Predicting Location Semantics Combining Active and Passive Sensing with Environment-independent Classifier

Masaya Tachikawa, Takuya Maekawa, and Yasuyuki Matsushita
Graduate School of Information Science and Technology, Osaka University
{tachikawa.masaya,maekawa,yasumat}@ist.osaka-u.ac.jp

ABSTRACT

This paper presents a method for estimating a user's indoor location without using training data collected by the user in his/her environment. Specifically, we attempt to predict the user's location semantics, *i.e.,* location classes such as restroom and meeting room. While indoor location information can be used in many real-world services, *e.g.,* context-aware systems, lifelogging, and monitoring the elderly, estimating the location information requires training data collected in an environment of interest. In this study, we combine passive sensing and active sound probing to capture and learn inherent sensor data features for each location class using labeled training data collected in other environments. In addition, this study modifies the random forest algorithm to effectively extract inherent sensor data features for each location class. Our evaluation showed that our method achieved about 85% accuracy without using training data collected in test environments.

ACM Classification Keywords: H.3.4 Information storage and retrieval: Systems and software.

Author Keywords: Indoor positioning; passive sensing; active probing

INTRODUCTION

Due to the recent advances in sensing technologies, several wearable lifelogging devices such as Narrative Clip and GoPro are now commercially available. Also, commercial smart devices including smartphones, smart watches, and smart glasses are already equipped with various sensors, and these devices are used to collect data from our daily life. Using daily-life sensor data collected by such devices, context recognition methods such as activity recognition and indoor positioning have been actively studied in the ubicomp research community. Activity recognition studies employ body-worn sensors including acceleration sensors, gyroscopes, and microphones to recognize daily activities such as walking, running, and house cleaning [5, 27, 28, 30, 29]. Indoor positioning studies rely on signaling technologies, for example, infrared [48], ultrasound [33], active sound probing [12, 45], Bluetooth [47], and Wi-Fi [25, 42]. The recognized context information can be used in real-world services, *e.g.,* context-aware systems, lifelogging, and surveillance of the elderly [18, 31]. In addition, the information is used to label lifelog data such as egocentric videos recorded by wearable cameras [7].

Many of the existing context recognition systems rely on supervised machine learning techniques and assume that training data are collected by a user in his/her daily environment. However, collecting and labeling sensor data by average persons is difficult and impractical. In this study, we propose a method for estimating a user's indoor location without using training data collected by the user. Specifically, we attempt to predict the user's location semantics, *i.e.,* location classes such as restroom and meeting room without using training data collected by the user. That is, our goal is to estimate a *type* of geographic location rather than estimate a specific location amongst a defined set of locations.

Existing data mining studies have tried estimating location semantics such as workplace, cafe, and home using GPS and GSM trajectory data [19, 26, 52]. The estimated location semantics can be used for recommending travel routes and shops, and understanding daily activities. In contrast, this study attempts to estimate room-level location semantics, which is also useful for understanding a user's daily life because the user's room-level location strongly relates to the user's activity. When a user is estimated to be located in a restroom, for example, we can easily estimate that the user is using the toilet. The estimated location semantics can also be used to label lifelog data. Furthermore, the location semantics can be used to adaptively control lifelogging devices, *e.g.,* turning off a wearable camera such as Narrative Clip when a user enters a restroom.

To estimate location semantics, we attempt to learn inherent sensor data features for each location class such as restroom and meeting room. In this study, we combine passive sensing and active probing to capture and learn inherent sensor data features for each location class using labeled training data collected in other environments. This approach enables us to predict a user's location semantics without using labeled training data collected by the user. In this study, we passively capture environmental features of each location class using sensors such as magnetometers and barometers. As for active probing, we probe the environment by emitting a sound chirp and then analyze the impulse response (IR). The active sound probing permits us to capture features of an environment such

UbiComp '16, September 12-16, 2016, Heidelberg, Germany

© 2016 ACM. ISBN 978-1-4503-4461-6/16/09…$15.00

DOI: **http://dx.doi.org/10.1145/2971648.2971684**

as the shape, dimension, ability to absorb sound, construction materials, and objects inside the environment. Because the active sound probing requires sound emission by a speaker, we assume that a wearable device including a smart watch or wearable camera device is attached to or clipped on the body. In our experiment, we attach a smartphone around the neck as shown in Fig. 1. While our experiment uses a smartphone to collect data, we believe that wearable devices such as smart glasses and smart watches are suitable for our purpose since smartphones are usually in a pants pocket or bag.

In addition, this study modifies the random forest algorithm [8] to effectively extract inherent sensor data features for each location class. The original random forest algorithm constructs trees focusing only on classification performance of training instances. In contrast, we attempt to find sensor data features common to multiple environments, which will be the inherent sensor data features for each location class.

In this study, we focus on laboratory/office environments and attempt to estimate location semantics, *i.e.,* a location class. When a user is in a room, for example, our method predicts a location class of the user's location such as meeting room, cafeteria, restroom, or office (desk). The estimation results can be useful in applications such as work management, automated daily work journaling, and maintenance of a work/break balance.

We briefly explain the procedures of our method using Fig. 2. The proposed method processes and analyzes daily life sensor data collected from a user for a long duration, *e.g.,* one-day data. The method first detects places where the user stayed for a long time using acceleration data (hereinafter simply called places). After detecting places, we cluster the places based on similarities in Wi-Fi signals collected at the places. With this clustering, detected places corresponding to the same room are grouped into one cluster. We then estimate a location class to which each cluster belongs using active probing and passive sensing.

The above mentioned clustering using Wi-Fi signals is not necessarily required to estimate a location class of a user's location. However, this clustering approach permits us to associate each cluster with its Wi-Fi signature, *i.e.,* Wi-Fi fingerprint, observed at the cluster location in advance. Therefore, after the association and semantic estimation of the clusters, we can estimate the user's current location semantics using only Wi-Fi data collected by the user's smartphone without performing active probing.

The contributions of this study are described as follows.
(1) To the best of our knowledge, this is the first study that predicts a user's room-level location semantics combining active proving and passive sensing without using training data collected by the user in his/her environment. (2) We modify the random forest algorithm to extract inherent sensor data features for each location class. (3) We evaluate our method using sensor data collected in four different real office/laboratory environments.

In the rest of this paper, we first introduce work related to indoor positioning. Then, we describe the design of our method

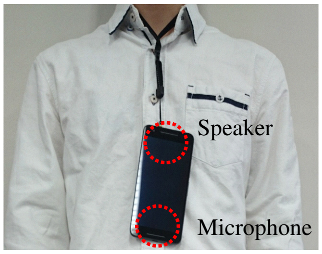


Figure 1. Smartphone attachment for performing active probing and passive sensing (Google Nexus 6P)

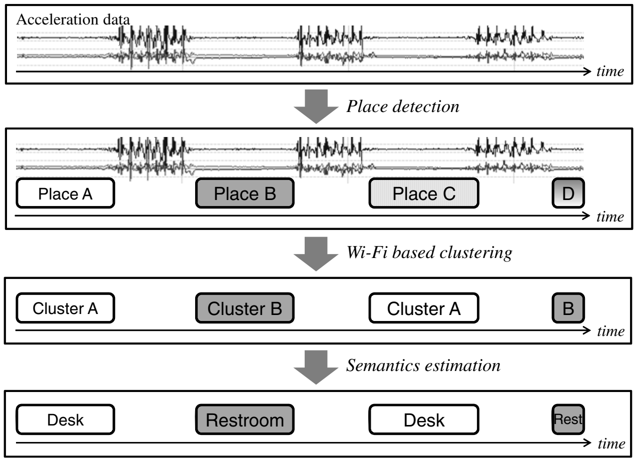


Figure 2. Procedures of estimating indoor location semantics

for estimating indoor location semantics. In the evaluation, we test our method using sensor data obtained in real environments.

