Pers Ubiquit Comput (2016) 20:83–96
DOI 10.1007/s00779-015-0899-z

ORIGINAL ARTICLE

LoCo: boosting for indoor location classification combining Wi-Fi and BLE

Matthew Cooper[1] · **Jacob Biehl**[1] · **Gerry Filby**[1] · **Sven Kratz**[1]

Received: 9 January 2015 / Accepted: 17 November 2015 / Published online: 12 January 2016

Abstract In recent years, there has been an explosion of devices that leverage location to provide users novel and engaging experiences. However, many applications fail to realize their full potential because of limitations in current location technologies. Current frameworks work well outdoors but fare poorly indoors. In this paper, we present LoCo, a new framework that can provide highly accurate room-level indoor location. LoCo does not require users to carry specialized location hardware—it uses radios that are present in most contemporary devices and, combined with a boosting classification technique, provides a significant runtime performance improvement. We provide experiments that show the combined radio technique can achieve accuracy that improves on current state-of-the-art Wi-Fi-only techniques. LoCo is designed to be easily deployed within an environment and readily leveraged by application developers. We believe LoCo's high accuracy and accessibility can drive a new wave of location-driven applications and services.

Keywords Indoor location detection · Multi-radio indoor positioning · Location-aware application frameworks

1 Introduction

Leveraging physical location to drive meaningful, context-specific user experiences has been a hallmark of mobile computing from the very beginning. While GPS and other satellite-based technologies provide pervasive outdoor location, reliable indoor location remains out of reach despite years of research. A primary adoption inhibitor to many proposed solutions is that they require users to wear or carry specialized devices for tracking, e.g., badges that recognize infrared beacons [37] or specialized radio devices.

Recently, there has been interest in leveraging existing infrastructure and devices to determine indoor location. Specifically, Wi-Fi-based localization can be deployed with smartphones such as Apple's iPhone and Google's Android platform. The typical approach to Wi-Fi localization is to use the client's received signal strength indicator (RSSI) of fixed Wi-Fi access points (APs) as a measure of distance between the APs and the device. These distances, along with the locations of the APs, are used in a multilateration calculation to predict the device's location. Due to the inherent error of these calculations, most tracking results have several meters of error.

An alternative to multilateration is location classification using RSSI scans collected by the mobile device. The approach trades precision for accuracy; the approach does not track a device's location as (x, y) coordinates, but rather provides a high confidence prediction of what location the device is in using a predefined discrete set of locations.

In this work, we propose new techniques that seek to improve overall accuracy while also making them more practical to deploy and be leveraged by application developers. Our new framework, called LoCo, achieves increased room-level accuracy by combining Wi-Fi RSSI

✉ Matthew Cooper
cooper@fxpal.com

Jacob Biehl
biehl@fxpal.com

Gerry Filby
filby@fxpal.com

Sven Kratz
kratz@fxpal.com

[1] FX Palo Alto Laboratory, Palo Alto, CA, USA

signals with signals from a new radio type, Bluetooth low energy (BLE). While our approach utilizes a relatively new technology, it is important to note that many existing phones already contain BLE radios (e.g., Apple's iPhone 4S and newer, Samsung Galaxy SIII and S4) and off-the-shelf BLE beacons are becoming increasingly available with steadily decreasing costs (e.g., *iBeacons*[1], *ToDs*[2], and *Estimotes*[3]).

The framework we present is designed to achieve room-level location accuracy that is meaningfully higher than previously developed techniques in order to enable new indoor location-based applications. State-of-the-art Wi-Fi fingerprinting methods (e.g., [6]) are documented to provide accuracy near 85 % for room-level localization. For applications such as equipment tracking or inventory management, this degree of accuracy is acceptable. However, for location applications and services that we envision, such as robustly detecting the presence of an individual in a room, this level of performance falls short. To improve accuracy, we extend Wi-Fi RSSI-based techniques with the use of BLE beacons that are deployed explicitly for indoor location identification. We show that integrating BLE enhances the accuracy of both our earlier work based on Wi-Fi [5] as well as a high-performing Wi-Fi RSSI-based baseline [6].

We present a series of experiments to validate this approach. First, we show that using BLE beacons alone allows for high-performance room-level location identification. Additionally, we show that combining signal strength measurements from the BLE beacons with those from Wi-Fi access points (APs) consistently improves performance over using either modality alone. In all configurations, the proposed approach dramatically reduces computational complexity, thus enabling location identification to be performed more frequently and on resource constrained devices (e.g., mobile smartphones). We perform additional analysis of the utility of the multiradio approach for location identification at sub-room level in an "open plan" office space. Finally, we describe experiments that demonstrate that while multiradio location classification is superior, it also is resilient across a range of densities of Wi-Fi access points.

2 Related work

We have structured the review of related work to cover previous contributions on indoor localization algorithms using only Wi-Fi, fusing Wi-Fi with other wireless technologies, and alternate sensing technologies.

2.1 Wi-Fi-only localization

There is a great deal of prior work on using Wi-Fi RSSI measurements to locate devices and their users. Many methods aim for precise physical localization (i.e., multilateration) rather than quantizing position to a room or region. While room information may not be required for localization, the locations of wireless access points (APs) typically are needed.

Liu et al. [19] provide a comprehensive survey of Wi-Fi location determination methods. They contrast triangulation methods that use geometric analysis to estimate location with scene analysis methods that match test measurements against templates of locations' radio frequency characteristics. More recent triangulation work has focused on approaches that model radio propagation statistically in combination with previously defined templates for the signal strengths observed at various locations [21, 40]. In [40], locations are also clustered according to whether base stations are in range to accelerate location determination by "incremental triangulation". Other approaches include localization via genetic programming [7], Bayesian networks [20], and combined mapping of locations with tracking of devices as a semi-supervised learning problem [27]. There has been work that performs trilateration through time of flight and/or angle of arrival (e.g., [8, 24, 39]). While sophisticated, these approaches require modification and/or special software to run on the Wi-Fi base stations. More importantly, while gains have been made in addressing multipath issues in dense environments like office buildings (e.g., [32]), current techniques have not been shown to be robust in real-world applications.

Matching methods represent the set of RSSI values at a given location by a known (absolute) set of values and minimize a distance measure to assign a test data point to a known location [6, 9, 18, 31]. Supervised classification for location estimation is somewhat less prevalent. Support vector machines and naive Bayes classifiers compared poorly with the Redpin and k-nearest-neighbors (kNN) approach in [18]. The work in [26] describes a method that combines generative modeling using EM to help address missing data and discriminative modeling with SVMs for localization.

Our system uses boosting [11] to build a set of room-specific classifiers. These ensemble classifiers combine simple decision stumps that threshold the difference between two RSSI measurements. The boosting training procedure greedily selects decision stumps corresponding to RSSI measurements that discriminate each room from the others. In contrast to many other methods, the *margin features* we employ emphasize the relative ordering of RSSI measurements more than their absolute values,

[1] http://support.apple.com/en-us/HT202880.

[2] http://www.todhq.com/.

[3] http://www.estimote.com/.

improving robustness. Furthermore, boosting has very low computational requirements at classification time, providing both substantial complexity reductions relative to matching (i.e., kNN) methods, and superior accuracy.

We previously reported on the use of boosting for Wi-Fi only classification in [5]. In this work, we introduce the use of the BLE beacons and present a much expanded evaluation with additional experimental results that explore the value and implications of a combined radio approach to room-level indoor location detection.

2.2 Fusing Wi-Fi and other wireless technologies

Rodriguez et al. [30] study Wi-Fi, Bluetooth, and Zigbee for indoor localization. They use kNN on the raw RSSI information for matching locations and observe an increase in location accuracy when multiple wireless sources are used. They mention problems with interference between wireless devices as more beacons are added. In the present work, this problem is mitigated by using low transmission energy on the Bluetooth beacons.

Incorporating additional location information from Bluetooth is proposed by [4]. This work uses Bluetooth beacons as an ancillary technology to narrow down the search space for Wi-Fi fingerprinting. They derive their results from simulations and modeling but did not conduct experiments with real hardware and environments to verify their results.

