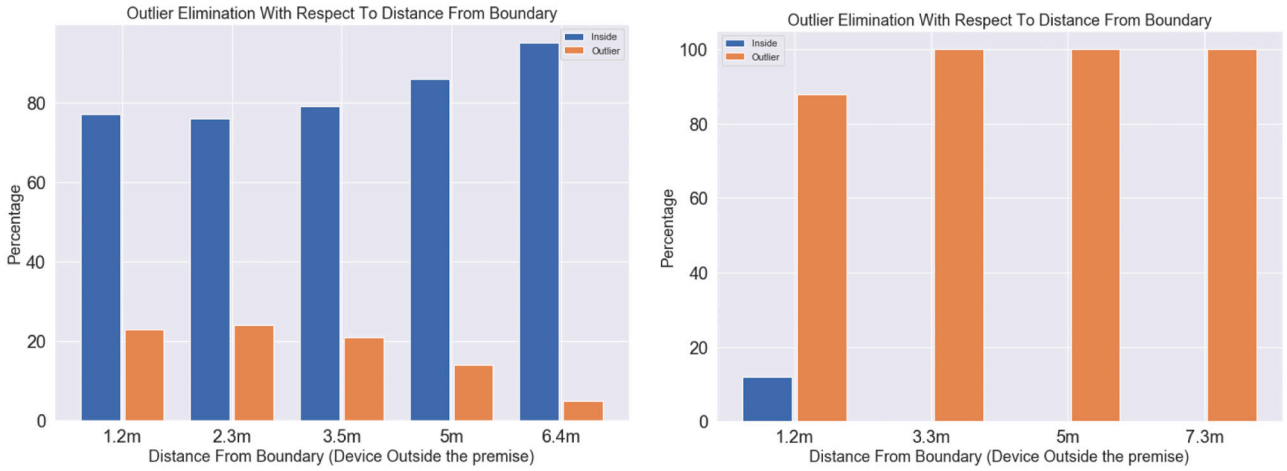
Table 9
Outlier detector performance — Research lab.

Clients	Ground truth	Inferred-Legitimate	Inferred-Outlier
C1	Inside boundary	**82**%	18%
C2	Inside boundary	**75**%	25%
C1	Outside boundary	15%	**85**%
C2	Outside boundary	14%	**86**%

In Fig. 12(a) and (b), we study the accuracy of outlier detection/classification as a function of the MN's true distance from the region boundary. As expected, the classification error is higher as the device moves closer to the edge of the fingerprinted region. Moreover, we see that the classification errors is higher (~20%, at distance = 1.2 m) when the MN is located within the fingerprinted region, as opposed to when it is actually located outside the region of interest ($\leq$ 10%, at distance = 1.2 m). This suggests that our AP-specific minimum threshold α_{min} is a little more aggressive (higher) than ideal. Note that our approach provides 100% accuracy in eliminating any WiFi-connected devices are located 3 m or more beyond the region boundary.

We further test the accuracy of the outlier detection in the collaboration space and measure the false-positives and false-negatives across various sections. We consider the lift-lobby as the outside region, while the sections S1, S2 and S6 are inside the indoor space. Table 10 lists the accuracy of outlier detection logic with the device placed within the lift lobby, while Tables 11 and 12 enumerate the accuracy of the outlier detection logic when the device was placed within the 2 sections, namely S1 and S2. As observed from the results, the overall accuracy of outlier detection drops to 63%. The accuracy of the outlier logic is much lower (22%) when a device is placed near the boundary but within Section S1, while the accuracy is much higher (85.5%) when a device is similarly placed in section S2. Our key findings are:

- Compared to the research lab settings, the accuracy is comparatively much lower, due to the very minimal separation between the lift lobby and sections S1, S2 & S6. While a glass door separates the lift lobby from S1 and S6, S2 and the lift lobby are separated by a thicker wall. Moreover, the fingerprinted RSSI values for AP01 at landmark-15 and in the lift lobby are almost identical, differing by a measly 3 dBm. Due to unpredictable and continuous changes in the environment as well as the device-specific antenna characteristics, such small differences in RSSI can generate localization errors, making it difficult to segregate devices lying close to the separation boundary.



(a) Device Inside the Boundary

(b) Device Outside the Boundary

Fig. 12. (b) Outlier elimination with respect to distance from boundary.

