Transferring Positioning Model for Device-free Passive Indoor Localization

Kazuya Ohara, Takuya Maekawa
{ohara.kazuya,maekawa}@ist.osaka-u.ac.jp
Graduate School of Information Science and Technology, Osaka University

Yasue Kishino, Yoshinari Shirai, Futoshi Naya
{kishino.yasue,shirai.yoshinari,naya.futoshi}
@lab.ntt.co.jp
NTT Communication Science Laboratories

ABSTRACT
This paper proposes a new method that makes it easy for us to construct a positioning model for device-free passive indoor localization by using model transfer techniques. With device-free passive indoor positioning, a wireless sensor network is used to detect the movement of a person based on the fact that RF signals transmitted between a transmitter and a receiver are affected by human movement. However, because device-free passive indoor positioning relies on machine learning techniques, we must collect labeled training data at many training points in an end user's environment. This paper proposes a method that transfers a signal strength model used for locating a person obtained in another environment (source environment) to the end user environment. With the transferred models, we can construct a positioning model for the end user environment inexpensively. Our evaluation showed that our method achieved almost the same positioning performance as a supervised method that requires labeled training data obtained in an end user's environment.

Categories and Subject Descriptors: H.3.4 Information storage and retrieval: Systems and software; I.5.2 Design Methodology: Pattern analysis.

Keywords: Indoor positioning, device-free passive positioning, model transfer.

INTRODUCTION
Due to the recent proliferation of wireless LAN communications, the price of Wi-Fi modules has fallen in recent years and this has triggered many indoor positioning studies based on Wi-Fi technologies. Wi-Fi based indoor positioning studies attempt to estimate the position of a person who possesses a Wi-Fi signal receiver such as a smart phone. This kind of indoor positioning method is useful for such applications as indoor navigation systems for underground malls and shopping malls. However, because this method assumes that a user always carries a signal receiver, it is difficult to apply it to applications for daily living such as the surveillance of an independently living elderly person and smart homes automation. Recently, a device-free passive indoor positioning technique has been studied that does not require the subject to carry a signal receiver [30, 22, 4, 11].

UbiComp '15, September 7-11, 2015, Osaka, Japan.

http://dx.doi.org/10.1145/2750858.2806061

In device-free passive indoor positioning, a wireless sensor network is used to detect changes in the environment and track the location of a person passively [30]. The device-free passive indoor positioning concept relies on the fact that RF signals are affected by changes in the environment, i.e., human movement. We place sensor nodes that continually record signal strength from a signal transmitter, and these signals are used to detect the changes in an environment and locate a person. Because the human body interferes with wireless signal transmissions, the signal strength received by a sensor node installed in an indoor environment depends on the position of the person in the indoor environment. Because multiple sensor nodes are installed in the environment, the set of received signal strength values (signal strength vector) obtained from the sensor nodes at time t depends on the position of the person at time t. Therefore, the vector is considered to be a fingerprint of the position. Fingerprinting techniques employ a set of such fingerprints to locate a person.

Because the fingerprinting approach relies on machine learning techniques, it consists of a training phase and a test phase. In the training phase, Wi-Fi signals (i.e., the unique identifiers of signal transmitters, the unique identifiers of signal receivers, and the received signal strengths from transmitters) are observed when a person is at known coordinates in a target environment. Then, the observed fingerprints are used as training data to learn an indoor positioning model. The positioning model estimates a person's coordinates by using signal strength data observed by the receivers (sensor nodes). However, in order to learn the indoor positioning model, we should collect labeled training data at many positions in the target environment. Collecting such training data in an end user's house is very costly and impractical because the end user has to input his/her coordinates at many training points by using an input device such as a smart phone. In this study, we try to construct an indoor positioning model for a target environment without using any labeled training data obtained at the target environment, by transferring a signal strength model from other environments (source environments) to the target environment. By doing so, we can easily construct a positioning model for an end-user environment by re-using labeled training data obtained from several source environments in advance. This approach is typically used in activity and speech recognition studies where labeled data from source users (speakers) are used to train recognition models in order to reduce the burden on end users.

The above-mentioned existing passive positioning studies mainly use raw signal strength data (mean strength within a time window) to construct a positioning model. However, because the signal strength changes according to various environmental factors such as changes in humidity and temperature, and the positions of house furnishings [2, 15, 1], device-free passive indoor positioning based on raw signal strength values (or mean values) in real environments is reportedly impractical [16]. Therefore, in the same way as recent device-free passive positioning studies [16], this study employs the variance of signal strength values within a sliding time window. Because water makes up most of the human body, when a person passes between a signal transmitter and a signal receiver, the signal strength received by the receiver changes, and consequently the variance value of the signal strengths increases. Recent studies have attempted to detect a person passing between a transmitter and a receiver by detecting the increase in the variance value in order to locate that person with sub-room-level accuracy. On the other hand, this study tries to achieve more accurate positioning by detecting a position (point) on a line segment connecting the transmitter and the receiver that the person passes, e.g., a person passes three meters from the receiver. We construct a model for estimating the passing point by using training data obtained in another environment (source environment), and we then transfer the model to the target environment. Furthermore, because device-free passive positioning methods can estimate a person's position only when the person passes between a transmitter and a receiver, we also attempt to track the position using a particle filter, even when a passing is not detected.