2.3 Alternate localization technologies

Localization systems have also been proposed using technologies other than Wi-Fi or Bluetooth. The most used alternative technologies are ultrasound, (active) radio frequency identification (RFID), and infrared (IR). The *ActiveBat* system [13] uses ultrasonic transmitters attached to devices and room-mounted ultrasound sensors that use time-of-flight multilateration for localization. While ActiveBat achieves a high degree of accuracy, it relies on relatively heavy instrumentation (100 sensors for an area of 280 m^2) of the localization space as well as the addition of hardware to all tracked devices. The *Cricket* system [28] follows up RF transmissions with ultrasound bursts. The difference in speed between RF and sound can be used to calculate the distance to a beacon.

The *SpotOn* [16] system uses multiple RFID readers for location triangulation of active RFID tags. They reported relatively poor accuracy, providing location estimates within voxels with 3 m side lengths. Ni et al. [23] improved upon this system by adding fixed active RFID tags to better calibrate the base stations. Using kNN techniques, their system achieves a relatively precise (1–2 m) localization accuracy, but at the cost of requiring a significant amount of fixed reference tags (one per m^2 covered).

ActiveBadge [37] was an early location system based on badges augmented with IR transmitters and IR detectors in the environment. The main limitation of using IR is that as a line-of-sight technology, badge occlusion poses problems. Also, the range is relatively limited (the authors state a range of about 6 m).

3 LoCo location framework

The LoCo framework consists of three main components: a deployment of Wi-Fi access points and Bluetooth low energy (BLE) beacons, a client service running on a smartphone, and a classification engine. We discuss each of these in detail below.

3.1 Constellation of Wi-Fi APs and BLE beacons

LoCo collects signal strength information from two different types of wireless transmitters. As in past work, the framework leverages the received signal strength indicator (RSSI) of the SSID beacons coming from in-range 802.11 Wi-Fi access points (APs). These values are easily accessible by most Wi-Fi frameworks, including most smartphone operating systems. Wi-Fi signals travel relatively far; a single AP may serve hundreds of square meters in an office building, providing data coverage for several users and offices. As a result, there can be significant distance, walls, objects, between the AP and the receiving device that can introduce variable attenuation of the signal.

In contrast to the longer range Wi-Fi signal, we also incorporate small Bluetooth low energy (BLE) beacons. Unlike "Classic Bluetooth" where a signal can travel a significant number of meters, BLE devices can be configured such that the signal only travels a few meters. The beacons we use (Fig. 1) are based on the BlueGiga BLE112 SoC. These devices transmit and receive on the 2.4-Ghz ISM radio band, the same band used by 802.11 b/g/n. The transmit power of these beacons is an order of magnitude lower than that of Wi-Fi. To address interference, the beacons divide the 2.4-Ghz band into 40 channels of 2 Mhz each from 2402 to 2480 Mhz. Three of these channels are dedicated to advertising packets and the remaining 37 to reliable data exchange in the connected mode.

In LoCo, we utilize the advertisement packets emitted by the beacons. We have modified the firmware of an off-the-shelf beacon [35] to include in the advertisement's payload the unique identity of the beacon. We also add a representation of the beacon's battery level to identify beacons with low battery levels prior to failure. In each

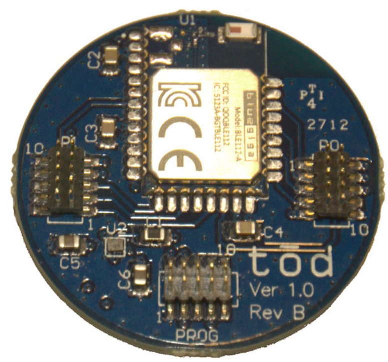


Fig. 1 The Bluetooth low energy beacons used as part of the LoCo location framework

advertising event, the advertising packet is sent on each of the three advertising channels sequentially. The advertising channels are configured at frequencies largely unused by common Wi-Fi base stations. The time delay between advertising events is configurable to help manage battery consumption. In our deployment, configurable RF transmit power is set to the minimum and the advertising packets are sent every 2 s. At this power setting, the device only consumes 27 mA of power to transmit and 0.5 μA when sleeping, thus allowing it to operate for over a year on a single CR2032 coin cell battery, or for several years when configured with a two "AA" cell battery pack.

For an early pilot experiment, we covered an approximate 1100 m^2 portion of our laboratory with [illegible] beacons; in later experiments, we covered approximately 1900 m^2 with 75 beacons. Even in the small quantities that we produced, the production cost for each beacon is approximately $25 USD/unit, resulting in a total of approximately $1900 USD, an order of magnitude less than many on-market commercial location solutions. Further, given the simplicity of these devices, it is likely that their price will fall significantly as the economies of scale improve.

While we built out our beacons to provide some useful functionality (unique payloads with batter level, large battery capacity, etc.), they are relatively similar in functionality to Apple's iBeacon specification. With slight modification to our client software, a deployment of off-the-shelf OEM iBeacons could be used with our approach. As these beacons become more popular, we believe it is further evident that our proposed system could have wide-ranging applicability.

We created custom cases for these beacons using a common 3D printer (see Fig. 2). We coated the inside of the cases with a special graphite paint formulated to absorb up to 90 dB of RF energy. This allows us to control, in a crude way, the propagation of the BLE signals. Since the beacons are deployed by attaching them to the ceiling (as depicted in Fig. 3), the coating helps reduce signals

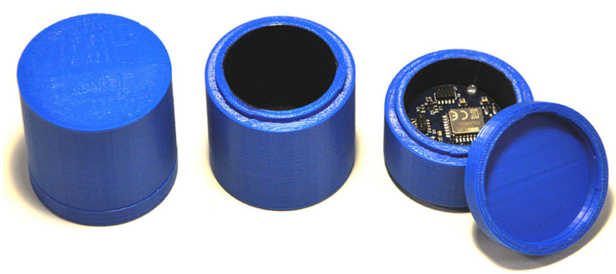


Fig. 2 The cases that we custom built using a 3D printer. The cases are made of ABS plastic, coated with a special graphite paint that is designed to absorb RF signals. Using this coating, the cases control, in a very crude way, the propagation of the BLE signals

traveling between rooms along the rails that hold the ceiling tiles in place. A variety of case heights is used, ranging from 10 mm for large spaces, to 45 mm for small, cubicle-sized locations. Experiments showed the distance from the BLE antenna to the aperture created by the uncoated portion of the case gave us limited control of the desk/floor area a beacon's signal would cover. Figures 4 and 5 show the placement of the Wi-Fi APs and BLE beacons used in our trial deployment and experiments.

The Bluegiga BLE112 firmware provides for control of the power output of the Bluetooth radio signal amplifier in the arbitrary range "1–15". In the initial configuration, we set the power to "1" to localize the radio transmission as much as possible. This was the setting for the beacons for the first experimental data set we collected as described below. However, this configuration was incompatible with the iBeacon use case which we rely on to trigger events in iOS clients. As a result, we increased the radio output to "5", thus increasing the range and visibility of the beacons as evidenced in the second data set we collected for our experiments.

3.2 Smartphone client service

A relatively simple service runs on users' smartphones to collect Wi-Fi and BLE signal measurements. We created an initial implementation of this scanning service using Google's Android OS. We use the internal API to retrieve a list of in-range Wi-Fi APs and the RSSI from those APs' SSID advertisements. A native Bluetooth low energy API was not released until Android 4.3 (API level 18). To support devices not running 4.3, we also added support for native BLE APIs that have been made available by the various device manufacturers. For instance, this enabled popular, yet older phones like the Samsung Galaxy SIII devices (model I-9300) running Android 4.1.2 to be compatible with LoCo. We used the SIII as well as LG Nexus 4 and LG Nexus 5 devices in our experiments and evaluation below.