Table 10
Outlier detector performance — Collaboration space (Lift lobby).

Client	Ground truth	Inferred-Legitimate	Inferred-Outlier	Crowd-Density
C1	Lift lobby	22.8%	77.2%	High
C2	Lift lobby	70.5%	29.5%	High
C3	Lift lobby	90%	10%	High
C1	Lift lobby	30.8%	69.2%	Low
C2	Lift lobby	3%	97%	Low
C3	Lift lobby	3%	97%	Low

Table 11
Outlier detector performance — Collaboration space (Section — S1).

Client	Ground truth	Inferred-Legitimate	Inferred-Outlier	Crowd-density
C1	S1	50%	50%	High
C2	S1	15%	85%	High
C3	S1	10%	90%	High
C2	S1	14%	86%	Low

Table 12
Outlier detector performance — Collaboration space (Section — S2).

Client	Ground truth	Inferred-Legitimate	Inferred-Outlier	Crowd-Density
C1	S2	92%	8%	High
C2	S2	78%	22%	High
C3	S2	95%	5%	High
C2	S2	77%	23%	Low

- Due to the relatively closer proximity of the APs (AP01 and AP02 are separated by a distance of ∼11 m), devices that move from the section S1 to the lobby, and vice versa, often remain attached to their current AP and do not generate new PROBE_REQUESTs. In such cases, the sparser cardinality of AP reports makes it harder to identify and eliminate outliers.
- The outlier detection logic performed significantly better in identifying devices located in section S2 as legitimate, compared to devices located in sections S1 and S6. In particular, devices located in section S2 have more unique/distinct fingerprints for the (AP01,AP02) tuple, compared to the landmarks in sections S1 and S6, thereby helping to reduce the underlying localization error.

6.3. Static device elimination

In this section, we discuss the results of our approach to eliminate the static devices from the workspaces. We exploited the data collected during the COVID-19 lockdown period to identify the ground-truth static devices, as the academic buildings were closed for employee access and we hypothesized that the WiFi data was generated only by the static devices present within the workspace. We utilize these devices as the ground truth, and compare this set of devices against the set of devices, inferred as *static* by our automated logic during the two week period prior to the start of the COVID-19 lockdown period. We observed the following key properties:

- When applied to the test set (devices observed prior the lockdown period), our static device filtering logic was successful in identifying *55%* of the ground-truth devices as static (true-positives), while *45%* was categorized as false-negatives. Devices classified as true-positives were observed to have a set of estimated locations that remained constant both across different periods of a single day, and across different days of the week.
- Among the 55% of the recognized static devices, 50% of the devices generated WiFi reports on all days of the week; the other 50% were reported only on 3–4 days of the week. This observation suggests that static devices do not necessarily generate location updates continuously, and underpins our decision to require static devices to be present for at least 3 days/week.
- Among the 8 devices that generated apparent false-negatives (i.e., we failed to identify as static, even though they continued to be observed during the lockdown period), 5 were found to have changed their estimated landmark location across different slots within the same day, thereby excluding them from membership in the static set. For the other 3 devices, their location was seen to remain unvarying within a day, but was seen to vary substantially across different days.

Overall, our static device elimination logic can itself be described as moderately successful, with an F1-score of 0.75. Our studies reveal that the automated identification of such static (or non-personal devices) is not a trivial problem. Indeed, the development of improved methods for identifying such non-personal devices (whether static or mobile, such as mobile TVs or robots) remains an open and challenging question. However, we shall shortly see (in Section 6.4) that even this modestly-accurate static device elimination mechanism helps to improve the final occupancy estimate.