RELATED WORK

Wi-Fi indoor positioning with no/few training data

Here we introduce studies that attempt to reduce the burdens related to constructing a Wi-Fi fingerprint database. Jiang *et al.* [20] attempt to learn a fingerprint for each room automatically by clustering Wi-Fi scan data observed in a user's daily life with the help of acceleration sensors. Pulkkinen *et al.* [34] employed semi-supervised manifold learning to obtain dense labeled fingerprints from partially labeled fingerprints. The authors constructed a non-linear projection that maps high-dimensional signal fingerprints onto a two-dimensional manifold. Several studies construct radio maps with no supervision by using simultaneous localization and mapping (SLAM) techniques [15, 37]. Hardegger *et al.* [17] perform SLAM based on the fact that certain daily activities are performed at particular places (*e.g.,* sleeping in a bedroom). Rai *et al.* [36] also attempt to automatically construct radio maps with the pedestrian dead reckoning (PDR) technique. Similarly, Taniuchi *et al.* [43] collect fingerprints with PDR techniques for automatic radio map update.

Indoor positioning with active sound probing

In several mobile computing studies [53, 10], sound beaconing has been employed to estimate relative positions to other devices. Kunze *et al.* [23] propose an absolute positioning method that estimates where a phone is placed combining vibration and short, narrow frequency beeps to sample the response of an environment.

Rossi *et al.* [38] propose an indoor positioning method using smartphones based on active sound fingerprinting. The authors measure impulse response at each indoor position and train a classifier that predicts a user's current position using the observed impulse response. In contrast, our study attempts to predict a location class by extracting inherent sensor data features for each location class combining active probing and passive sensing. Tung *et al.* [45] also employ active sound probing for indoor location tagging and achieve 1cm resolution. Similar to our study, Fan *et al.* [12] attempt to estimate a location class using active sound probing. However, the study focuses only on a restroom class to adaptively turn on/off a wearable camera to preserve privacy of a user. In contrast, we tackle multi-class classification in office/laboratory environments combining active probing and passive sensing, and design a classifier that can capture inherent sensor data features for each location class. Also, we employ synchronous averaging techniques for active sound probing to cancel out environmental noises, which are not investigated in the above studies.

Indoor positioning with other sensors

Passive sound fingerprinting is usually used to locate a user or understand a user's location context. Tarzia *et al.* [44] extract sound fingerprints based on acoustic background spectrum of rooms to locate a smartphone user. Similar to Wi-Fi fingerprinting, collected fingerprints are stored in a fingerprint database in advance. In addition to acoustic features, Azizyan *et al.* [3] employ acceleration, image, Wi-Fi, and light features obtained from smartphone sensors to estimate logical location labels of stores such as Wal-Mart and Starbucks. These passive sound based methods require training data collected in environments of interest.

Several studies employ a magnetometer in a smartphone to locate the smartphone user by employing magnetic fingerprints because indoor environments have objects that show high magnetic field values such as pillars and electrical appliances [46, 16]. Barometers are also used to understand a user's indoor position and trajectory, *e.g.,* floor level estimation and detection of climbing and descending stairs [4, 2].

In addition to the above sensors, a light sensor, camera, and NFC reader have been used to locate a user and understand a user's location context [32, 51, 54]. However, the camera-based method may generate privacy concerns.

Bao *et al.* [6] attempt to estimate room-level location semantics using features such as average duration of stay, maximum duration of stay, and frequency of visits. In contrast, we attempt to capture inherent sensor data features for each location class.

PROPOSED METHOD

Overview

Figure 3 shows an overview of our method. Our proposed method mainly consists of two phases; training and test phases. In the training phase, training data are collected in several environments and a classifier for estimating a location class is trained. In the test phase, a user collects unlabeled sensor data using a sensor device such as a smartphone in his/her environment. Our method first detects places where the user stayed for a long time using acceleration data. After that, the method clusters the detected places based on similarities in Wi-Fi signals collected at the places. The method then extracts features from sensor data collected at each location cluster by means of active probing and passive sensing. The method finally estimates a location class to which each cluster belongs using the extracted features and the classifier trained on sensor data from other environments. We explain the procedures of our method in detail.

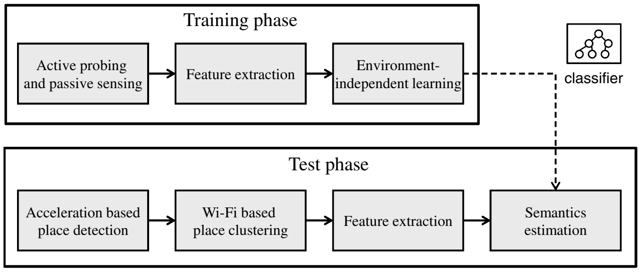

Training phase
Active probing and passive sensing
Feature extraction
Environment-independent learning
classifier
Test phase
Acceleration based place detection
Wi-Fi based place clustering
Feature extraction
Semantics estimation

Figure 3. Overview of our proposed method

Table 1. Smartphone sensors used in this study

sensor	sampling rate	use
accelerometer	30 Hz	place detection
Wi-Fi module	0.1 Hz	place clustering
magnetometer	30 Hz	semantic estimation
barometer	30 Hz	semantic estimation
microphone	44.1 kHz	semantic estimation

Sensor data collection

In the training phase, sensor data are manually collected using active probing and passive sensing at various places whose location classes are known. In this study, we passively collect magnetic sensor data, barometric pressure data, and sound data, which are usually used in existing indoor positioning studies as mentioned in the related work section. As for active probing, we measure impulse responses.

In the test phase, we assume that a user's sensor device such as smartphone always collects acceleration data. When the user is estimated to be staying still, the smartphone automatically collects Wi-Fi signals and sensor data using active probing and passive sensing.

Table 1 summarizes sensors used in our study.

Acceleration based place detection

Our method processes and analyzes daily life sensor data collected from a user for a long duration. As shown in the upper portion of Fig. 2, we first find places where the user stayed for a long time using acceleration data. When acceleration data from the user's smartphone are stable, we simply assume that the user is stationary. To accomplish this, we compute the variance for each sliding time window, *i.e.,* moving variance, and find segments whose variance values are smaller

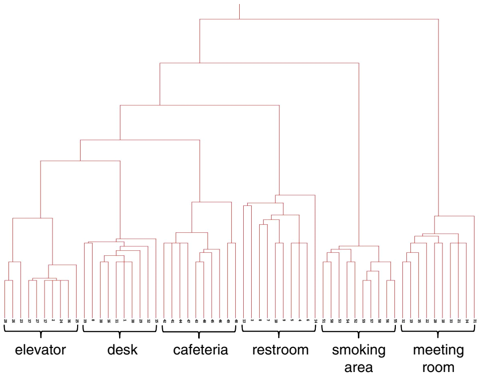


Figure 4. Dendrogram of hierarchical clustering constructed using sensor data collected in our experimental environment

than a threshold. Note that, because the acceleration sensor in a smartphone is three-axis, we compute the variance value for each axis and average them. Segments whose durations are longer than a threshold correspond to places where the user stayed for a long time.