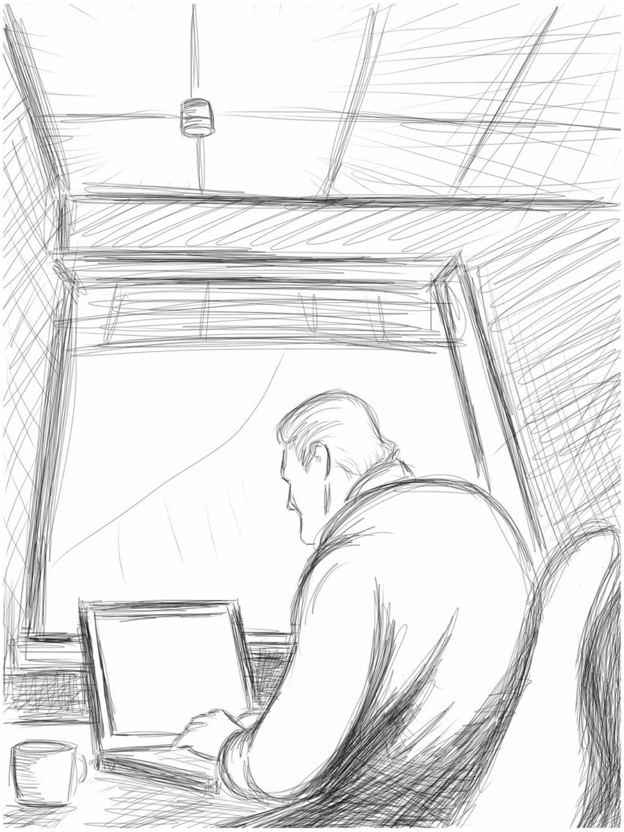


Fig. 3 A sketch depicting how the BLE beacons are deployed on ceilings in our environment

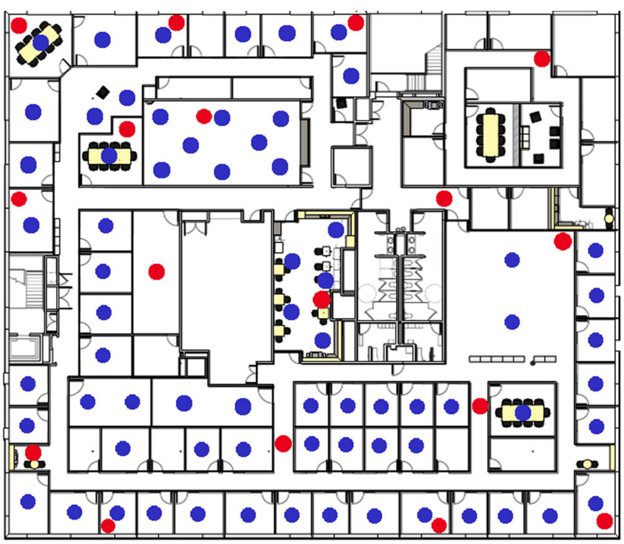


Fig. 4 Floor plan showing the placement of both the BLE beacons (*blue dots*) and Wi-Fi APs (*red dots*) used in our first data collection. Note that we did not deploy beacons in closets, machine/server rooms, or restrooms (color figure online)

To make the LoCo framework available to application developers, the Android client is implemented as an Android service. With fewer than 30 lines of code, an app developer can create a dependence on LoCo and make queries to the service for the device's location. Centralizing in a service not only makes LoCo easily available to multiple apps, the model also ensures that only one instance of the scanning routine is running at a given time. This eliminates duplicate scanning that can reduce battery performance. When an application subscribes to LoCo, it can also specify how often the framework should determine its location.

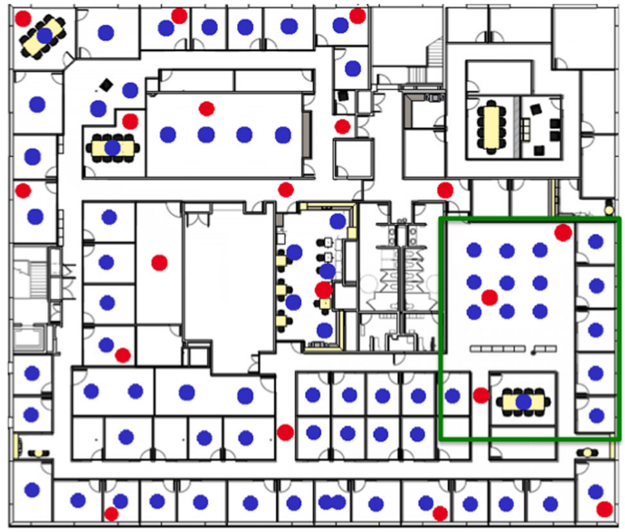


Fig. 5 Floor plan showing the placement of both the BLE beacons (*blue dots*) and Wi-Fi APs (*red dots*) used in our second experiment (larger deployment). The region enclosed by the green rectangle is used for the smaller scale experiment focused on the open-plan area on the right side (color figure online)

3.3 Classification engine

The LoCo framework can process scan data in two ways. The classifier along with a trained definition library can be loaded on to the smartphone allowing classification to be performed on device. As we describe below, our technique has low computational complexity, allowing it to run on smartphones without consuming a significant amount of energy. Alternatively, the scan results can be sent as a JSON structure to a cloud-based classification server. With this method, classification can be performed and stored on a server, allowing for external applications to leverage web services to query for a device's location.

We use the ensemble learning method boosting [11, 14] to estimate the device's room. While straightforward matching methods such as [6] provide high-level performance, they typically require the storage of a search data structure that grows with the training set size. Also, the application of specialized data structures for efficient kNN search with nonmetric distance measures can be difficult

[3]. Accelerated versions of kNN also scale poorly with feature dimensionality, limiting expansion of the location system to areas with large numbers of APs and beacons.

We use rooms throughout to generically refer to locations of interest; however, the method is applicable to any discrete spatial quantization. Specifically, multiple rooms can be collapsed into a single location, and a single room can be subdivided into multiple locations for classification purposes; this requires only appropriate corresponding modifications to the training procedure. This is explored in Sect. 4.2

Boosting methods have several appealing attributes, including simplicity, interpretability, and scalability. As a supervised classification method, the bulk of its computation occurs in an *off-line* training procedure. While binary boosting algorithms have been extensively analyzed and tested, multiclass boosting (such as the room identification problem here) remains an area of active research [17, 22]. We use a one versus all formulation such that the estimated room is simply the maximum score over the set of per-room classifiers:

$$\mathrm{room}^*(S) = \operatorname*{argmax}_{\mathrm{room}} F_{\mathrm{room}}(S) \tag{1}$$

For each room, we construct a binary classifier that outputs a score representing the probability that the RSSI scan vector S was observed in that room:

$$F_{\mathrm{room}}(S) = \sum_m \alpha_m h_m(S). \tag{2}$$

Each per-room classifier combines “weak learners”, h_m according to the scalar weights α_m. The weak learners are decision stumps that compare a scalar feature to a threshold θ_m:

$$h_m(S) = \begin{cases} 1 & X_m \geq \theta_m \\ 0 & \text{otherwise} \end{cases}. \tag{3}$$

In training, the thresholds θ_m are tuned to minimize error. We define the feature vector comprised of elements X_m that is computed from each RSSI vector S below.

To assemble a rich pool of weak learners from which to construct the per-room classifiers, we propose *margin features*. Given the observed RSSI vectors, we compute the set of unique pairwise differences (margins) between the vectors' elements. For an environment with B total BSSIDs and beacons, the resulting margin feature vectors have size $B \cdot (B-1)$. Intuitively, these features express coarse order information for pairs of BSSIDs/beacons.