6.4. Occupancy estimation

Direct, large-scale validation of the outlier detector logic is difficult, due to the difficulty in obtaining ground truth. Instead, we now provide

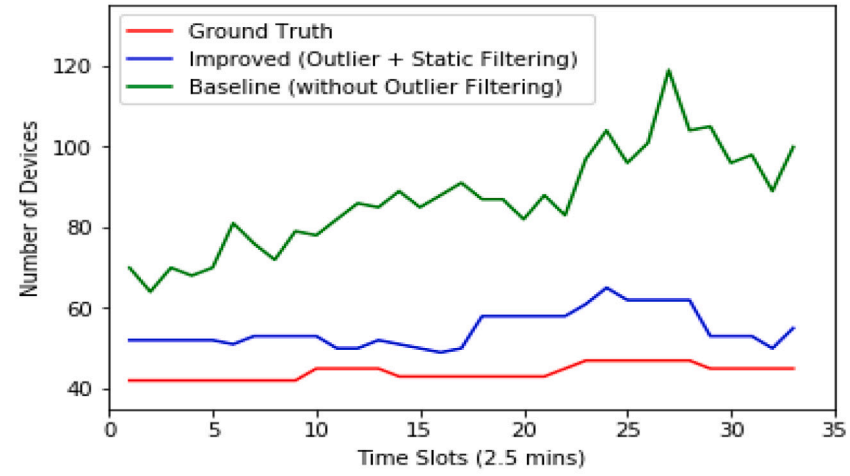


Fig. 13. Occupancy count for collaboration space.

an indirect evidence of the benefit of such outlier elimination — namely, a dramatic improvement in the estimation of *overall occupancy*. We analyzed the error in occupancy estimation, primarily across both the collaborative space and the residential dorm, where the occupancy exhibited significant fluctuation. (In contrast, the occupancy pattern of the access-controlled research lab is relatively stable.)

In the collaborative space, we also manually surveyed the occupants in sections {S1,S2,S3,S6} to record the number of WiFi-connected devices each occupant possessed during the experimentation period. We observed the entry-exit times of these occupants to estimate the number of ground-truth devices at each interval, over an 1.5 h observation window. In Fig. 13, we plot the ground-truth devices against the total number of devices estimated for two different crowd-levels (high and low). The device-count was estimated every 2.5 mins (with temporal-smoothing and outlier elimination), and compared with the baseline approach (which did not include threshold-based outlier elimination or temporal-smoothing). Our experimental results show a 72.8% reduction in the occupancy estimation error (compared to the baseline) under high crowd density, and an even more dramatic 91.6% reduction in the occupancy estimation error in a low-crowd environment. We make the following observations:

- The higher occupancy error experienced by the baseline is due to the devices from Sections S4 and S5 getting included within the region of study. This is also due to the erroneous inclusion of devices that are present outside the collaboration workspace but that are associated with one of the APs inside the space.
- The under-counting observed in the low-crowd scenario is due to the devices in Section S6 being occasionally marked as an outlier and being eliminated from the occupancy period.
- By observing the devices that were seen to be stationary throughout the 1.5 h observation window, we found that the temporal smoothing logic helped in improving the overall localization accuracy by *4%*.

Table 13
Average occupancy estimation error (collaborative space).

Baseline (without outlier or static filtering)	Improved (static filtering only)	Improved (cardinality + static + outlier filtering)
109.3%	96.9%	27.05%

To understand the relative contributions from the different filtering schemes, we plot, in Table 13, the average estimation error obtained by our improved algorithm, but with/without outlier elimination and static device filtering. In this specific case, most of the static devices were identified to be outside the region of interest, and thus could be

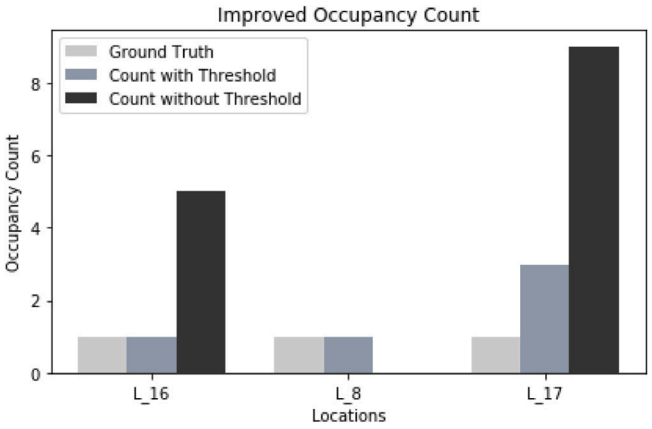


Fig. 14. occupancy count — with/without threshold-based outlier filtering.

eliminated purely by the use of outlier detection. However, static device filtering (only for devices localized to sections {S1,S2,S3,S6}) achieved an average 12.37% additional reduction in the occupancy error.