Wi-Fi based place clustering

We cluster the detected places based on Wi-Fi RSSI signals in order to group detected places corresponding to the same room into one cluster. A Wi-Fi RSSI scan consists of pairs of the unique MAC address of an AP and the received signal strength from the AP. Because we can observe multiple scans at each place, we first average the scans to construct a representative scan of the place. Based on the average scans, we cluster the places.

In order to cluster averaged scans, we need to compute the distance between two scans. Note that, because a scan includes signal strengths from only observed APs, the numbers of APs included in different scans are different. Assume that we compute the distance between scan s_i and scan s_j. When signal from AP a_n is included only in s_i, we regard the signal strength from a_n in s_j as -100 dBm, which corresponds to the minimum signal strength value. Then, the distance between s_i and s_j is computed by

$$d(s_i, s_j) = \frac{1}{|\mathcal{A}|} \sum_{a_n \in AP} |s_i(a_n) - s_j(a_n)|,$$

where $\mathcal{A}$ is a set of APs that are included in s_i or s_j, and $s_i(a_n)$ is signal strength from a_n included in s_i.

We cluster scans (places) using agglomerative hierarchical clustering [21] based on the above Wi-Fi distances. The agglomerative hierarchical clustering is a bottom-up approach where each data point starts in its own cluster and a pair of the closest clusters is iteratively merged as one. In the agglomerative hierarchical clustering, we use Ward's minimum variance criterion [49] to compute the distance between two clusters. In this study, we merge clusters until we cannot find a pair of clusters having the distance smaller than a threshold.

Figure 4 shows a constructed dendrogram in the hierarchical clustering using sensor data collected in our experimental environment. As shown in the dendrogram, Wi-Fi signals collected in the same places are similar to each other.

Each computed cluster corresponds to a particular room and we then estimate location semantics of the cluster.

Feature extraction

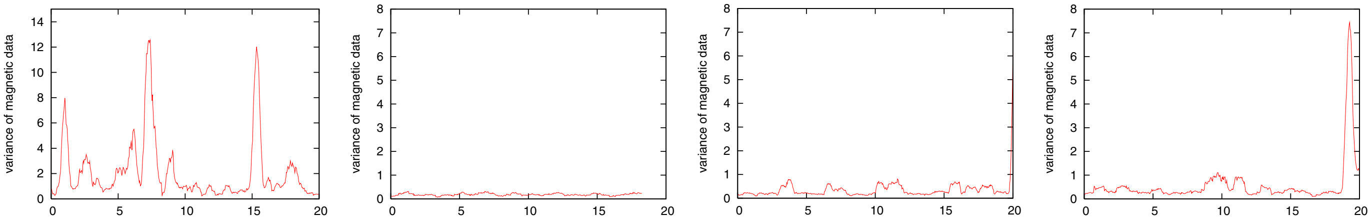
We assume that, in the training phase, sensor data are manually collected at various places whose location classes are known, *i.e.,* via a site survey. During the test phase, a user's smartphone intermittently collects sensor data (using 20-second intervals in our implementation) when the user is stationary. We extract features from this sensor data, which will be used to train a classifier and test it. We passively collect sensor data from a magnetometer, barometer, and microphone. As for active probing, we emit a sound chirp and then analyze the impulse response. Note that, when we start recording sounds for active probing, we also activate the magnetometer and barometer and collect data for one second. After finishing the sound recordings, we again activate the microphone to passively record environmental sounds. For each sensor data segment, we extract features and construct a feature vector concatenating the extracted features. We explain how we extract sensor data features in detail.

Magnetic sensor data

As mentioned in the related work section, indoor environments have objects that show high magnetic field values. For example, Fig. 5 (a) shows time series data of variances of magnetic sensor readings, *i.e.,* moving variance computed for each sliding window, collected when a person was in an elevator. Due to the motors and permanent magnets of the elevator, the sensed magnetic data fluctuated greatly. Also, in our preliminary experiment, we found that magnetic sensor data are stable in many rooms when a user stays still as shown in Fig. 5 (b). In contrast, magnetic sensor data fluctuate when a person is at outdoor or semi-outdoor places such as a smoking area as shown in Fig. 5 (c). This may be because outdoor and semi-outdoor environments are not surrounded by steel materials unlike indoor environments, and thus experience more magnetic fluctuations. Also, as shown in Fig. 5 (d), the magnetic data collected in a cafeteria also showed higher fluctuation. In such crowded environments, the magnetic sensor data are affected by other persons in the area. In order to capture these fluctuations in magnetic sensor data, we use the variance of a collected magnetic sensor data segment as a feature. Note that, because the magnetometer used in this study is three-axis, we compute the variance value for each axis and average them.

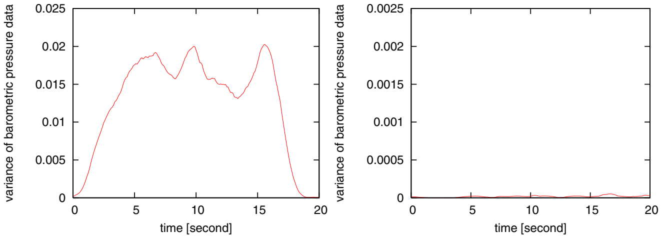
Barometric pressure data

As shown in Fig. 5 (a), magnetic sensor data collected in an elevator sometimes show large variance values. However, these values depend on product types of elevators and thus will not be inherent features for this class. Meanwhile, several smartphone-based context recognition studies employ a barometer to detect door open/close events of buildings with HVAC systems, which maintain a comfortable indoor temperature and pressure [50]. In addition to the indoor context, the barometer can be used to detect changes in altitude when



(a) elevator (b) desk (c) smoking area (d) cafeteria

Figure 5. Variances of magnetic data collected in (a) elevator, (b) desk, (c) smoking area, and (d) cafeteria



(a) elevator (b) desk

Figure 6. Variances of barometric pressure data collected in (a) elevator and (b) desk

a user is in an elevator. We believe that changes in the barometer sensor data can be observed in any elevators. Fig. 6 (a) shows time series data of variances of barometric pressure data observed in an elevator. Also, Fig. 6 (b) shows time series data of variances of barometric data observed in an office. As shown in the figure, barometric pressure data greatly fluctuate in the elevator. To capture the fluctuations, we use the variance of a collected barometric pressure sensor data segment as a feature.

Sound data

This study also employs environmental sounds passively recorded by the smartphone. With the recorded sound, we attempt to capture acoustic features that indicate a user's location such as the sound of a crowd of people or the sound of wind. In [11], the Mel-Frequency Cepstral Coefficient (MFCC) is reported to be the best transformation scheme for environmental sound recognition. Chen *et al.* [9] achieves highly accurate recognition of bathroom activities such as showering, flushing, and urination by using the MFCC features. This study also uses a 13 order MFCC of each captured sound segment windowed by a Hamming window.

Impulse response

Impulse responses permit us to measure and capture the acoustic characteristics of a space. Since an impulse is a signal that is 1 at time zero and zero otherwise, containing all frequencies in frequency domain equally, the impulse response contains all acoustic information about a space, *e.g.,* a room, between the audio source and receiver positions. Specifically, impulse responses contain time-domain acoustic information such as reflections, echoes and reverberation. Since factors of a room such as the construction materials, shape, size, and furnishings in the room affect the observed impulse responses [24, 39], the impulse responses can be a useful fingerprint of the room.

Note that the goal of this study is to capture an inherent feature for a location class. Because instances of the same location class serve the same purpose, their environmental factors are strongly correlated. For example, since a meeting room is a place for meeting, it contains tables, chairs, projector screens, and white boards. Also, a restroom contains lavatory basins, mirrors, and water resistant floors. Therefore, impulse responses measured in the same class of locations have similar fingerprints.