Specifically, we transform the RSSI vector $S \in R^B$ into a margin feature vector with elements:

$$X_m = S(a_m) - S(b_m), \tag{4}$$

for $a_m, b_m \in \{1, \cdots, B\}$. Missing RSSI values for specific BSSIDs or beacons in the training set are set to a nominal

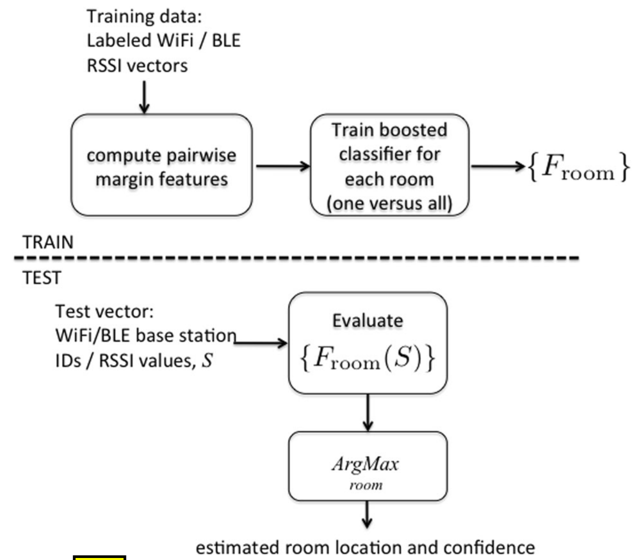


Fig. 6 Graphic describing the data flow using boosted classification with margin features for room identification. Processing above the *dashed line* (TRAIN) is performed once offline, given a set of labeled training data. Processing below the *dashed line* (TEST) is performed for each test scan

value, R_{min} to indicate they were not visible to the mobile client. Thus, the fact that specific beacons or BSSIDs are not visible at specific locations is incorporated into the features.

The margin features computed from the training scans form the input to classifier training. The training procedure identifies a location-specific set of weak learners of the form in (3) that best discriminates that location from all others. The weak learners and their relative weights (α_m in (2)) are learned in a greedy iterative procedure that optimizes error using a per-sample weighting over the training data [14]. These per-sample weights are repeatedly adjusted to emphasize misclassified samples. This process is repeated for each location to construct the set of classifiers $\{F_{\mathrm{room}}\}$ in Fig. 6. The output of training F_{room} is the specification of its selected weak learners, h_m and their weights α_m in (2). Each weak learner is determined by two hardware IDs a_m, b_m (MAC addresses of specific Wi-Fi BSSIDs and BLE beacons) and a threshold, θ_m. We employ a multiclass extension of the discrete adaboost classifier described in [41] (SAMME—algorithm 2).

For location determination, each of the per-room classifiers is applied to a given test RSSI scan vector S_{test}. We compute only the required set of RSSI differences of (4) that were selected in classifier training. These differences are compared to the thresholds (θ_m) determined during training and then combined linearly per (2). The final room estimate is determined by comparing the scores $\{F_{\mathrm{room}}(S_{\mathrm{test}})\}$ and selecting the maximum per (1). If any BLE beacons or Wi-Fi BSSIDs that contribute to a per-room classifier are not present in S_{test}, the missing values are set to $R_{\min}$. Figure 6 depicts the data flow. Note that the bulk of the computation occurs in the off-line training procedure. The training complexity is governed by the size of the available feature pool (weak learners) and the number of training samples. The test complexity depends only on the number of *selected* weak learners. Thus, unlike matching methods, classification test complexity is decoupled from the training set size.

4 Experiments

We report results of experiments designed to investigate:

- classification performance and time complexity,
- spatial granularity for defining locations,
- beacon and Wi-Fi access point density, and
- distance-based error analysis.

To validate our system, we collected two sets of data from a single floor of our office. The RSSI values were recorded by a variety of Android mobile devices that included Samsung Galaxy SIIIs, LG Nexus 4s, and LG Nexus 5s. The primary goals of the experiments are two-fold. We first assess the utility and limitations of the BLE beacons for location classification. Secondly, we benchmark our boosting method for room-level location classification against existing methods.

Because the BLE beacons are explicitly deployed for location sensing, we consider first a simple baseline that associates each location with the hardware ID of the beacon with maximal RSSI in its corresponding scans in the training set. At classification time, we simply determine the BLE beacon with maximal RSSI in each test scan and then assign the location associated with that BLE as the estimate. This simple method provides the first baseline method for comparison and is denoted "MAX BLE" in Tables and Figures below. We also evaluated versions of the Redpin system [6], which we believe represents a state-of-the-art method for room classification [18]. We follow their basic matching approach using kNN classification with $k = 5$. The matching is based on a distance measure between RSSI vectors X, Y defined as:

$$
\begin{aligned}
d_{\text{RP}}(X,Y) = {} & \alpha \sum_{b} [\![(X(b) > 0) \wedge (Y(b) > 0)]\!] \\
& - \beta \sum_{b} [\![(X(b) > 0) \oplus (Y(b) > 0)]\!] \\
& + \gamma\, C(X,Y),
\end{aligned}
\tag{5}
$$

where b indexes the hardware IDs of the BSSIDs and beacons in X and Y. The first summation counts the BSSIDs/beacons that appear in both X and Y. The second summation penalizes BSSIDs/beacons that are visible in exactly one of X or Y. We use negated euclidean distance as the correlation measure C, and set $\alpha = 1.0, \beta = 0.4, \gamma = 0.2$. For classification, brute force kNN search is executed for the given test RSSI vector. The classification is determined by the majority vote among the k nearest training samples. Redpin was developed for use with Wi-Fi RSSI data; however, we also report results for the method using the BLE RSSI data to provide a broader sense of the beacons' utility. We also naively concatenate the Wi-Fi and BLE RSSI data to establish a combined Redpin baseline. For the boosting classification system described here, we train classifiers using the BLE and Wi-Fi RSSI data both individually and in combination. For this, we compute margin features from the RSSI data following (4), and run 45–90 iterations of classifier training off-line to learn each per-location classifier. For testing, we apply the per-location classifiers to each test scan and return the highest scoring location as the estimate. Unless otherwise noted, we report accuracy and timing results averaged over ninefold cross-validation, and train all boosting classifiers for 90 iterations. Wall timing results here are computed using a PC with a 2.8-GHz AMD processor.

4.1 Classification accuracy and efficiency

The first data set consists of 1181 scans collected in 55 locations in which RSSI values are observed from a total of [illegible] BLE beacons and 159 unique Wi-Fi BSSIDs. The large number of Wi-Fi devices is the result of two conditions. First, the 17 APs that we used are commercial Cisco APs that can service multiple networks and provide access across multiple frequencies (mainly channels on 2.4 and 5 GHz). These APs produced 42 unique BSSIDs. Second, because our office shares a building with other companies and has close neighbors, the remaining unique BSSIDs are from these organizations' APs. The data set contains an average of 19.05 scans per location (SD = 5.21).

Table 1 summarizes the experimental results for the first data set. The right column shows the average time required to classify one test scan. The top row (MAX/BLE) shows results using the predefined map of the BLE beacons. This approach simply determines the maximum BLE RSSI and returns the corresponding location of the BLE as the estimate. It has lower accuracy, but demonstrates the value of a simple and efficient method using the BLE beacons. The next three rows show results for the Redpin method using BLE beacons alone (Redpin/BLE), Wi-Fi alone (Redpin/Wi-Fi), and the combined mode (Redpin/BLE + Wi-Fi). The bottom three rows show corresponding results for location classification via boosting. Throughout, we use boldface to indicate the result with the highest accuracy in the tables.