To study the occupancy estimation outcome in the residential building, we manually recorded the total occupancy in the public area over a 1-h observation window. We further differentiated the occupancy estimates across 3 different landmarks: L_8, L_{16} and L_{17}. Fig. 14 plots the ground truth as well as the estimate occupancy values, both with and without our threshold-based outlier elimination mechanism. Consistent with the prior exemplar (Table 3), we see that, without the threshold, the occupancy is consistently over-estimated, as the count includes several devices that attach to the indoor APs, even though the users are located outside the building. We observed that the introduction of our outlier-based elimination process results in an at least 80% reduction in the occupancy estimation error, across all 3 landmarks.

On closer examination, we see that the improvement in estimation accuracy is more dramatic for landmarks L_{16} and L_{17}. These two locations lie on the boundary of the building, making it more likely for them to include extraneous MNs that connect to the APs but are actually located outside. In contrast, for landmark $L8$, we see that, in the absence of the threshold-based filtering, the occupancy count (ground truth = 1) is erroneously estimated to be 0. This implies that, the device actually located at L_8 was incorrectly mapped to a different landmark. This example illustrates a secondary benefit of outlier detection: it not only eliminates devices located outside the fingerprinted area, but also helps to improve the accuracy of location estimation by rejecting location estimates that violate *landmark-specific thresholds* (Eq. (4)). An analysis of a 9-h data trace showed that we were able to eliminate (as outliers) approx. 18% of the clients that were initially mapped to a location within the fingerprinted areas of the building.

Results for the academic research lab are quantitatively similar to the Collaborative Space and are not detailed here, due to space considerations. As a quick summary, the use of suggested strategies was seen to reduce the average occupancy estimation error to 27.5%, compared to an average error of 109.3% obtained by the use of the baseline methods.

7. Conclusion

In this paper we tackled two practical problems existing in the implementation of accurate server-side indoor localization: (a) low AP cardinality for stationary clients and (b) over-counting of devices located outside the fingerprinted area. While the first problem might be less acute in venues where users typically have low residency times (e.g., in train stations), the second problem is universal. We tackled the low cardinality problem by explicitly delineating stationary periods for each MN, and using hitherto-discarded *stale* reports from other APs to augment the cardinality of AP measurements. We also tackled the over-counting or mis-attribution problem by effectively using a Voronoi-tessellation approach to devise a *per-landmark, per-AP* acceptable range of signal strength readings, and iteratively discarding location estimates that did not conform to these ranges. In addition, we also proposed a couple of additional filtering strategies, to mask the transient fluctuations in location and to eliminate static, non-personal devices present in such environments. Our empirical results prove that these mechanisms have clear and significant practical impact: (a) they reduce the median localization error for clients from ~6–8 m to 4 m; (b) they reduce the error of aggregated occupancy estimates by 80+%; and (c) temporal smoothing helps reduce errors by ~ 4%. In ongoing work, we are integrating these mechanisms into our occupancy monitoring system, and plan to utilize such occupancy estimates as part of smart HVAC and lighting control mechanisms for energy-efficient building operation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research/project is supported by the National Research Foundation, Singapore under its Strategic Capabilities Research Centres Funding Initiative. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore. The authors gratefully acknowledge the support of researchers from SMU's erstwhile LiveLabs research center, who helped provide the base WiFi RSSI data.
(Updated in Nov 2021)