The variance values that change when a person passes between a transmitter and a receiver depend on such factors as whether or not there is an obstacle such as a wall between the transmitter and the receiver, and the material of the obstacle. In this study, we find a pair consisting of a transmitter and a receiver in a source environment that has a similar signal strength feature to that of a pair in a target environment based on unlabeled sensor data obtained in the target environment and floor plans (layout of walls) of the target and source environments. And then we transfer the model of the pair in the source environment to the pair in the target environment. We consider that obtaining information about the layout of walls is much easier for an end user than obtaining information about the material of the walls, and collecting unlabeled sensor data is also much easier than collecting labeled data. When end users collect labeled sensor data, they have to manually record their precise coordinates in their environments (e.g., by using video recordings). On the other hand, in our approach, we ask the end user to walk at random around the environment to obtain the unlabeled sensor data.

To the best of our knowledge, this is the first study that attempts to design a model transfer method for device-free passive indoor positioning. The research contributions of this paper are that (1) we propose a model transfer method for device-free passive indoor positioning that requires the end user to have only a floor plan of a target environment and unlabeled sensor data obtained in the target environment, and (2) we confirm the effectiveness of our method in real environments.

RELATED WORK

Standard Wi-Fi based indoor positioning

Indoor positioning methods rely on signaling technologies such as infrared (IR) [26], ultrasound [18], radio-frequency identification (RFID) [31], ultra-wideband (UWB) [23], FM radio wave [12, 17], Bluetooth [25], and Wi-Fi [13]. In particular, because Wi-Fi technology is widespread, many researchers have attempted to construct indoor positioning systems by utilizing Wi-Fi access points (APs). An advantage of Wi-Fi based positioning is that we can use a smart phone with a Wi-Fi module as a signal receiver. For standard Wi-Fi based positioning, fingerprinting techniques have also usually been employed to measure indoor positions [13]. Fingerprinting employs a training phase in which Wi-Fi signals (i.e., the unique MAC addresses of APs and the received signal strengths from APs) are observed at known coordinates. A set of APs and their signal strengths constitute a fingerprint that is unique to those coordinates.

Constructing indoor positioning models with little effort

Here we introduce studies of standard Wi-Fi positioning that try to reduce the amount of work related to collecting Wi-Fi fingerprints. In [9], the authors attempted 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. Several studies constructed radio maps with no supervision by using simultaneous localization and mapping (SLAM) techniques [7, 20]. In [8], the authors perform SLAM based on the fact that certain daily activities are undertaken at particular places (e.g., sleeping in a bedroom). In [19, 10], the authors also tried to construct radio maps automatically with the pedestrian dead reckoning (PDR) technique. In [24], the authors attempted to track a user with no Wi-Fi fingerprint. The authors corrected the accumulated error of acceleration-based PDR by employing landmarks with known coordinates. With this approach, some sensors may observe specific sensor data values at a given landmark. For example, an acceleration sensor inside an elevator may observe characteristic signals. On the other hand, we try to reduce the work involved in collecting Wi-Fi fingerprints by constructing an indoor positioning model based on training data obtained in other environments.

Adaptive indoor positioning

We introduce Wi-Fi positioning studies that attempt to cope with the instability of Wi-Fi based positioning methods caused by changing environmental dynamics. In [2], the authors investigated the effect of environmental factors (people, doors, and humidity) on indoor positioning with Wi-Fi, and developed a sensor-network-assisted adaptive indoor positioning method by sensing the environmental factors. In [29], to cope with the temporal dynamics of signal strengths, the authors installed small numbers of Wi-Fi sensor nodes in an environment, and employed a regression analysis to learn/estimate the temporal predictive relationship between the signal strength values received by the sensor nodes and that received at a test point. On the other hand, recent device-free passive indoor positioning studies (including our study) have used the variance values of signal strengths instead of raw signal strength values (or mean values) to cope with the environmental dynamics [16, 30]. For more detail about the usefulness of variance values, refer to [16] and [30].

Device-free passive indoor positioning

Because standard Wi-Fi based indoor positioning assumes that a user always possesses a signal receiver, it is difficult to use this approach for such applications as the surveillance of an independently living elderly person. Therefore, device-free passive indoor positioning technologies have recently been attracting attention. In [30], the authors collect signal strengths with receivers when a person is at each training point in a training phase. In a test phase, when a signal strength vector s is given, they estimate the position of a person x that maximizes $P(x|s)$ by using the Bayes' theorem $P(x|s) = \frac{P(s|x)P(x)}{P(s)}$. In addition, they test several methods for modeling the conditional probability $P(s|x)$ such as the Gaussian distribution, a histogram, and the Gaussian kernel [11]. Furthermore, they pre-process received signal data by using the moving average and the moving variance [30]. In [22], the authors evaluated the above methods in real environments and investigated the effects of various parameters such as the number of nodes and window size. In [28], the authors employed a discriminative classifier, e.g., linear discriminant analysis, non-linear discriminant analysis, and instance based classification, to estimate which training point a user is at.