Table 1 Classification accuracy and (wall) timing results for the first data set with 55 locations and 1181 scans

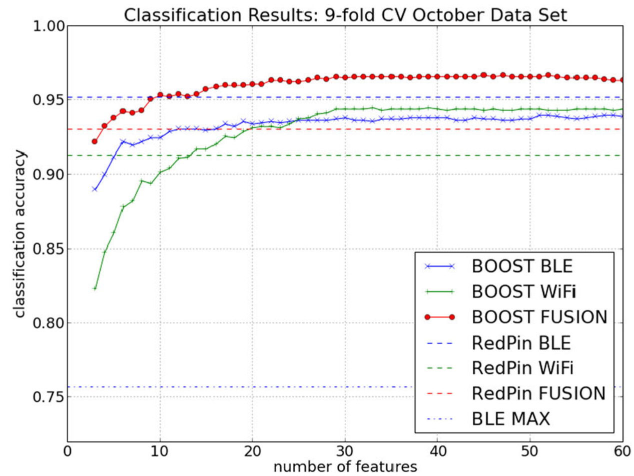
Method	RSSI data	Accuracy	Time (s)
MAX	BLE	0.757	0.00072
Redpin [6]	BLE	0.952	1.354
	Wi-Fi	0.913	3.054
	BLE + Wi-Fi	0.930	4.335
Boosting	BLE	0.937	0.00492
	Wi-Fi	0.943	0.00562
	BLE + Wi-Fi	**0.966**	0.00431

Accuracy is computed using ninefold cross-validation. Test timing is averaged per RSSI scan in seconds. The boosting results here use 45 iterations of classifier training

Table 2 Classification accuracy and (wall) timing results for the second data set with 56 locations and 2294 scans

Method	RSSI data	Accuracy	Time (s)
MAX	BLE	0.629	0.00159
Redpin [6]	BLE	0.964	2.0835
	Wi-Fi	0.950	7.2236
	BLE + Wi-Fi	0.973	9.165
Boosting	BLE	0.965	0.00627
	Wi-Fi	0.956	0.00653
	BLE + Wi-Fi	**0.991**	0.00660

Accuracy is computed using ninefold cross-validation. Test timing is averaged per RSSI scan in seconds. The boosting results here use 90 iterations of classifier training



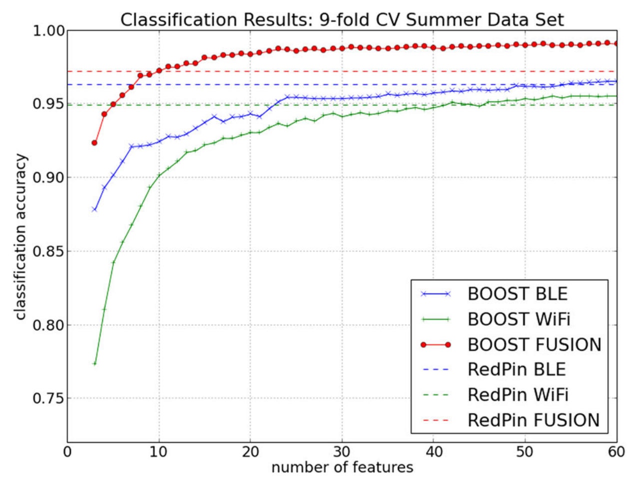


Fig. 7 The plots show the accuracy of different versions of both baselines and our system on the first (*top*) and second (*bottom*) experimental data sets. The boosting classification systems show variation with the number of training iterations (selected weak learners) along the *x*-axis, while the baseline systems are horizontal lines in the plots

The single modality results show consistent levels of accuracy for both Redpin and boosting in the range 0.91–0.95. The Redpin BLE-only results outperform the boosting BLE-only results (boosting/BLE) by a small margin, while boosting outperforms Redpin using Wi-Fi only. Also, BLE and Wi-Fi are able to provide complementary information that is exploited by the boosting classification method in the fusion (boosting/BLE + Wi-Fi) results to improve accuracy. The BLE + Wi-Fi boosting classifier performs the best among the evaluated methods with significantly lower computational overhead than the Redpin systems.

The second data set is larger scale in terms of the number of BLE beacons, 74 and the number of Wi-Fi AP beacon IDs, 254. In addition, this data set uses the BLE beacons at the higher power setting producing generally stronger BLE RSSI values. The mean BLE RSSI value from the first data set is -95.012 (std $= 4.41$), while for these data the mean is -91.157 (std $= 6.35$)[4]. Between the collections of the two data sets used, our company deployed additional Wi-Fi access points to support unrelated research activities. This data set consists of 2294 total RSSI scans collected using the same Android devices as before, with an average of 31 scans per location (std $= 5.73$). The experimental results appear in Table 2 and include the same baseline and boosting location classification configurations as before.

Figure 7 graphically displays performance on the two data sets for the boosting location classifiers. As more RSSI margin features and their corresponding weak learners are integrated, performance improves. The boosting/BLE system asymptotes relatively quickly with approximately twenty weak learners in the left panel. This is because fewer BLE beacons are visible from each location due to their positioning, lower power, and shielding. The lower panel shows that the classifiers use more of the higher

[4] RSSI is measured on a logarithmic scale.

powered BLE beacons in the second data set. Notice that the increased range of the BLE beacons degrades the performance of the map-based MAX/BLE method relative to the first data set. This is because on average 9.86 (std = 3.62) beacons are visible in each scan in the second data set compared to only 2.95 (std = 1.32) beacons in the first data set.

In contrast, the boosting/Wi-Fi system shows more steady improvement with the addition of weak learners. This reflects the fact that many more Wi-Fi BSSIDs may be in range of each location, and in turn, many more Wi-Fi-based features may be useful for classification. The fusion system also shows some gradual improvement and reaches peak accuracy with about thirty weak learners. The horizontal dashed lines show the performance for the Redpin and MAX baselines.

For the first data set, the Redpin/BLE system performs at a higher accuracy than the boosting/BLE classifier, while on the second data set they perform at the same level. For both data sets, the Redpin/BLE system shows excellent performance and is superior to the Wi-Fi-only variant. This trend is also evident in the boosting results, though with the inclusion of a large number of weak learners, the BLE-only and Wi-Fi-only performance converges to a similar level. These experiments show that BLE beacons offer a competitive alternative to Wi-Fi for room-level location identification.

In both experiments, the boosting system that combines BLE and Wi-Fi data shows the highest accuracy. For fusing the RSSI data, we compute margin features from each modality independently and combine them for classifier training. Figure 8 visualizes the order of selection and

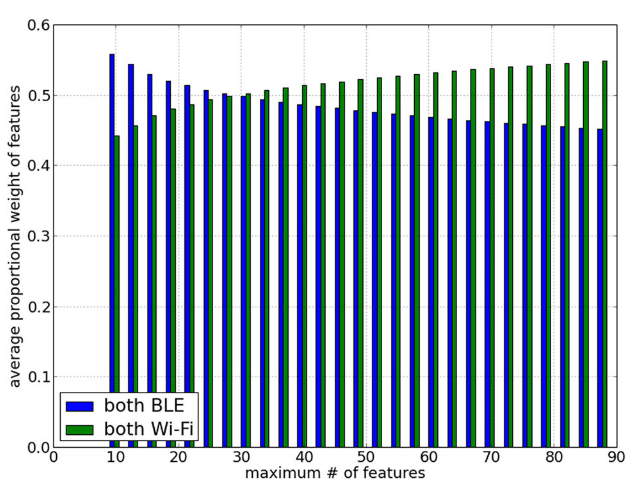


Fig. 8 The plots show the relative weighting of BLE-based and Wi-Fi-based features in the combined boosting classifiers on the first (*top*) and second (*bottom*) experimental data sets. The weights are first normalized per room, and the proportions allocated to BLE and Wi-Fi are averaged per room and per fold to generate the plots

relative proportion in which the two modalities are used by the boosting classifiers on the second larger data set. Specifically, it shows the average normalized weights (α_m in (2)) associated with BLE margin features and Wi-Fi margin features. The horizontal axis represents the maximum number of decision stumps allowed in each classifier. The weights are first summed per location and used to normalize the proportion allocated to each modality. These results are then averaged over all locations and cross-validation folds. The plots show that BLE-based features are greedily selected first, but because of the abundance of Wi-Fi data and its larger range, more and more Wi-Fi features are added to the classifiers in subsequent iterations. This suggests that the BLE information is more discriminative than Wi-Fi as expected, since it is deployed explicitly for location classification. However, as training progresses, the relative density and longer range of Wi-Fi provide complementary information.