References

[1] C. Yang, H. Shao, Wifi-based indoor positioning, IEEE Commun. Mag. 53 (3) (2015) 150–157.
[2] S. Depatla, A. Muralidharan, Y. Mostofi, Occupancy estimation using only wifi power measurements, IEEE J. Sel. Areas Commun. 33 (7) (2015) 1381–1393.
[3] S. Yiu, M. Dashti, H. Claussen, F. Perez-Cruz, Wireless rssi fingerprinting localization, Signal Process. 131 (2017) 235–244.
[4] S. Sen, B. Radunovic, R.R. Choudhury, T. Minka, You are facing the mona lisa: Spot localization using phy layer information, in: 10th International Conference on Mobile Systems, Applications, and Services, MobiSys '12, 2012.
[5] S. Sen, D. Kim, S. Laroche, K.-H. Kim, J. Lee, Bringing cupid indoor positioning system to practice, in: Proceedings of the 24th International Conference on World Wide Web, WWW '15, 2015.
[6] 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, SenSys '13, 2013.
[7] B.S. Ciftler, S. Dikmese, I. Guvenc, K. Akkaya, A. Kadri, Occupancy counting with burst and intermittent signals in smart buildings, IEEE Internet Things J. 5 (2) (2018) 724–735.
[8] R.K. Balan, A. Misra, Y. Lee, Livelabs: Building an in-situ real-time mobile experimentation testbed, in: Proceedings of the 15th Workshop on Mobile Computing Systems and Applications, HotMobile '14, 2014.
[9] P. Bahl, V.N. Padmanabhan, Radar: an in-building rf-based user location and tracking system, in: Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, 2000, 775–784. http://dx.doi.org/10.1109/INFCOM.2000.832252.
[10] S. Xia, Y. Liu, G. Yuan, M. Zhu, Z. Wang, Indoor fingerprint positioning based on wi-fi: An overview, ISPRS Int. J. Geo-Inf. 6 (2017) 135, http://dx.doi.org/10.3390/ijgi6050135.
[11] W. Kang, Y. Han, Smartpdr: Smartphone-based pedestrian dead reckoning for indoor localization, IEEE Sens. J. 15 (5) (2015) 2906–2916, http://dx.doi.org/10.1109/JSEN.2014.2382568.
[12] Q. Chang, S. Van de Velde, W. Wang, Q. Li, H. Hou, S. Heidi, Wi-fi fingerprint positioning updated by pedestrian dead reckoning for mobile phone indoor localization, in: J. Sun, J. Liu, S. Fan, X. Lu (Eds.), China Satellite Navigation Conference (CSNC) 2015 Proceedings, Vol. III, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015, pp. 729–739.