Based on the above device-free passive indoor positioning studies, this paper proposes a model transfer method for reducing the effort involved in training an indoor positioning model.

Modeling signal propagation

Many researchers have investigated the features of radio wave propagation by employing, for example, path loss and multipath delay analysis in indoor/outdoor environments [3, 21, 6]. If we can simulate the received signal strength from a transmitter at any point based on these studies, we can construct an indoor positioning model for an end user environment without the need for labeled training data. However, it is very difficult to calculate signal attenuation in indoor environments because it is affected by various factors such as wall material, fixtures, and furnishings. Also, it is difficult for the end user to prepare such information about the factors in his/her environment. In this study, we employ unlabeled sensor data obtained in a target environment without using such information.

PROPOSED METHOD

Assumed environment

In this paper, we assume that multiple nodes that can receive Wi-Fi signals are installed in an indoor environment as shown in Fig. 1. Also, we install a hub (Wi-Fi AP) that transmits Wi-Fi signals and aggregates signal strength information measured by the nodes. Each node measures the strengths of signals transmitted from the AP and sends the information to the AP. Then, the AP or a computer connected to the AP estimates a person's position by using the aggregated signal strength information. That is, we can obtain information

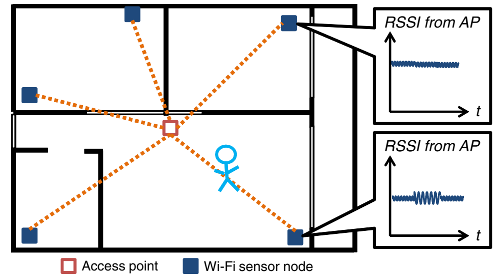

Figure 1. Example setup of sensor nodes and access point.

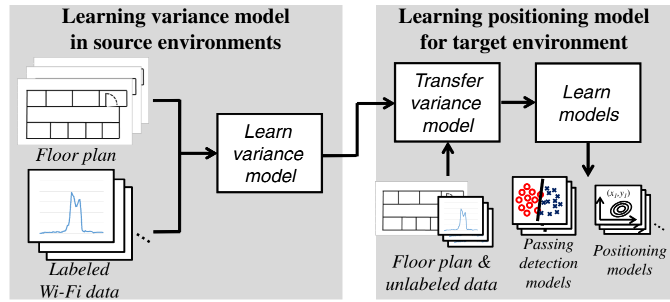

Figure 2. Overview of our method for training positioning model of target environment.

about the signal strength between each node and the AP as shown in Fig. 1 (dotted line segments). For each node and AP pair, we can obtain the signal strength received by the node.

Moreover, because we employ device-free passive indoor positioning, we assume that there is one person in an indoor environment. In this study, we use Raspberry Pi micro computers with two Wi-Fi modules as sensor nodes. We use tcpdump (monitor mode) to measure signal strength information. Our sensor node measures signal strength values at about 1.5 kHz.

Overview

Fig. 2 shows an overview of our method for training the positioning model of a target environment. For each transmitter and receiver pair in a source environment (or environments), we first learn the relationship between the variance value of signal strengths and a position (point) on a line segment connecting the transmitter and the receiver that a person passes. We call the model *a signal strength variance model* or *a variance model*. With a variance model, we can compute a variance value v that will be measured when a person passes a segment x meters from a segment edge. We then associate a pair in the source environment with a pair in the target environment whose signal strength characteristics seem to be similar. After that, we transfer the variance model of the pair in the source environment to that of the pair in the target environment. Finally, we learn a model for detecting a person who passes between a node and an AP, and also learn models for estimating the coordinates of the user. By using the outputs of the models, we track a person in the target environment based on a particle filter, which is a state-of-the-art robust tracking method.

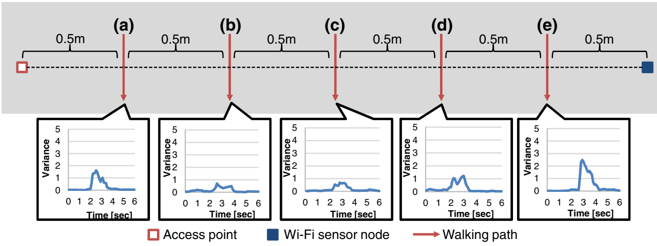

Figure 3. Example sensor data: there is no obstacle between AP and node. The distance between AP and node is three meters.