4.2 Spatial granularity

In additional experiments, we compared these methods for location classification at a finer spatial granularity. One portion of our office includes nine cubicles of size 2.5 by 2.5 m arranged in a three-by-three grid. The area is separated from a row of five conventional (walled) offices by a hallway 1.5 m wide, and from a larger conference room by a hall. The area is enclosed in a green rectangle in Fig. 5. In this area, we performed location classification of the nine cubicles and six other rooms based on BLE, Wi-Fi, and their combination. As before, there is a BLE beacon located on the ceiling in each location, and various Wi-Fi BSSIDs visible both from our internal network and from neighboring organizations. We sampled a data set comprised of 445 scans as part of the larger second data set studied above. The scans include RSSI data from 17 BLE beacons in the building and 163 unique Wi-Fi BSSIDs.

Experimental results for all previously evaluated baseline and boosting variants appear in Table 3. The results for

Table 3 Classification accuracy results for the open space data set with 15 locations and 445 scans

Method	RSSI data	Accuracy
MAX	BLE	0.504
Redpin [6]	BLE	0.818
	Wi-Fi	0.876
	BLE + Wi-Fi	0.907
Boosting	BLE	0.818
	Wi-Fi	0.925
	BLE + Wi-Fi	**0.955**

Accuracy is computed using sixfold cross-validation and 45 iterations of boosting classifier training

BLE alone in this setting show a decline from the earlier experiments, suggesting that there are limitations to their spatial resolution for location classification. In particular, relying on the maximal observed BLE RSSI shows poor performance. On average, each scan contains RSSI measurements from 12.54 BLE beacons (std = 2.45). The boosting classifiers that use Wi-Fi alone, or in combination with BLE show improved performance relative to Redpin in this setting. The boosting classification method with the complete set of measurements shows the best performance at 95 % accuracy for this higher granularity. Thus, the boosting classifiers again offer both an efficiency advantage and superior accuracy to the Redpin results using Wi-Fi only or together with BLE at this higher spatial granularity.

4.3 Wi-Fi access point density

A key to the relative efficiency of boosting in this application is the feature selection inherent in its classifier training. The increase in the number of visible hardware IDs moving from the first to the second data set produced a significant complexity increase in the Redpin systems. On the other hand, because the boosting system selects a small subset of margin features and corresponding hardware IDs for each room classifier, the complexity remained stable and low for both data sets.

We performed additional experiments to investigate the impact of Wi-Fi density on the performance of these systems. For this, we made two changes to our experimental protocol. First, we used the second experimental data set, but restricted the Wi-Fi information to access points from our organization (152 of 254 total BSSIDs). Secondly, we performed repeated experiments using sixfold cross-validation by randomly sampling a proportion of the *access points* and withholding all their associated RSSI measurements during classifier training.[5] Here the error bars are computed over each fold and iteration. We graphically depict the results in Fig. 9. To create each point in the plots, we perform four iterations of random selection of access points to simulate lower Wi-Fi density. The plot shows accuracy versus the number of Wi-Fi BSSIDs available. Generally, the Redpin system outperforms boosting here by a small margin, but boosting is competitive over a wide range of Wi-Fi densities. The fusion results show that the boosting system does very slightly better than Redpin over the bulk of the experimental conditions, and both use BLE to provide high performance in relatively low-density Wi-Fi environments. As before, the boosting system operates with substantially reduced computation.

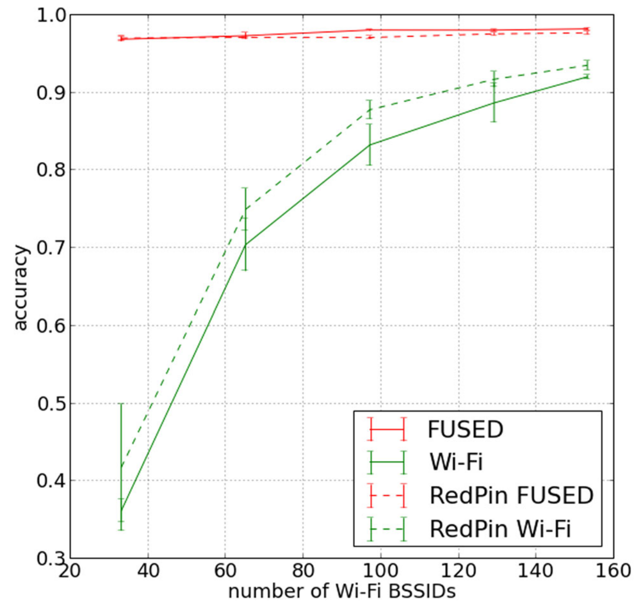


Fig. 9 The plot shows the accuracy of Wi-Fi-only and combined classification using Redpin and boosting as the number of visible Wi-Fi hardware IDs varies (*x*-axis)

4.4 Distance-based error analysis

We also characterize misclassification mistakes made using both BLE and Wi-Fi with the boosting classification approach. For these experiments, we first associate a specific distance-based cost to each possible misclassification, in contrast to the simple binary cost that defines the accuracy measure we have used to this point. For this, we compute distances between the centroids of each pair of locations in the second data set. Note that the analysis here is not directly comparable to distance-based error for localization systems because we still are using room-level ground truth rather than absolute position.

As mentioned earlier, because the BLE beacons are explicitly deployed for location classification, we expect that the mistakes BLE-based methods make will most often be other nearby locations relative to Wi-Fi. The results in Fig. 10 affirm this intuition over a range of Wi-Fi densities simulated as before. When only a few Wi-Fi base stations are included, the variance shown by the error bars is higher. This is because the access points only effectively cover some subset of locations. As more and more base stations are included, the error variance and average error decrease to a point comparable to and eventually lower than the BLE result indicated by the single point in the plots.

Here, we again average the distance-based cost of all misclassifications averaged over the entire data set, again

[5] Multiple BSSIDs are associated with a single Wi-Fi access point. Here, we include and withhold BSSIDs by access point.

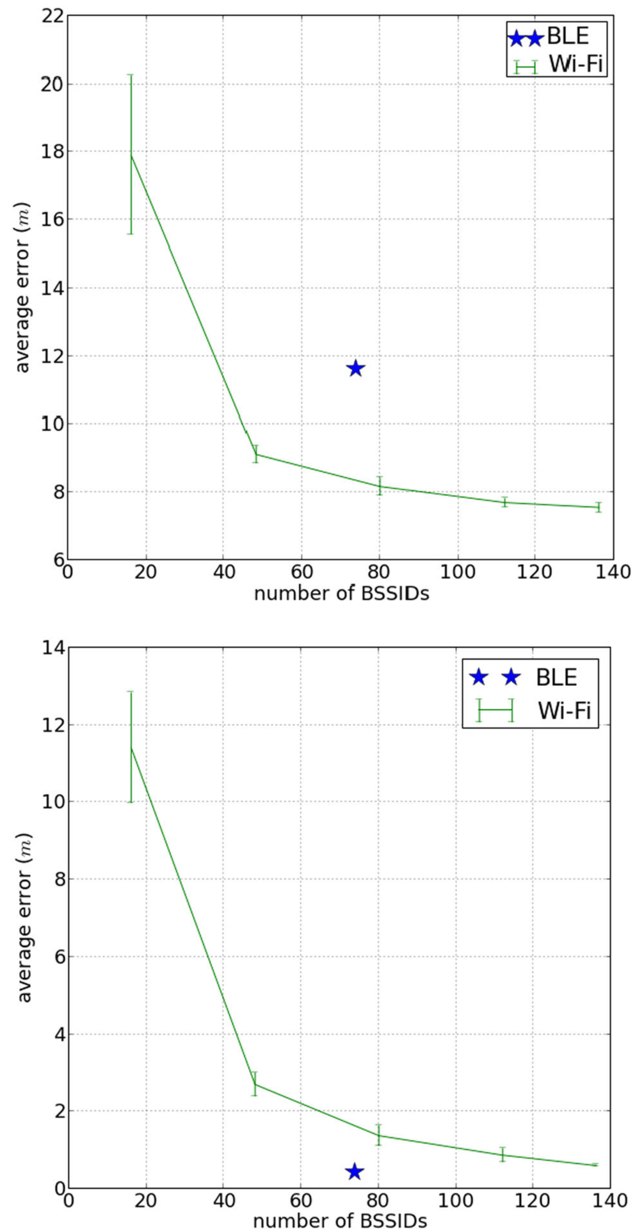


Fig. 10 The plots show the average misclassification distance in meters for Wi-Fi as the number of access points and their associated BSSIDs in the available RSSI data is varied. The single point represents the BLE performance. The *upper plot* shows the average error for misclassified scans only. The *lower panel* shows the average error results for all scans, including correctly classified results