[13] D. Jaisinghani, R.K. Balan, V. Naik, A. Misra, Y. Lee, Experiences & challenges with server-side wifi indoor localization using existing infrastructure, in: Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, MobiQuitous '18, ACM, New York, NY, USA, 2018, pp. 226–235, http://dx.doi.org/10.1145/3286978.3286989.
[14] M. Youssef, A. Agrawala, The horus location determination system, Wirel. Netw. 14 (3) (2008) 357–374.
[15] J. Xiao, K. Wu, Y. Yi, L. Wang, L.M. Ni, Pilot: Passive device-free indoor localization using channel state information, in: 2013 IEEE 33rd International Conference on Distributed Computing Systems, 2013, pp. 236–245.
[16] B. Li, J. Salter, A.G. Dempster, C. Rizos, Indoor positioning techniques based on wireless lan, in: LAN, First IEEE International Conference on Wireless Broadband and Ultra Wideband Communications, 2007, pp. 13–16.
[17] K. Kaemarungsi, P. Krishnamurthy, Properties of indoor received signal strength for wlan location fingerprinting, in: The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, MOBIQUITOUS, 2004, pp. 14–23, http://dx.doi.org/10.1109/MOBIQ.2004.1331706.
[18] S. Mazuelas, A. Bahillo, R.M. Lorenzo, P. Fernandez, F.A. Lago, E. Garcia, J. Blas, E.J. Abril, Robust indoor positioning provided by real-time rssi values in unmodified wlan networks, IEEE J. Sel. Top. Sign. Proces. 3 (5) (2009) 821–831, http://dx.doi.org/10.1109/JSTSP.2009.2029191.
[19] J. Talvitie, M. Renfors, E.S. Lohan, Distance-based interpolation and extrapolation methods for rss-based localization with indoor wireless signals, IEEE Trans. Veh. Technol. 64 (4) (2015) 1340–1353, http://dx.doi.org/10.1109/TVT.2015.2397598.
[20] K. El-Kafrawy, M. Youssef, A. El-Keyi, A. Naguib, Propagation modeling for accurate indoor wlan rss-based localization, in: 2010 IEEE 72nd Vehicular Technology Conference - Fall, 2010, pp. 1–5, http://dx.doi.org/10.1109/VETECF.2010.5594108.
[21] M.B. Kjaergaard, Indoor location fingerprinting with heterogeneous clients, Pervasive Mob. Comput. 7 (1) (2011) 31–43, http://dx.doi.org/10.1016/j.pmcj.2010.04.005.
[22] M. Wang, Z. Zhang, X. Tian, X. Wang, Temporal correlation of the rss improves accuracy of fingerprinting localization, in: IEEE INFOCOM 2016 - the 35th Annual IEEE International Conference on Computer Communications, 2016, pp. 1–9.
[23] S. Li, M. Hedley, K. Bengston, D. Humphrey, M. Johnson, W. Ni, Passive localization of standard wifi devices, IEEE Syst. J. 13 (4) (2019) 3929–3932.
[24] Z. Li, T. Braun, D.C. Dimitrova, A passive wifi source localization system based on fine-grained power-based trilateration, in: 2015 IEEE 16th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2015, pp. 1–9.
[25] L. Sun, C. Sinong, Z. Zheng, L. Xu, Mobile device passive localization based on ieee 802.11 probe request frames, Mob. Inf. Syst. 2017 (2017) 1–10.
[26] J. Xiong, K. Jamieson, Arraytrack: A fine-grained indoor location system, in: 10th USENIX Conference on Networked Systems Design and Implementation, Nsdi'13, 2013.
[27] J. Wilson, N. Patwari, Radio tomographic imaging with wireless networks, IEEE Trans. Mob. Comput. 9 (5) (2010) 621–632.
[28] A.R. Hunt, A wideband imaging radar for through-the-wall surveillance, in: Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense III, Vol. 5403, SPIE, 2004, pp. 590–596.
[29] A. Networks, Rtls - integrating with the rtls data feed, 2012, URL https://community.arubanetworks.com/aruba/attachments/aruba/unified-wired-wireless-access/23715/1/RTLS_integrationv6.docx.
[30] I. Meraki, Location analytics. URL https://meraki.ciscocom/solutions/location-analytics.
[31] X. Hu, L. Song, D. Van Bruggen, A. Striegel, Is there wifi yet?: How aggressive probe requests deteriorate energy and throughput, in: Proceedings of the 2015 Internet Measurement Conference, IMC '15, ACM, New York, NY, USA, 2015, pp. 317–323, http://dx.doi.org/10.1145/2815675.2815709.
[32] J. Freudiger, How talkative is your mobile device?: An experimental study of wi-fi probe requests, in: Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks, WiSec '15, ACM, New York, NY, USA, 2015, pp. 8:1–8:6, http://dx.doi.org/10.1145/2766498.2766517.
[33] D. Jaisinghani, V. Naik, S.K. Kaul, R. Balan, S. Roy, Improving the performance of wlans by reducing unnecessary active scans, 2018, arXiv:1807.05523.
[34] D.F. Watson, Computing the n-dimensional delaunay tessellation with application to voronoi polytopes*, Comput. J. 24 (2) (1981) 167–172, http://dx.doi.org/10.1093/comjnl/24.2.167.

Anuradha Ravi is a research scientist working with Living Analytics and Research Centre at Singapore Management University. She received her Ph.D. from Indian Institute of Technology Roorkee, India in 2016. Her research interests are in Mobile Computing, Wireless Networking and Cyber Physical Systems. Over her research career, Anuradha has had experiences in working with real-time testbeds, which has gained her best paper awards and accolades at conferences. Anuradha has worked as an Assistant Professor and tutor in academic institutes. She has also worked in IT MNCs in India, prior to pursuing Ph.D.

Archan Misra is Professor, and the Associate Dean of Research, in the School of Information Systems at Singapore Management University (SMU). Archan leads or has led a number of multi-million dollar, large-scale research initiatives at SMU, including the Center for Applied Smart-Nation Analytics (CASA) and the LiveLabs research center, and is a recent recipient of the prestigious Investigator grant (from Singapore's National Research Foundation) for sustainable man–machine interaction intelligence. Over a 20+ year research career spanning both academics and industry (at IBM Research and Bellcore), Archan has published on, and practically deployed, technologies spanning wireless networking, mobile & wearable sensing and urban mobility analytics. His current research interests lie in ultra-low energy execution of machine intelligence algorithms using wearable and IoT devices. Archan holds a Ph.D. from the University of Maryland at College Park, and chaired the IEEE Computer Society's Technical Committee on Computer Communications (TCCC) from 2005–2007.