Since generating a very short and strong pulse, *i.e.,* an impulse, in real environments is difficult, several methods for calculating impulse responses without actually generating an impulse have been proposed [1, 41]. This study employs a sine wave sweep that reportedly tolerates non-linearity and time-variance very well, and does not require tight synchronization between the sampling clock of the signal generator and that of the digitizing unit employed for capturing the response, which means that the measurement can be conducted by a commercial smartphone [13, 14]. The mathematical definition of the sine sweep is as follows:

$$x(t) = \sin\left[\frac{\omega_1 T}{\ln\frac{\omega_2}{\omega_1}} \exp^{\frac{t}{T}\ln\frac{\omega_2}{\omega_1}} - 1\right]. \tag{1}$$

This is a sweep which starts at angular frequency ω_1 and ends at ω_2, taking T seconds. In this study, an excitation signal by a smartphone is sweeping from 20 Hz to 20 kHz, and the duration of the sweep is 0.1 seconds.

After our smartphone application starts recording sounds using the smartphone microphone, the application generates a sine wave sweep. After the sine sweep stops, the application continues to record for 0.5 seconds to capture the reverberation. The application also records a timestamp indicating the time when the sweep was emitted. We then obtain the impulse response convolving the recorded audio with the time-reversal-mirror of the emitted signal. Fig. 7 shows an example of the computed impulse response, and we can find substantial artifacts in the late part of the impulse response at higher frequencies. They are caused by slight time variances of the environment, *e.g.*, noises generated by crowd of people. To cancel out the high frequency noises, we employ synchronous averaging of a number of distinct impulse responses [14]. To do this, we generate a sine wave sweep several times

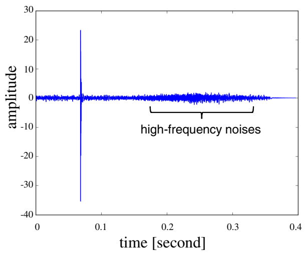


Figure 7. Example computed impulse response

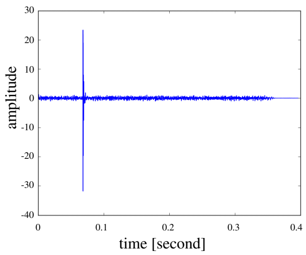


Figure 8. Example computed impulse response with synchronous averaging

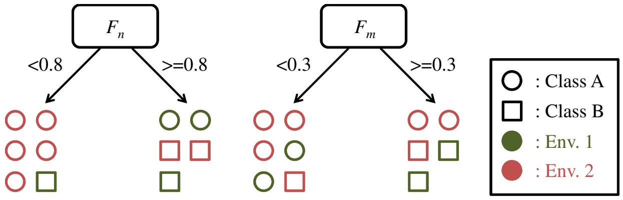


Figure 9. Examples of splits in decision tree

and average the obtained impulse responses. Our application emits sine wave sweeps at intervals of 0.75 seconds.

Note that there is a variable latency between a time when our application requests the operation system to output a sine wave sweep and the time when the OS actually plays the sweep. To compute the *synchronous* average of the impulse responses, we should detect the actual start time of the sweep. When we compare two segments of impulse responses, we find a time-shift between the two waveforms that maximizes the correlation between them. Using the detected time-shift values, we align the recorded impulse responses and then average them. Fig. 8 shows the result of the synchronous average of 16 impulse responses. As shown in the result, noises in the late part of the impulse response are reduced.

We then extract features from the averaged impulse response. Since we extract a 13-order MFCC for each sliding time window, 13 dimensional features, *i.e.,* MFCC coefficients, are computed for each window. Finally, we aggregate all the MFCC coefficients of all the windows by computing the mean value of each dimension.

Environment-independent learning

As above, we extract features from magnetic sensor data, barometric pressure data, passively recorded sound data, and impulse responses. We then construct a feature vector concatenating the extracted feature values, and classify the vector into an appropriate location class. To classify vectors collected in an environment of interest, we employ a random forest trained on labeled feature vectors collected in other environments. Because the training data contain training instances from multiple environments, the trained classifier can overfit a particular training environment. We show an example using Fig. 9.

Figure 9 shows example nodes in decision trees, and the left node splits training instances based on feature F_n and the right node splits training instances based on feature F_m. The circles and rectangles show instances belonging to class A and class B, respectively. Also, green-colored instances are collected in environment 1, and red-colored instances are collected in environment 2. In the left node, an instance whose value of F_n is smaller than 0.8 goes to the left branch; otherwise the instance goes to the right branch. In contrast, in the right node, an instance whose value of F_m is smaller than 0.3 goes to the left branch; otherwise the instance goes to the right branch. The standard decision tree learning algorithm learns the splitting rules based only on class labels, *i.e.,* class A vs class B, as mentioned in the related work section, and the information gain of the left split is identical to that of the right split.

Here we focus on instances that go to the left branch. Since the instances belonging to class A in the left example are all collected in environment 2, we can say that this splitting rule overfits a sensor data feature of instances belonging to class A collected in environment 2. In contrast, since the instances belonging to class A in the right example are collected in both the two environments (2 instances from environment 1 and 3 instances from environment 2), the splitting rule captures an inherent sensor data feature for instances belonging to class A.

Therefore, when we determine a splitting rule, we take into account environments where training instances were collected. As shown in Fig. 9, we introduce labels of environments and compute the information gain of a split using the environment labels as well as class labels. The standard decision tree learning algorithm computes information gain based only on class labels of instances. Based on the computed information gain, we find a split that yields little ambiguity in terms of class labels. (See [35] for more detail about decision tree learning.) In our proposed method, we employ environment labels in the computation of information gain in order to find a split that yields great ambiguity in terms of environment labels as well as little ambiguity in terms of class labels. Specifically, we compute modified information gain by

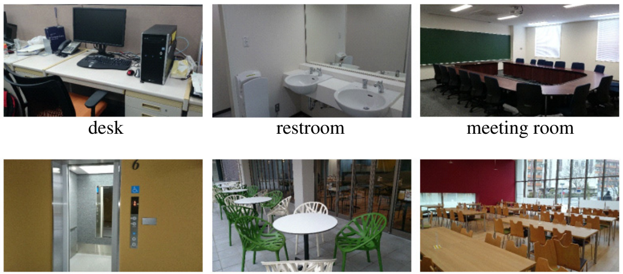
$$\begin{aligned} IG'(S) &= H_c(S) - H_e(S) \\ &\quad - \sum_{i \in \{L,R\}} \frac{|S^i|}{S} \big(H_c(S^i) - H_e(S^i)\big), \end{aligned}$$

where $H_c(S)$ shows Shannon entropy calculated based on class labels and $H_e(S)$ shows Shannon entropy calculated based on environment labels. By subtracting information gain computed based on environment labels from that computed based on class labels, we can obtain a split that yields great ambiguity in terms of environment labels as well as little ambiguity in terms of class labels.

We employ the above criterion $IG'(S)$ for constructing decision trees used in the random forest. We classify feature vectors collected at the same location cluster using the random forest and determine the final result for the cluster based on a majority vote.