using sixfold cross-validation and four iterations at each Wi-Fi AP density. We examine the distance-based error in two plots. In the top plot, we compute the average distance over only misclassified samples. In this case, the *number* of misclassifications does not influence the results, and BLE shows higher error (11.63 m) relative to Wi-Fi (approximately 9 m) at the same relative density. In the lower panel, we average the error over all scans, including correctly classified samples, and see that in this case BLE shows a lower average error, because it makes fewer errors than Wi-Fi. As the Wi-Fi access point density increases, the distance-based error converges to a level (average error of 0.555 m) close to that of the BLE-only boosting system (average error of 0.405 m). The fusion system that uses BLE with Wi-Fi reduces this average error to 0.1172 meters, again demonstrating the complementary nature of these modalities for indoor location classification.

4.5 Experimental summary

These experiments validate our approach to location classification. First, the accuracy of the boosted classifiers shows a high level of performance using BLE and Wi-Fi both individually and in combination. We show that the explicit deployment of BLE hardware for indoor location classification is effective and competitive with Wi-Fi-based methods using both Redpin and boosting. Distance-based analysis of the BLE-only classifier's mistakes also shows a lower average cost relative to Wi-Fi. Second, the boosting approach shows consistent performance gains by combining BLE with Wi-Fi information to demonstrate that BLE beacons can be deployed to complement existing Wi-Fi infrastructure for high-performance location classification. Additionally, analysis of the boosting classifiers and their composition and weighting of weak learners further affirms the complementary nature of the BLE and Wi-Fi RSSI measurements. Finally, the error analysis above using distance validates the improved accuracy results offered by combining BLE and Wi-Fi. There are both fewer errors in terms of accuracy, and also lower cost errors in terms of distance, made by the combined classifier relative to any single modality result.

Importantly, the boosted classifiers off-load the bulk of the computational complexity to an off-line training process. The test complexity of the boosted classifiers is consistently orders of magnitude faster than the Redpin results. While more effort could be made to optimize both the kNN and boosting code, these results demonstrate the significant computational savings provided by supervised classification relative to the matching approaches prevalent in prior work.

5 Discussion

The LoCo Framework, along with our analysis of its performance, demonstrates that it is possible to perform accurate room-based location classification using practical, accessible methods. While our work is not the first to investigate leveraging multiple sources of radio signals to determine device location, we do believe we are the first to demonstrate the efficacy of a hybrid radio approach with locating that achieves the level of accuracy and performance demonstrated by our experiments. This is a significant step forward in providing a usable platform for creating location applications and services for settings such as the workplace.

We believe LoCo can have immediate utility in creating next generation location applications and services. A particular motivation for us is to drive improvements in office communication tools and technologies. For instance, providing real-time location information of colleagues in chat, email, and other communication applications is now easily implementable with LoCo. We have leveraged LoCo to enhance an existing office presence tool [38], and we are now planning a study to understand how improved location information translates to meaningful changes in communication behavior. Even further, LoCo could be leveraged to build important office safety systems. For example, a system that can inform building security personnel if employees are present in the building after hours. Similarly, in the event of a fire or earthquake, the last known location of workers could be provided to first responders.

Additionally, we believe one of the most powerful uses of the framework is as a platform for performing long-term studies of workplace interaction and behavior. Since the framework only requires the user to install an application on his or her smartphone, the burden and overhead on the users are low. Unlike technologies such as the Sociometric Badges [25], no specialized devices need to be worn or carried by workers; neither is there any maintenance of such devices—a user simply must carry a device she likely already carries with her during day-to-day activities.

There are many additional applications for indoor localization. There have been various integrations of indoor localization with audio guides in exhibition spaces or museums [15, 34, 36]. Further applications include indoor location-based gaming [10, 33], location-aware shopping assistance [12, 42], location-based advertising [1], and indoor presence awareness [2, 37, 38].

While there are advantages introduced by our framework, it is important to note its limitations. A particularly burdensome limitation is that our approach requires that a set of training data be collected at each location. This process is simple but involves taking a device into room and having it perform several scans that it submits to the classification engine with an attached ground truth label. Further, as environments change (e.g., a Wi-Fi AP is moved or replaced), a new survey is likely necessary. There are, however, several ways to address these issues. One approach, as first proposed by [6], is to crowd source users in providing ground truth. For instance, when a user notices a location is not classified, or not classified correctly, he could simply tell the client application the correct location and the particular scan could be sent to the classification engine as an additional piece of data used in the training process. Another approach, which we are actively investigating, is to use a small robot to perform a survey. With this method, a new, up-to-date model can be created on a regular, ongoing basis with relatively low human overhead.

Our location tracking approach is by design limited in spatial precision. That is, our framework is not designed to determine the precise location within a room a device is located. In large rooms, this would limit the utility of leveraging LoCo to classify activities of individuals or groups, or to actuate changes or events within the physical space. Our belief is that achieving these goals will likely be done through fusing LoCo location information with other sensor sources. For instance, high-precision within-room location (error of only a few centimeters) is possible with acoustic beacons [29]. We are actively working on fusing various methods of precise within-room techniques into the LoCo Framework.

6 Conclusion and future work

Location is playing an ever-increasing role in mobile computing. Many of the most popular applications and services used today exploit knowledge of the user's current location. For many of these applications, the accuracy of current technologies is adequate. However, we believe the next generation of mobile services will demand accurate, reliable location information in indoor environments. In this paper, we presented the LoCo framework. We discussed how it has been designed and developed to provide application and service developers access to indoor location information. We demonstrated an evaluation of the system and explained its performance and capabilities compared to existing state-of-the-art techniques. Further, we performed this evaluation using hardware and software that is commercially available today.

Our work to date serves as inspiration for future activities. First, we intend to deploy LoCo in different environments to understand its performance characteristics more broadly. This effort will also allow us to better investigate how well LoCo scales to larger configurations, for instance, within a multistory high rise in a densely urban environment. Second, we intend to leverage LoCo to prototype new and novel applications that use LoCo's accurate location information to provide interesting and compelling experiences for users in office and retail settings. Finally, so that we can all explore the future of location-driven mobile computing, we will work to make LoCo available to others in the research community.