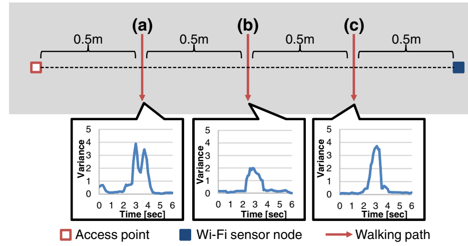

Figure 4. Example sensor data: there is no obstacle between AP and node. The distance between AP and node is two meters.

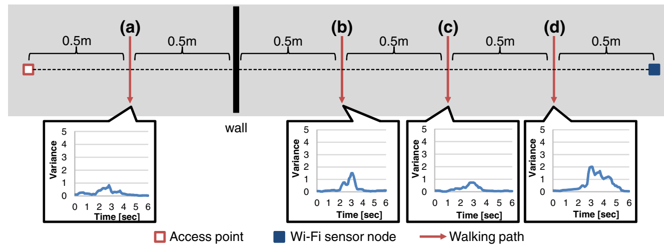

Figure 5. Example sensor data: there is a wall (gypsum board) between AP and node. The distance between AP and node is three meters.

Here we summarize the information required for constructing an indoor positioning model for the target environment; floor plans of the source and target environments (positions of APs and nodes, and layout of walls), labeled sensor data obtained in the source environments, unlabeled sensor data obtained in the target environment, and sensor data obtained when there is no person in the source and target environments. The floor plans, unlabeled sensor data, and sensor data obtained when no one is present are used to associate a pair in the source environment with a pair in the target environment. In this work, training data obtained in the source environment are time series of signal strength variances measured by each receiver and the trajectory information about a person. From the trajectory, we can know the person's coordinates in the source environment at any time slice.

In this work, for each time slice t, we judge whether or not the person passes between an AP and a node by using sensor data obtained at time t. When the person is judged to have passed between any of the pairs, we estimate the person's coordinates in detail by using sensor data obtained at time t. Otherwise, we estimate the coordinates of the person based on a predefined motion model.

Learning variance model in source environment

Example data

We first show example sensor data. As shown in Fig. 3, we installed an AP and a sensor node and collected the strength values of signals from the AP by using the node when a person passed between the AP and the node. Fig. 3 shows a time series of signal strength variance values observed by the node when the person walked along each path[1]. As shown in the figure, the variance values become larger as the path becomes closer to the transmitter or the receiver. This is because, when the human body is close to the transmitter or the receiver, it greatly impedes the signal to the receiver or from the transmitter. As shown in Fig. 3, the variance value depends on the position (point) on a line segment connecting the transmitter and the receiver that the person passes. With the variance value, we estimate the person's position on the line segment connecting the AP and the node and track the person using a particle filter. Note that, although many existing studies employ the mean value of signal strengths as vector element values, we employ the variance of signal strengths.

[1]The variance value shows the moving variance computed for each sliding window with a 90% overlap whose window size is one second.

We also installed an AP and a node so that the distance between them was two meters as shown in Fig. 4 and collected sensor data. As shown in the figure, we observed sensor data that are different from those shown in Fig. 3. The variance values in Fig. 4 are somewhat larger than those in Fig. 3. Therefore, when we transfer a model, we should select a pair in the source environment whose distance is similar to that of a pair in the target environment.

Fig. 5 shows variance data time series obtained when there is a wall (gypsum board) between an AP and a node. As shown in the figure, the variance values become larger as the walking path approaches the wall or the receiver. This may be caused by the signal attenuation and diffraction of the wall. On the other hand, Fig. 5 also shows a variance data time series obtained when a person walked between the wall and the AP. As shown in the figure, the variance values are small. As above, because the feature of the variance data depends greatly on the area (sub-line segment) divided by walls, we should construct a variance model for each sub-line segment. (For example, the line segment connecting a transmitter and a receiver in Fig. 5 consists of two sub-line segments.)

Fig. 6 shows sensor data obtained when there are two walls between an AP and a node. Also, as shown in Fig. 5, when a person passes a sub-line segment whose end points are a transmitter and a wall, the variance values do not increase greatly. On the other hand, as shown in Fig. 5, when a person passes a sub-line segment whose end points are a wall and a receiver, the variance values increase greatly. That is, the feature of the variance model depends greatly on its corresponding end points. Therefore, when we perform model transfer, we should associate a sub-line segment in a target environment with that in a source environment whose end points are same as those of the sub-line segment in the target environment.

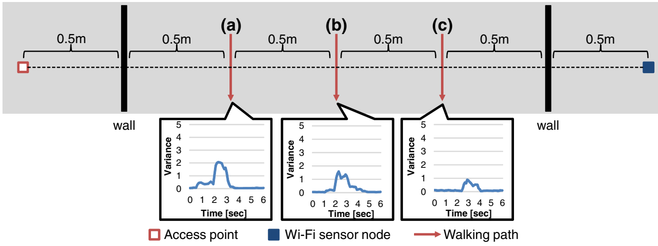

Figure 6. Example sensor data: there are two walls (gypsum board) between AP and node. The distance between AP and node is three meters.