Table 2. Our experimental environments

environment	department	built in	campus
1	Information science	2004	A
2	Information science	2015	A
3	Engineering	1992	A
4	Science	1966	B



desk restroom meeting room

elevator smoking area cafeteria

Figure 10. Six different places in environment 2

Table 3. Detailed information about locations

	env	size (m)	height (m)	floor	objects in place
desk	1	18×11	3	carpet	laptop PC
	2	18×10	3	carpet	desktop, laptop, display
	3	16×6	3	carpet	desktop, display, whiteboard
	4	7×4	2.5	tile	display, laptop
restroom	1	6×4	2	linoleum	
	2	8×4	2	linoleum	
	3	4×3	2	tile	
	4	6×5	2	tile	
meeting room	1	11×7	3	carpet	whiteboard, table
	2	10×10	3	carpet	blackboard, round table
	3	10×10	4	tile	blackboard, table
	4	8×5	2.5	tile	blackboard, table
elevator	1	2×1.5	2.5	tile	mirror (1m), wall: felt
	2	1.5×1.5	2.5	rubber	mirror (2m), wall: felt
	3	1.5×1.5	2.5	tile	mirror (2m), wall: tile
	4	1.5×1	2	tile	wall: tile
smoking area	1	n/a	n/a	tile	
	2	n/a	n/a	tile	table, chair
	3	n/a	n/a	concrete	
	4	n/a	n/a	tile	table, chair
cafeteria	1	15×6	6	tile	
	2	25×15	6	tile	
	3	35×25	12	tile	
	4	30×18	4	tile	

Table 4. Classification accuracy of methods

	precision	recall	F-measure
Proposed	0.791	0.780	0.778
RF	0.763	0.760	0.754
only mag	0.190	0.235	0.174
only press	0.219	0.284	0.220
only sound	0.158	0.166	0.161
only IR	0.512	0.554	0.524

EVALUATION

Data Set

To evaluate our method, we collected data from four different office/laboratory environments (buildings) in our university, which are listed in Table 2. One building is located at a different campus from the other buildings. In each environment, a participant collected sensor data in six locations; desk, restroom, meeting room, elevator, smoking area (outdoor resting place), and cafeteria. Since we have six different location classes, this problem is a six-class classification problem. We selected these six classes based on existing indoor positioning studies using active probing [12, 38] and places where a participant stayed for a long time in the participant's daily life. (We will introduce an evaluation using real life sensor data later.) The desk class means a desk of a participant in an office or laboratory. The smoking areas of the four environments are outdoor or semi-outdoor places. Fig. 10 shows places in environment 2 used in our study. Also, Table 3 shows detailed information about six locations in the four environments.

In each environment, a participant conducted 10 sessions of data collection, visiting six different places in an arbitrary order and performing activities related to the places. A smartphone (Google Nexus 6P) was attached around the neck as shown in Fig. 1 during the experiment. Sensors used in the experiment are listed in Table 1, and we also collected camera images to obtain ground truth. The duration of each session was about 10 minutes.

Evaluation methodology

Since the main contribution of this study is indoor semantics estimation, this evaluation focuses on the classification accuracy for the estimation. As for the Wi-Fi based clustering, our results did not have any errors, *i.e.,* the purity was 1 and six clusters were generated for each environment. The threshold used in the hierarchical clustering was 6, which achieved the best performance in our preliminary experiment.

We conducted our evaluation using "leave-one-environment-out" cross validation, where sensor data from one environment are used as test data and sensor data from the remaining environments are used to train a classifier, using the following methods.

- *Proposed*: This is our proposed method that employs magnetic and barometric pressure data collected using passive sensing and impulse responses collected using active probing. This method trains an environment-independent random forest classifier using environment labels. Note that here we do not use sound data collected using passive sensing because using the sound features degraded the classification performance, which will be investigated in detail later.
- *RF*: This method also employs magnetic and barometric pressure data collected using passive sensing and impulse responses collected using active probing. Note that this method uses the standard random forest algorithm to train a classifier.
- *only mag*: This is also our proposed method when we only use magnetic sensor data. This method also uses an environment-independent random forest classifier.
- *only press*: This is also our proposed method when we only use barometric pressure data.
- *only sound*: This is also our proposed method when we only use sound data.
- *only IR*: This is also our proposed method when we only use the impulse responses.

Classification accuracy for each of the above methods is evaluated using the macro-averaged precision, recall, and F-measure, calculated based on the classification results of feature vectors.

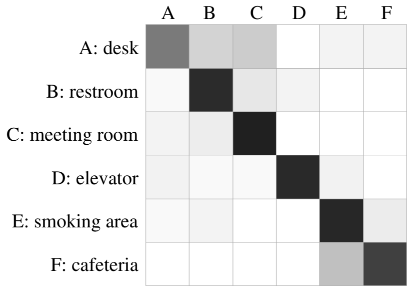


Figure 11. Visual confusion matrix for *Proposed*

Table 5. Classification accuracies of *Proposed* for six location classes

	precision	recall	F-measure
desk	0.778	0.525	0.627
restroom	0.729	0.833	0.778
meeting room	0.735	0.878	0.800
elevator	0.941	0.842	0.889
smoking area	0.708	0.850	0.773
cafeteria	0.857	0.750	0.800

Results

Classification performance

Table 4 shows the average precision, recall, and F-measure of *Proposed*. Surprisingly, our method could achieve about 78% average F-measure while this method does not use training data collected in a test environment. Fig. 11 shows a visual confusion matrix for *Proposed* created based on classification results of feature vectors. Our method accurately estimated location semantics of the test instances combining passive sensing and active probing while the accuracies for the desk class are somewhat poor. Since we employ "leave-one-environment-out" cross validation, our method could detect these location classes *without* using labeled training data collected in a test environment.

The accuracies for the desk class were somewhat poorer than those for the other classes. As shown in Fig. 11, several desk instances are mistakenly classified into the meeting room class. Also, Table 5 shows the classification accuracies for the six location classes. This may be because sensor data features for desk places in different environments are somewhat different. For example, while desktop PCs were placed on the desks in environments 2, 3 and 4, a laptop PC was placed in environment 1. Also, a whiteboard is placed around the desk in environment 3. Because whiteboards are also placed in meeting rooms, IR features related to these objects degrade the accuracies for the desk class.

In addition, the smoking area and cafeteria instances were somewhat confusing as shown in Fig. 11. This may be because the smoking area and cafeteria are large places and IR features related to reverberation are similar to each other.

Table 4 also shows the accuracies for *RF*, which employs the standard random forest algorithm. As shown in the result, we can confirm the effectiveness of our environment-independent learning method, and the improvement was about 2.4%.

Table 6. Classification accuracies of *Proposed* for four environments

environment	precision	recall	F-measure
1	0.801	0.768	0.769
2	0.893	0.886	0.885
3	0.791	0.780	0.778
4	0.807	0.788	0.791

Table 7. Classification accuracies when we do not use magnetic, barometric pressure, sound, or IR features

	precision	recall	F-measure
all	0.586	0.587	0.581
w/o mag and sound	0.773	0.760	0.754
w/o press and sound	0.588	0.609	0.594
w/o IR and sound	0.179	0.240	0.198
w/o sound	0.791	0.780	0.778

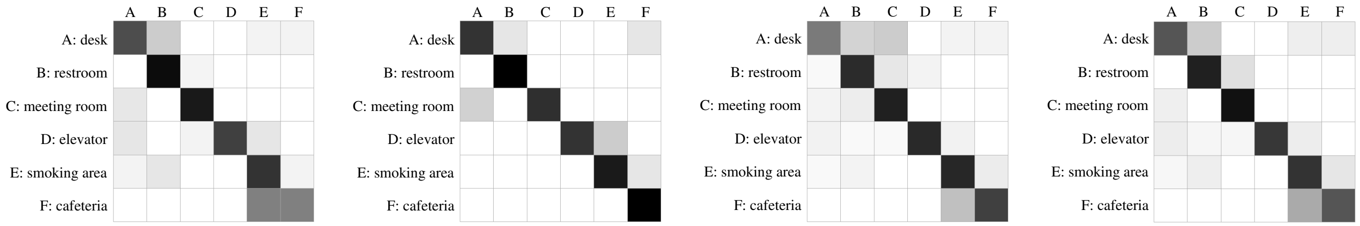
Classification performance for each environment

Table 6 shows the classification accuracies of *Proposed* for the four environments. As shown in the result, our method could achieve good accuracies in the environments. The accuracies for environment 1 were somewhat poorer than those for the other environments because the accuracies related to the smoking area and cafeteria were poor. Fig. 12 shows confusion matrices of *Proposed* for the four environments. As shown in Fig. 12 (a), many cafeteria instances were mistakenly classified into the smoking area class. This may be because the cafeteria in environment 1 is somewhat smaller than the cafeterias in the other environments. In environment 3, several desk instances were mistakenly classified into the meeting room class. This is because a whiteboard is placed around the desk in environment 3 as mentioned above.