References

1. Aalto L, Göthlin N, Korhonen J, Ojala T (2004) Bluetooth and WAP push based location-aware mobile advertising system. In: Proceedings of the 2nd international conference on mobile systems, applications, and services, ACM, pp 49–58
2. An X, Wang J, Prasad RV, Niemegeers I (2006) OPT: online person tracking system for context-awareness in wireless personal network. In: Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality, ACM, pp 47–54
3. Andoni A, Indyk P (2006) Efficient algorithms for substring near neighbor problem. In: Proceedings of the seventeenth annual ACM-SIAM symposium on discrete algorithm, SODA '06, ACM, New York, NY, USA, pp 1203–1212. doi:10.1145/1109557.1109690
4. Aparicio S, Perez J, Bernardos A, Casar J (2008) A fusion method based on Bluetooth and WLAN technologies for indoor location. In: IEEE international conference on multisensor fusion and integration for intelligent systems, 2008. MFI 2008, pp 487–491
5. Biehl JT, Cooper M, Filby G, Kratz S (2014) Loco: a ready-to-deploy framework for efficient room localization using Wi-Fi. In: Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing, UbiComp '14, ACM, New York, NY, USA, pp 183–187. doi:10.1145/2632048.2636083
6. Bolliger P (2008) Redpin-adaptive, zero-configuration indoor localization through user collaboration. In: Proceedings of the first ACM international workshop on mobile entity localization and tracking in GPS-less environments, MELT '08, ACM, New York, NY, USA, pp 55–60. doi:10.1145/1410012.1410025
7. Chintalapudi K, Padmanabha Iyer A, Padmanabhan VN (2010) Indoor localization without the pain. In: Proceedings of the sixteenth annual international conference on mobile computing and networking, MobiCom '10, ACM, New York, NY, USA, pp 173–184. doi:10.1145/1859995.1860016
8. Cong L, Zhuang W (2002) Hybrid tdoa/aoa mobile user location for wideband cdma cellular systems. IEEE Trans Wirel Commun 1(3):439–447. doi:10.1109/TWC.2002.800542
9. Dempster A, Li B, Quader I (2008) Errors in deterministic wireless fingerprinting systems for localization. In: ISWPC 2008, pp 111–115
10. Ferdinand P, Müller S, Ritschel T, Wechselberger U (2005) The eduventure—a new approach of digital game based learning combining virtual and mobile augmented reality games episodes. In: Pre-conference workshop? Game based Learning? of DeLFI 2005 and GMW 2005 conference, vol. 13. Rostock
11. ...d Y, Schapire R (1997) A decision-theoretic generalization ...line learning and an application to boosting. J Comput Syst Sci 55(1):119–139. doi:10.1006/jcss.1997.1504
12. Gross HM, Böhme HJ, Schröter C, Müller S, König A, Martin C, Merten M, Bley A (2008) Shopbot: progress in developing an interactive mobile shopping assistant for everyday use. In: IEEE international conference on systems, man and cybernetics, 2008. SMC 2008. pp 3471–3478
13. Harter A, Hopper A, Steggles P, Ward A, Webster P (2002) The anatomy of a context-aware application. Wirel Netw 8(2/3):187–197
14. ...e T, Tibshirani R, Friedman J (2001) The elements of sta-...l learning. Springer, New York
15. Heller F (2008) Corona: realizing an interactive experience in visually untouchable rooms using continuous virtual audio spaces. Master's thesis, RWTH Aachen University
16. Hightower J, Want R, Borriello G (2000) Spoton: An indoor 3D location sensing technology based on RF signal strength. UW CSE 00–02-02, University of Washington, Department of Computer Science and Engineering, Seattle, WA 1
17. Li P (2010) Robust logitboost and adaptive base class (abc) logitboost. In: UAI 2010, proceedings of the twenty-sixth conference on uncertainty in artificial intelligence, pp 302–311. arXiv:1203.3491
18. Lin H, Zhang Y, Griss M, Landa I (2009) Enhanced indoor locationing in a congested Wi-Fi environment. Tech. Rep. MRC-TR-2009-04, Carnegie Mellon Silicon Valley
19. Liu H, Darabi H, Banerjee P, Liu J (2007) Survey of wireless indoor positioning techniques and systems. IEEE Trans Syst Man Cybern Part C Appl Rev 37(6):1067–1080
20. Madigan D, Elnahrawy E, Martin RP, Ju WH, Krishnan P, Krishnakumar AS (2005) Bayesian indoor positioning systems. In: Proceedings of the 24th joint conference of the IEEE Computer and Communication Societies (INFOCOM 2005)
21. Martin E, Vinyals O, Friedland G, Bajcsy R (2010) Precise indoor localization using smart phones. In: Proceedings of the international conference on Multimedia, MM '10, ACM, New York, NY, USA, pp 787–790. doi:10.1145/1873951.1874078
22. Mukherjee I, Schapire R (2010) A theory of multiclass boosting. In: Lafferty J, Williams CKI, Shawe-Taylor J, Zemel R, Culotta A (eds) Advances in Neural Information Processing Systems, vol. 23., pp 1714–1722
23. Ni LM, Liu Y, Lau YC, Patil AP (2004) Landmarc: indoor location sensing using active RFID. Wirel Netw 10(6):701–710
24. Niculescu D, Nath B (2003) Ad hoc positioning system (APS) using AOA. In: INFOCOM 2003. Twenty-second annual joint conference of the IEEE computer and communications. IEEE Societies, vol. 3., pp 1734–1743. doi:10.1109/INFCOM.2003.1209196
25. Olguín DO, Waber BN, Kim T, Mohan A, Ara K, Pentland A (2009) Sensible organizations: technology and methodology for automatically measuring organizational behavior. Trans. Syst. Man Cybern. Part B 39(1):43–55. doi:10.1109/TSMCB.2008.2006638
26. Ouyang RW, Wong AKS, Lea CT, Chiang M (2011) Indoor location estimation with reduced calibration exploiting unlabeled data via hybrid generative/discriminative learning. In: IEEE transactions on mobile computing, vol. 99. (PrePrints). doi:10.1109/TMC.2011.193
27. Pan JJ, Pan SJ, Yin J, Ni LM, Yang Q (2012) Tracking mobile users in wireless networks via semi-supervised colocalization. In: IEEE transactions on pattern analysis and machine intelligence, pp 587–600
28. Priyantha NB, Chakraborty A, Balakrishnan H (2000) The cricket location-support system. In: Proceedings of the 6th annual international conference on Mobile computing and networking, ACM, pp 32–43
29. Rishabh I, Kimber D, Adcock J (2012) Indoor localization using controlled ambient sounds. In: International conference on indoor positioning and indoor navigation (IPIN), pp 1–10. doi:10.1109/IPIN.2012.6418905
30. Rodrigues ML, Vieira LFM, Campos MF (2011) Fingerprinting-based radio localization in indoor environments using multiple wireless technologies. In: IEEE 22nd international symposium on ...nal indoor and mobile radio communications (PIMRC), ...03–1207
31. Roxin A, Gaber J, Wack M, Nait-Sidi-Moh A (2007) Survey of wireless geolocation techniques. In: IEEE Globecom
32. Sen S, Lee J, Kim KH, Congdon P (2013) Avoiding multipath to revive inbuilding wifi localization. In: Proceeding of the 11th annual international conference on mobile systems, applications, and services, MobiSys '13, ACM, New York, NY, USA, pp 249–262. doi:10.1145/2462456.2464463

33. Sukthankar G (2002) The dynadoom visualization agent: A handheld interface for live action gaming. In: Workshop on ubiquitous agents on embedded, wearable, and mobile devices (conference on intelligent agents and multiagent systems), Bologna, Italy
34. Terrenghi L, Zimmermann A (2004) Tailored audio augmented environments for museums. In: Proceedings of the 9th international conference on Intelligent user interfaces, ACM, pp 334–336
35. Tod beacons (2014) http://www.todhq.com/
36. Wakkary R, Newby K, Hatala M, Evernden D, Droumeva M (2004) Interactive audio content: an approach to audio content for a dynamic museum experience through augmented audio reality and adaptive information retrieval. In: Museums and the web conference
37. Want R, Hopper A, Falcão V, Gibbons J (1992) The active badge location system. ACM Trans Inf Syst 10(1):91–102
38. Wiese J, Biehl JT, Turner T, van Melle W, Girgensohn A (2011) Beyond 'yesterday's tomorrow': towards the design of awareness technologies for the contemporary worker. In: Proceedings of the 13th international conference on human computer interaction with mobile devices and services, MobileHCI '11, ACM, New York, NY, USA, pp 455–464. doi:10.1145/2037373.2037441
39. Xiong J, Jamieson K (2012) Towards fine-grained radio-based indoor location. In: Proceedings of the twelfth workshop on mobile computing systems and applications, HotMobile '12, ACM, New York, NY, USA, pp 13:1–13:6. doi:10.1145/2162081.2162100
40. Youssef M, Agrawala A (2005) The horus wlan location determination system. In: Proceedings of the 3rd international conference on Mobile systems, applications, and services (MobiSys '05), pp 205–218
41. Zhu J, Zou H, Rosset S, Hastie T (2009) Multi-class adaboost. Stat Interface 2:349–360
42. Zhu W, Owen CB, Li H, Lee JH (2004) Personalized in-store e-commerce with the promopad: an augmented reality shopping assistant. Electron J E-commer Tools Appl 1(3):1–19