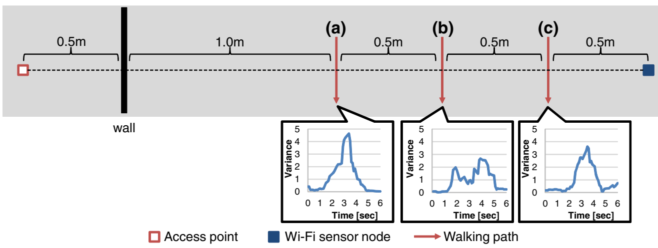

Figure 7. Example sensor data: there is a wall (concrete) between AP and node. The distance between AP and node is three meters.

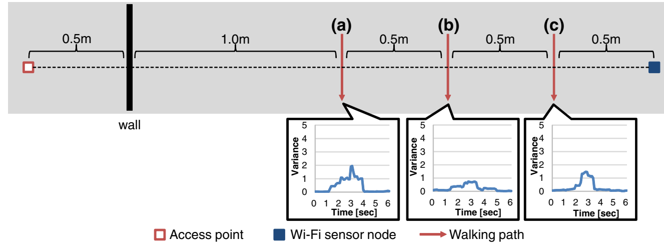

Figure 8. Example sensor data: there is a wall (gypsum board) between AP and node. The distance between AP and node is three meters.

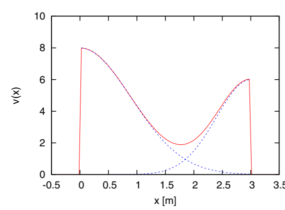

Figure 9. Example mixture of two Gaussian functions. The length of the sub-line segment is 3 meters.

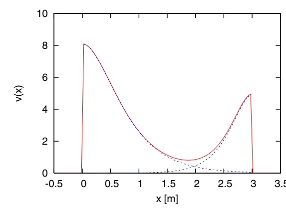

Figure 10. Example mixture of two functions based on Gumbel distribution. The length of the sub-line segment is 3 meters.

Fig. 7 shows variance data time series obtained when there is a wall (concrete) between an AP and a node. Fig. 8 also shows variance data time series obtained when there is a wall (gypsum board) between an AP and a node. As shown in the figures, the observed variance values depend on materials between an AP and a node. However, it is impractical for an end user to prepare information about the materials. Even if the end user can obtain the information, the relationship between a material and variance values observed when a person passes between a receiver and a transmitter has not yet been formulated. Therefore, in our model transfer method, we employ unlabeled sensor data obtained when a person actually passes between a receiver and a transmitter instead of the material information.

Learning variance model

We model the relationship between the variance value and the position (point) on a line segment connecting a transmitter and a receiver in a source environment through which a person passes. In the following procedure, we transfer the model to a pair in a target environment. Note that, as mentioned above, we construct a variance model for each sub-line segment divided by walls when there are walls between a transmitter and a receiver. Because the variance values become larger as the person becomes closer to the transmitter or the receiver (or wall), we use a mixture of two Gaussian functions to model the relationship between the variance value and the position as shown in Fig. 9 by using the following formula.

$$v(x) = \begin{cases} 0 & x < 0, x > l, \\ a_1 \exp\left(-\frac{x^2}{2b_1^2}\right) + a_2 \exp\left(-\frac{(x-l)^2}{2b_2^2}\right) & otherwise, \end{cases}$$

where x shows that the person passes x meters from one end of a sub-line segment and $v(x)$ shows the variance value obtained at that time. Also, l is the length of the sub-line segment. We estimate the parameters of Gaussian functions a_1, b_1, a_2, and b_2 based on the Levenberg-Marquardt algorithm [14] by using x values and corresponding variance values included in labeled training data. Assume that a person passes x meters from one end of a line segment at time t. In this method, the maximum variance value in a t_w-second time window whose center is t becomes a variance value corresponding to x.

In this paper, we also test a mixture of our designed functions based on a Gumbel distribution, which has a model parameter of skewness, instead of Gaussian functions as shown in Fig. 10. We compare them in the evaluation section.

$$v(x) = \begin{cases} 0 & x < 0, x > l, \\ a_1 \exp\left(-\frac{x}{b_1}\right) \exp\left[-\exp\left(-\frac{x}{b_1}\right)\right] + & otherwise. \\ a_2 \exp\left(-\frac{x-l}{b_2}\right) \exp\left[-\exp\left(-\frac{x-l}{b_2}\right)\right] & \end{cases}$$

In addition to the above parametric model based approaches, we also evaluate a non-parametric based approach. In the approach, we compute a variance value in the target environment by directly using training data obtained in the source environment. We explain it later.

Learning positioning model in target environment

Fig. 11 shows an overview of the process for learning positioning models in a target environment. We first find sub-line segments from source environments whose variance features are similar to the features of a sub-line segment from the target environment. We then transfer the variance models for those sub-line segments from the source environments to a variance model for the target environment. Next, we learn a model for detecting whether or not a person passes the sub-line segment in the target environment. Finally, we construct a positioning model for each sub-line segment in the target environment. These positioning models estimate the coordinates on the sub-line segment at which a user is located.