Contributions of sensors

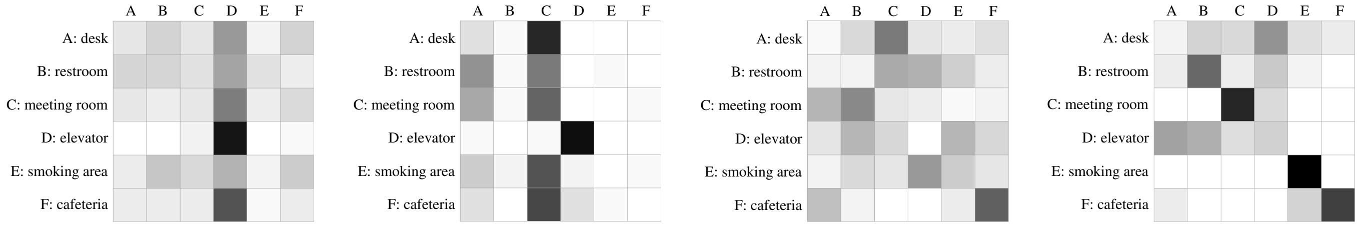
Table 4 also shows the classification accuracies when we use only a single sensor. As shown in the results, the IR was the best contributor and *only IR* achieved about 50% accuracies. Fig. 13 shows confusion matrices of *only mag*, *only press*, *only sound*, and *only IR*. As shown in Fig. 13 (d), it was difficult for *only IR* to correctly estimate location semantics of the desk and elevator instances. Also, the sound features were useful for detecting the cafeteria instances as shown in Fig. 13 (c) because they can capture information about ambient sounds such as crowd of people. However, the overall accuracies were quite poor. In addition, the magnetic and barometric pressure sensor data features are useful for detecting only the elevator instances. Because *only IR* cannot precisely detect the elevator instances, we can say that the magnetic and barometric pressure sensor data are served to complement the IR features.

Table 7 shows the classification accuracies when we do not use a particular sensor(s). Note that *all* uses all the sensor data features including sound features collected using passive sensing. As shown in the results, *w/o mag and sound* greatly outperformed *w/o press and sound*, and we can say that the barometric pressure sensor is much more useful than the magnetic sensor. This is because, as shown in Fig. 5 (a), the magnetic feature values do not always exhibit high variance values. In contrast, as shown in Fig. 6 (a), the barometric pressure feature values are always large when an elevator moves. Also, as shown in Table 7, using the sound fea-



(a) environment 1 (b) environment 2 (c) environment 3 (d) environment 4

Figure 12. Visual confusion matrix of *Proposed* for each environment



(a) *only mag* (b) *only press* (c) *only sound* (d) *only IR*

Figure 13. Visual confusion matrices for *only mag*, *only press*, *only sound*, and *only IR*

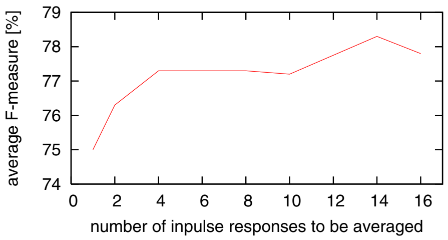


Figure 14. Relationship between the number of impulse responses to be averaged and F-measure

Table 8. Classification accuracies of *Proposed* for six location classes calculated based on classification results of location clusters in each session

	precision	recall	F-measure
desk	1.00	0.487	0.655
restroom	0.975	1.00	0.987
matting room	0.796	0.975	0.876
elevator	0.975	0.975	0.975
smoking area	0.792	0.950	0.864
cafeteria	0.714	0.750	0.732
average	0.875	0.856	0.848

tures degraded the classification accuracy (*all* vs *w/o sound*). While the sound features were useful for detecting the cafeteria instances as is shown in Fig. 13 (c), we confirmed that the sound features degraded the classification accuracies for the desk, restroom, and meeting room classes because these places are quiet.

Effect of synchronous averaging

The above results were obtained using the synchronous averaging of 16 impulse responses. Here we investigate the number of impulse responses to be averaged. Fig. 14 shows the transition of the average F-measure for *Proposed* when we change the number of impulse responses to be averaged. As shown in the result, the F-measure gradually increases as the number of impulse responses becomes large.

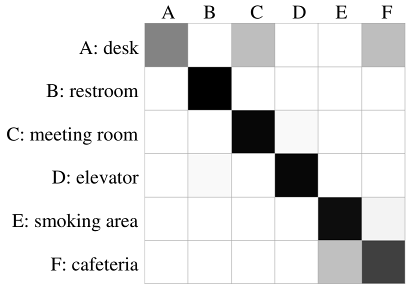


Figure 15. Visual confusion matrix for *Proposed* calculated based on classification results of location clusters in each session

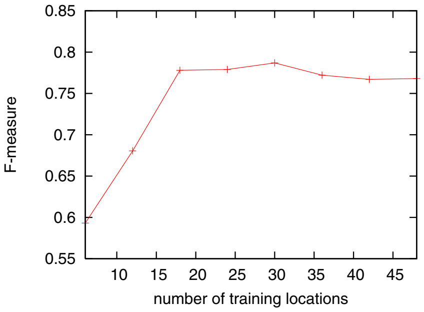


Figure 16. Transition of F-measures for *Proposed* when we changed the amount of training data

Amount of training data

We collected additional training data in 30 different locations. (5 locations for each location class) Fig. 16 shows the transition of the F-measures for *Proposed* when we changed the number of training locations. As shown in the result, the F-measure does not change even when we increase the amount of training data. To further improve the accuracy, we believe that additional information (*e.g.,* time when a user visits a location) is required.

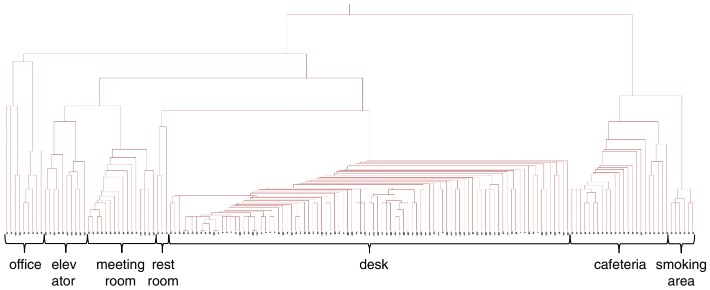


Figure 17. Dendrogram of hierarchical clustering constructed using real life sensor data collected in environment 2

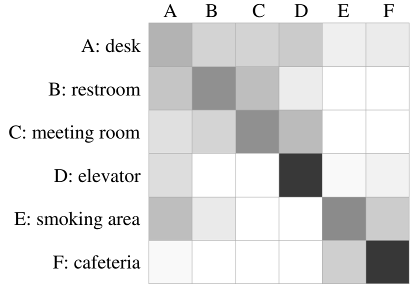


Figure 18. Visual confusion matrix for *Proposed* using real life data computed based on classification results of feature vectors

Classification performance based on location clusters

The above results were computed based on classification results of feature vectors. Based on the predicted location classes of the feature vectors, we can determine a location class for each location cluster using a majority vote. Table 8 and Fig. 15 show the results computed based on the classification results of the clusters. Because the majority vote reduces the effect of sporadic errors, we could achieve high accuracy of about 85%.