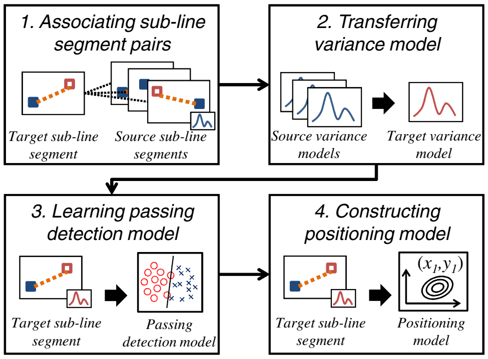

Figure 11. Overview of the process for learning positioning models in target environments.

Associating sub-line segment pairs

We first find a sub-line segment of a receiver and transmitter (or wall) pair in a source environment whose variance feature might be similar to that of each pair in the target environment. We call a sub-line segment in a source environment and that in a target environment *a source sub-line segment* and *a target sub-line segment*, respectively. We then transfer the variance model of the source sub-line segment to the target sub-line segment. We narrow down the appropriate source sub-line segments for each target sub-line segment as follows.

(1) We first select source sub-line segments whose end points are the same as those of the target sub-line segment. For example, if the end points of the target sub-line segment are a receiver and a wall, we select source sub-line segments whose end points are also a receiver and a wall.

(2) From the above selected source sub-line segments, we then select those that have the same numbers of walls between APs and nodes corresponding to the sub-line segments as the target sub-line segment. The information is obtained from the floor plans (layout of walls and positions of AP and nodes).

(3) From the selected source sub-line segments, we find top-k sub-line segments whose variance features might be similar to those of the target sub-line segment according to the following three criteria.

- **Length of sub-line segment**: Sub-line segments with similar lengths may have similar signal features. We use the absolute difference between the two lengths, i.e., the lengths of the target and source sub-line segments.

- **Signal strength**: Sub-line segments with similar signal strengths observed by their receivers when there is no person on the segments may also have similar signal features because materials present between the receiver and the transmitter might be similar to each other. We compare the signal strength distribution of the target sub-line segment with that of the source sub-line segment by using the KL divergence. The KL divergence is a commonly used measure of the difference between two probability distributions P and Q. The KL divergence of Q from P is defined as

$$D_{\mathrm{KL}}(P\|Q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)}\, dx.$$

  Since we employ Gaussian distributions, $p(x) = \mathcal{N}(\mu_1, \sigma_1)$ and $q(x) = \mathcal{N}(\mu_2, \sigma_2)$, where μ and σ correspond to the mean and the variance of the signal strengths, respectively.

- **Variance value**: We consider that, if distributions of variance values obtained when a person randomly walks across sub-line segments many times are similar to each other, the segments will also have similar variance models. In other words, similar variance models may output similar variance value distributions when a person walks across their corresponding segments many times. Therefore, we compare the distributions of variance values obtained when a person has passed through many times.
  As mentioned in the introduction section, we assume that unlabeled sensor data are obtained in the target environment. We employ outlier detection techniques to detect when the user passed through the target sub-line segment by using a variance time series computed from the unlabeled sensor data. By doing so, we can obtain outlying variance values, which correspond to variance values observed when the user passed the segment. We compare the distribution of the outlying variance values of the target sub-line segment with that of the outlying variance values of the source sub-line segment by using the above mentioned KL divergence. To detect outliers, we first model the log of the variance values of unlabeled sensor data with a Gaussian distribution. Then, when the log variance value in a one-second time window deviates by five-times the standard deviation, we regard the maximum variance value in the window as an outlier variance value.
  If we have sufficient unlabeled data, we consider that we can find appropriate source sub-line segments. We investigate the required amount of unlabeled data in the evaluation section.

After computing the above three distance values (criteria), we standardize them and then compute the average distance. After that, we perform a k-nearest neighbor search based on the average distance to find the top-k source sub-line segments. We investigate the above three criteria in the evaluation section.

Transferring variance model

As above, we can obtain k source sub-line segments (and their variance models) for each target sub-line segment. By using the parameters of the k models, we compute the parameters of the model of the target sub-line segment. We compute each target model parameter by using the weighted average of corresponding parameters of the k models. For example, we use the following formula to compute a_1 for the target environment:

$$a'_1 = D_k \sum_{i=1}^{k} \frac{1}{d_i} a_{1,i},$$

where $a_{1,i}$ is a_1 for the ith model of the selected source sub-line segment, d_i is the Euclidean distance for the ith model used in the above kNN search, and $D_k = \sum_{i=1}^{k} \frac{1}{d_i}$. Note that we use the length of the target sub-line segment as l in the transferred model. (See equations in *Learning variance model* section.)

Here we explain how we transfer models when we use the non-parametric based method. We construct a variance model for a target sub-line segment because we use variance values computed from the model in the following procedures (e.g., learning positioning model). In the non-parametric based method, we compute a variance value directly from training data obtained at the selected k source segments. When we want to obtain a variance value at x for the target segment, we first compute a variance value at x for each selected source segment based on interpolation. That is, for each selected source segment, we find two variance values that are obtained at points close to x; one is larger than x and the other is smaller than x. We call the larger and smaller values x_l and x_s, respectively. We then estimate a variance value at x for the ith source segment based on linear interpolation as follows.

$$v(x)'_i = (1 - \frac{x - x_s}{x_l - x_s})v(x_s)_i + \frac{x - x_s}{x_l - x_s}v(x_l)_i,$$

where $v(x)_i$ is the variance value of the ith source segment at x. Note that, when we cannot use interpolation, we use linear extrapolation. Finally, by using the computed variance value at x for each selected segment, we compute the weighted average of the variance value at x where weight corresponds to the inverse of the distance used in the above kNN search.

By using the above methods, we can compute the variance value at any point on the target sub-line segment.

Learning passing detection model

Before constructing a positioning model by using the transferred variance model, we learn a model for detecting whether or not a person passes between a pair consisting of an AP and a node in the target environment, which is used in a particle filter. In this study, we call the model *a passing detection model*. This model is prepared for each node and AP pair. In this study, we detect a person passing through based on a two-class SVM (*passing* and *non-passing* classes). The training data of the SVM are the variance values observed when a person passed or did not pass. Note that, to train the SVM, we use labeled variance data of source sub-line segments that are associated with sub-line segments included in the target pair of interest as training data. (Remember that a line segment connecting an AP and a node consists of several sub-line segments. Also, in the section entitled *Associating sub-line segment pairs*, we used outlier detection techniques to detect people passing through from unlabeled sensor data. Here we use the supervised discriminative classification approach, i.e., SVM, because we can use labeled training data.)

Here, when a sub-line segment consists of a wall and a transmitter as shown in Fig. 5, variance values obtained when a person passes through become small and so the performance of the passing detection model will decrease. Therefore, in this work, when we construct a passing detection model, we ignore sub-line segments that may have poor detection performance. We estimate the performance of each sub-line segment very simply. We construct a binary classifier (SVM for *passing* and *non-passing* classes) for each sub-line segment and compute its classification performance (average F-measure) by using its training data. We regard a sub-line segment whose classification performance is lower than a threshold (th_{pass}) as a segment with poor detection performance. Because we ignore sub-line segments with poor detection performance, we detect only passage across sub-line segments with high detection performance. The passing detection is performed for each sliding window by using the maximum variance value in the window.

Constructing positioning model

By using the transferred variance models, we can compute a variance value at any point between an AP and a node in the target environment. On the other hand, when a person passes a point x between the AP and the node, we can obtain $v(AP, A)$ from signals received by the node, where A and AP are the identifiers of the node and the AP, respectively. In this study, we compare the value with the transferred variance models to estimate x.

When a passing detection model consisting an AP and node pair detects a person passing at a time window, we obtain $v(AP, A)$ from the node, which is the maximum variance value in the time window. Then we find a point that maximizes the likelihood estimator and the point becomes the estimated position:

$$\hat{x} = \arg\max p(x_i|v(AP, A)).$$

Here, $p(x_i|v(AP, A)$ is the output of the transferred variance model. Note that, because the variance model is bimodal as shown in Figs. 9 and 10, the positioning model outputs top-2 estimated positions and they are used in the following particle filter.

Tracking with particle filters

Overview

Using the above passing detection models and positioning models as a basis, we track a person in the target environment by using a particle filter [5, 27] that is usually used to estimate the states of non-linear systems. Its algorithm works in a three-step process: sampling, weight calculation, and resampling. In the sampling process, new particles are generated from particles at the previous time slice ($t - 1$) and are moved based on a motion model. The generated particles show prior estimations of coordinates at time t. In the weight calculation process, the particle weights are computed based on an importance function with a measurement at time t. In this study, a measurement corresponds to the estimated coordinates of positioning models. Particles that are close to the measurement and match the importance function have heavy weights. In the resampling process, particles are re-sampled according to their weights.

The above three procedures are iterated for each t_w-second sliding window with no overlap.

Sampling

In the sampling process, we estimate the coordinates of the ith particle at time t based on a motion model. The probability with which the ith particle at time $t - 1$ will move to the coordinates x^t at time t is computed according to the following bivariate Gaussian distribution.

$$p(x^t|p_i^{t-1}) = \mathcal{N}(x^t|p_i^{t-1} + v_e(p_i^{t-1})\Delta t, \Sigma_i^{t-1}),$$

where p_i^{t-1} indicates the coordinates of the ith particle at time $t-1$, $v_e(p_i^{t-1})$ is the speed of the ith particle at time $t-1$, and Σ_i^{t-1} is the covariance matrix of the bivariate Gaussian distribution. The mean of the distribution corresponds to extrapolated coordinates at time t simply computed from a particle's speed and coordinates at time $t-1$. Also, its standard deviation corresponds to the distance between the mean of the distribution and the particle's coordinates at time $t-1$, and its covariances are zero. By using this standard deviation value, we can also take into account the situation where the person stops walking, i.e., the position of a particle does not change. According to the distribution, we sample p new particles for each particle at time $t-1$.