In particular, the accuracies for restroom and elevator were very high. When the elevator is not moving, barometer data do not change and thus the accuracies for elevator decreases. However, by aggregating classification results of feature vectors, we could greatly improve the accuracy. Also, Wi-Fi signals in a restroom are stable because there are few people. Therefore, the Wi-Fi-based majority vote worked well.

Evaluation using real life data

We collected sensor data in a participant's real life for a day in environment 2. Fig. 17 shows a constructed dendrogram from the collected Wi-Fi data. In addition to instances belonging to the six location classes, our acceleration based place detection method found a place corresponding to a university office because the participant was called to the office. However, this is an unusual event and it was difficult for us to collect long-term training data at a department office in each environment, we used the remaining six location classes in this study.

Figure 18 shows a visual confusion matrix computed based on classification results of feature vectors. While the average F-measure was 0.517, the average F-measure computed based on classification results of location clusters was 1.00. This is because we could determine the final classification results based on sensor data collected for a long time using a majority vote.

DISCUSSION

Device heterogeneity

We collected additional sensor data using a Google Nexus 5 in environment 2 to investigate the effect of the device heterogeneity. We train a classifier on training data collected by Google Nexus 6P in environments 1, 3, and 4, and test sensor data collected by Nexus 5 in environment 2. The average F-measure was only 0.520, and all desk and restroom instances were mistakenly classified into the meeting room class. This may be caused by the differences in microphone sensitivity and positions of the microphones (and speakers) in the two smartphones. While we found that device-dependent training is required, our experiment revealed that using training data from only three environments achieves accurate estimation.

We can also cope with this problem by learning sound-feature differences between two different devices. By recording the same sound using the two devices, we can calculate a transformation matrix that transforms sensor data collected by one device to those by another device. The implementation of this is a part of our future work.

Time dependence of sensor data

In our experiment, we collected sensor data at cafeterias during lunch hour because people usually visit cafeterias during lunch hour. Note that, sensor data collected during lunch hour and the other hours are different because the number of customers is different. Since our method relies on machine learning techniques, we should collect data at various hours to cope with the problem.

Energy consumption

We assume that acceleration sensors are always on, and we collect data from other sensors only when a user is estimated to be staying still. While the energy consumption of a Wi-Fi module is much higher than that of an acceleration sensor (about six times [22]), we assume that we collect Wi-Fi data (and magnetic and pressure sensor data) for only several seconds at each place. Also, the energy consumption of magnetic sensors, barometers, and speakers are low [40]. While the energy consumption of a microphone is high (about half of Wi-Fi) [22], we turn on the microphone for only about 10 seconds at each place.

CONCLUSION

This paper presented a method for predicting the user's location semantics without using training data collected by the user in his/her environment. In this study, we combined passive sensing and active sound probing to capture and learn inherent sensor data features for each location class using labeled training data collected in other environments. As a part of our future work, we plan to implement our sound probing application on a smart watch to enhance the feasibility of our approach.

Acknowledgment

This study is partially supported by JST CREST.

REFERENCES

1. Nobuharu Aoshima. 1981. Computer-generated pulse signal applied for sound measurement. *The Journal of the Acoustical Society of America* 69, 5 (1981), 1484–1488.

2. Satoshi Asano, Yuki Wakuda, Noboru Koshizuka, and Ken Sakamura. 2012. A robust pedestrian dead-reckoning positioning based on pedestrian behavior and sensor validity. In *IEEE/ION Position Location and Navigation Symposium (PLANS 2012)*. 328–333.

3. Martin Azizyan, Ionut Constandache, and Romit Roy Choudhury. 2009. SurroundSense: mobile phone localization via ambience fingerprinting. In *MobiCom 2009*. 261–272.

4. Dipyaman Banerjee, Sheetal K Agarwal, and Parikshit Sharma. 2015. Improving floor localization accuracy in 3D spaces using barometer. In *International Symposium on Wearable Computers (ISWC 2015)*. 171–178.

5. Ling Bao and Stephen S Intille. 2004. Activity recognition from user-annotated acceleration data. In *Pervasive 2004*. 1–17.

6. Xuan Bao, Bin Liu, Bo Tang, Bing Hu, Deguang Kong, and Hongxia Jin. 2015. PinPlace: associate semantic meanings with indoor locations without active fingerprinting. In *UbiComp 2015*. 921–925.

7. Mark Blum, Alex Sandy Pentland, and Gehrard Tröster. 2006. Insense: Interest-based life logging. *IEEE Multimedia* 13, 4 (2006), 40–48.

8. Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5–32.

9. Jianfeng Chen, Alvin Harvey Kam, Jianmin Zhang, Ning Liu, and Louis Shue. 2005. Bathroom activity monitoring based on sound. In *Pervasive 2005*. 47–61.

10. Ionut Constandache, Xuan Bao, Martin Azizyan, and Romit Roy Choudhury. 2010. Did you see Bob?: human localization using mobile phones. In *MobiCom 2010*. 149–160.

11. Michael Cowling. 2004. *Non-speech environmental sound recognition system for autonomous surveillance*. Ph.D. Dissertation. Griffith University.

12. Mingming Fan, Alexander Travis Adams, and Khai N Truong. 2014. Public restroom detection on mobile phone via active probing. In *International Symposium on Wearable Computers (ISWC 2014)*. 27–34.

13. Angelo Farina. 2000. Simultaneous measurement of impulse response and distortion with a swept-sine technique. In *Audio Engineering Society Convention 108*. Audio Engineering Society.

14. Angelo Farina. 2007. Advancements in impulse response measurements by sine sweeps. In *Audio Engineering Society Convention 122*. Audio Engineering Society.

15. Brian Ferris, Dieter Fox, and Neil Lawrence. 2007. WiFi-SLAM using Gaussian process latent variable models. In *IJCAI 2007*. 2480–2485.

16. Brandon Gozick, Kalyan Pathapati Subbu, Ram Dantu, and Tomyo Maeshiro. 2011. Magnetic maps for indoor navigation. *IEEE Transactions on Instrumentation and Measurement* 60, 12 (2011), 3883–3891.

17. Michael Hardegger, Gerhard Tröster, and Daniel Roggen. 2013. Improved ActionSLAM for long-term indoor tracking with wearable motion sensors. In *International Symposium on Wearable Computers (ISWC2013)*. 1–8.

18. Steve Hodges, Lyndsay Williams, Emma Berry, Shahram Izadi, James Srinivasan, Alex Butler, Gavin Smyth, Narinder Kapur, and Ken Wood. 2006. SenseCam: A retrospective memory aid. In *Ubicomp 2006*. 177–193.

19. Sibren Isaacman, Richard Becker, Ramón Cáceres, Stephen Kobourov, Margaret Martonosi, James Rowland, and Alexander Varshavsky. 2011. Identifying important places in people's lives from cellular network data. In *Pervasive 2011*. 133–151.

20. Yifei Jiang, Xin Pan, Kun Li, Qin Lv, Robert P Dick, Michael Hannigan, and Li Shang. 2012. ARIEL: Automatic Wi-Fi based room fingerprinting for indoor localization. In *Ubicomp 2012*. 441–450.

21. Stephen C Johnson. 1967. Hierarchical clustering schemes. *Psychometrika* 32, 3 (1967), 241–254.

22. Christine E Jones, Krishna M Sivalingam, Prathima Agrawal, and Jyh Cheng Chen. 2001. A survey of energy efficient network protocols for wireless networks. *wireless networks* 7, 4 (2001), 343–358.

23. Kai Kunze and Paul Lukowicz. 2007. Symbolic Object Localization Through Active Sampling of Acceleration and Sound Signatures. *UbiComp 2007* (2007), 163–180.

24. Heinrich Kuttruff. 2009. *Room acoustics*. CRC Press.

25. Anthony LaMarca, Yatin Chawathe, Sunny Consolvo, Jeffrey Hightower, Ian Smith, James Scott, Timothy Sohn, James Howard, Jeff Hughes, Fred Potter, and others. 2005. Place lab: Device positioning using radio beacons in the wild. In *Pervasive 2005*. 116–133.

26. Byoungyoung Lee, Jinoh Oh, Hwanjo Yu, and Jong Kim. 2011. Protecting location privacy using location semantics. In *KDD 2011*. 1289–1297.

27. Hong Lu, Wei Pan, Nicholas D. Lane, Tanzeem Choudhury, and Andrew T. Campbell. 2009. SoundSense: scalable sound sensing for people-centric applications on mobile phones. In *MobiSys 2009*. 165–178.

28. Paul Lukowicz, Jamie A Ward, Holger Junker, Mathias Stäger, Gerhard Tröster, Amin Atrash, and Thad Starner. 2004. Recognizing workshop activity using body worn microphones and accelerometers. In *Pervasive 2004*. 18–32.

29. Takuya Maekawa and Shinji Watanabe. 2011. Unsupervised Activity Recognition with User's Physical Characteristics Data. In *International Symposium on Wearable Computers (ISWC 2011)*. 89–96.

30. Takuya Maekawa, Yutaka Yanagisawa, Yasue Kishino, Katsuhiko Ishiguro, Koji Kamei, Yasushi Sakurai, and Takeshi Okadome. 2010. Object-based activity recognition with heterogeneous sensors on wrist. In *Pervasive 2010*. 246–264.

31. Takuya Maekawa, Yutaka Yanagisawa, Yasue Kishino, Koji Kamei, Yasushi Sakurai, and Takeshi Okadome. 2008. Object-blog system for environment-generated content. *IEEE Pervasive Computing* 7, 4 (2008), 20–27.

32. Busra Ozdenizci, Vedat Coskun, and Kerem Ok. 2015. NFC internal: An indoor navigation system. *Sensors* 15, 4 (2015), 7571–7595.

33. Nissanka B Priyantha, Anit Chakraborty, and Hari Balakrishnan. 2000. The cricket location-support system. In *MobiCom 2000*. 32–43.

34. Teemu Pulkkinen, Teemu Roos, and Petri Myllymäki. 2011. Semi-supervised learning for WLAN positioning. In *ICANN 2011*. 355–362.

35. John Ross Quinlan. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

36. Anshul Rai, Krishna Kant Chintalapudi, Venkata N Padmanabhan, and Rijurekha Sen. 2012. Zee: Zero-effort crowdsourcing for indoor localization. In *MobiCom 2012*. 293–304.

37. Patrick Robertson, Maria Garcia Puyol, and Michael Angermann. 2011. Collaborative pedestrian mapping of buildings using inertial sensors and FootSLAM. In *International Technical Meeting of The Satellite Division of the Institute of Navigation*.

38. Mirco Rossi, Julia Seiter, Oliver Amft, Seraina Buchmeier, and Gerhard Tröster. 2013. RoomSense: an indoor positioning system for smartphones using active sound probing. In *the 4th Augmented Human International Conference*. 89–95.

39. Thomas D Rossing, F Richard Moore, and Paul A Wheeler. 2002. *The science of sound*. Vol. 3. Addison Wesley San Francisco.

40. Kartik Sankaran, Minhui Zhu, Xiang Fa Guo, Akkihebbal L Ananda, Mun Choon Chan, and Li-Shiuan Peh. 2014. Using mobile phone barometer for low-power transportation context detection. In *the 12th ACM Conference on Embedded Network Sensor Systems (SenSys 2014)*. 191–205.

41. Manfred R Schroeder. 1979. Integrated-impulse method measuring sound decay without using impulses. *The Journal of the Acoustical Society of America* 66, 2 (1979), 497–500.

42. Daisuke Taniuchi and Takuya Maekawa. 2014. Robust Wi-Fi based indoor positioning with ensemble learning. In *IEEE 10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2014)*. 592–597.

43. Daisuke Taniuchi and Takuya Maekawa. 2015. Automatic Update of Indoor Location Fingerprints with Pedestrian Dead Reckoning. *ACM Transactions on Embedded Computing Systems (TECS)* 14, 2 (2015), 27:1–27:23.

44. Stephen P Tarzia, Peter A Dinda, Robert P Dick, and Gokhan Memik. 2011. Indoor localization without infrastructure using the acoustic background spectrum. In *MobiSys 2011*. 155–168.

45. Yu-Chih Tung and Kang G Shin. 2015. EchoTag: accurate infrastructure-free indoor location tagging with smartphones. In *MobiCom 2015*. 525–536.

46. He Wang, Souvik Sen, Ahmed Elgohary, Moustafa Farid, Moustafa Youssef, and Romit Roy Choudhury. 2012. No need to war-drive: Unsupervised indoor localization. In *MobiSys 2012*. 197–210.

47. Yapeng Wang, Xu Yang, Yutian Zhao, Yue Liu, and Laurie Cuthbert. 2013. Bluetooth positioning using RSSI and triangulation methods. In *IEEE Consumer Communications and Networking Conference (CCNC 2013)*. 837–842.

48. Roy Want, Andy Hopper, Veronica Falcão, and Jonathan Gibbons. 1992. The active badge location system. *ACM Transactions on Information Systems (TOIS)* 10, 1 (1992), 91–102.

49. Joe H Ward Jr. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association* 58, 301 (1963), 236–244.

50. Muchen Wu, Parth H Pathak, and Prasant Mohapatra. 2015. Monitoring building door events using barometer sensor in smartphones. In *UbiComp 2015*. 319–323.

51. Qiang Xu, Rong Zheng, and Steve Hranilovic. 2015. IDyLL: Indoor Localization using Inertial and Light Sensors on Smartphones. In *UbiComp 2015*. 307–318.

52. Zhixian Yan, Dipanjan Chakraborty, Christine Parent, Stefano Spaccapietra, and Karl Aberer. 2013. Semantic trajectories: Mobility data computation and annotation. *ACM Transactions on Intelligent Systems and Technology (TIST)* 4, 3 (2013), 49.

53. Zengbin Zhang, David Chu, Xiaomeng Chen, and Thomas Moscibroda. 2012. Swordfight: Enabling a new class of phone-to-phone action games on commodity phones. In *MobiSys 2012*. 1–14.

54. Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. 2014. Learning deep features for scene recognition using places database. In *NIPS 2014*. 487–495.