Weight calculation

We then compute the particle weights by using measurements obtained at time t. When a passing detection model detects a person passing, its associated positioning model outputs measurements, i.e., estimated coordinates. Therefore, this process is executed only when passing detection models detect a person passing. The weight of the ith particle is computed according to the probability density function of a mixture of bivariate Gaussian distributions whose mean values are measurements obtained by using $w_i = \sum_n \mathcal{N}(p_i^t|m_n^t)$, where p_i^t is the coordinate of the ith particle at time t and m_n^t is the nth measurement at time t. Note that, when two or more passing detection models detect a person passing, multiple measurements will be output.

Resampling

From the weighted samples, we resample r particles according to their weights. Here, the probability with which a particle is resampled is proportional to its weight. The posterior estimated coordinates of a person at time t are the weighted average coordinates of the r particles.

EVALUATION

Data set

We collected sensor data in our graduate school buildings. Note that, in order to collect data for home-like environments, we selected rooms that are used as lounges and bedrooms with many domestic appliances and furnishings (e.g., televisions, beds, tables, washstands, bookshelves, etc.). The materials forming the walls include metal, concrete, gypsum board, and wood. Fig. 12 shows our four experimental environments and their settings. We installed ten nodes in each environment to ensure that each environment had the same amount of training data for constructing positioning models, which facilitated our cross-validation evaluation. Also, we designed the node layouts so that the nodes were evenly distributed in each environment. Environments 1 and 2 have similar layouts, since they are both used as laboratory offices. Similarly, environments 3 and 4 have similar layouts, since they are both used as classrooms. To obtain labeled training data, we asked a participant to walk at random around for

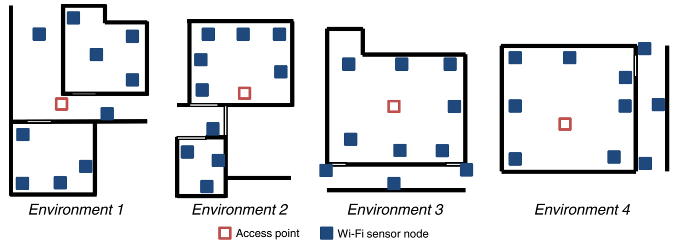

Figure 12. Floor plans of our experimental environments. The sizes of our environments 1, 2, 3, and 4 are 11.7m × 16.3m, 8.4m × 12.9m, 10.5m × 12.0m, and 10.4m × 7.9m, respectively.

Table 1. Experimental parameters used in this study.

params	value	description
k	2	kNN search for pair association.
th_{pass}	0.7	Threshold for passing detection model.
t_w	1 sec	Width of time window.
p	10	p particles generated from a particle.
r	1000	r particles re-sampled.

about twenty minutes in each environment and we recorded the walking with a video camera to label the data. To obtain unlabeled data, the participant walked at random around for about ten minutes in each environment. We also obtained labeled test data (walked randomly ten times for about ten minutes) in each environment. Table 1 summarizes the experimental parameters used in this study.

Evaluation methodology

To investigate the effectiveness of our proposed method, we test the following methods.

- $Supervised$: The passing detection models and positioning models are trained on labeled training data obtained in the same environment.
- $Random$ $(Gaussian)$: When we associate sub-line segments in source environments with each sub-line segment in a target environment, we randomly select k segments. Note that we use the Gaussian function to construct the variance model.
- $Random$ $(Gumbel)$: When we associate sub-line segments, we randomly select k segments. We use the Gumbel distribution based function to construct the variance model.
- $Random$ $(none)$: When we associate sub-line segments, we randomly select k segments. We employ non-parametric approaches to construct the variance model
- $Proposed$ $(Gaussian)$: This is our proposed method. Note that we use the Gaussian function to construct the variance model.
- $Proposed$ $(Gumbel)$: This is our proposed method. We use the Gumbel distribution based function to construct the variance model.
- $Proposed$ $(none)$: This is our proposed method. We employ non-parametric approaches to construct the variance model.

When evaluating test data obtained in an environment (target environment), we regard the other three environments as source environments, constructing source variance models for each sub-line segment in those three environments. Note that since variance models are based on sub-line segments,

Table 2. MAEs (meters) for proposed methods.

	env.1	env.2	env.3	env.4	avg.
$Supervised$	1.78	1.40	1.74	1.61	**1.63**
$Random\ (Gaussian)$	2.03	1.78	2.20	2.36	2.09
$Random\ (Gumbel)$	2.41	2.06	2.69	2.60	2.44
$Random\ (none)$	2.13	1.86	2.49	2.39	2.21
$Proposed\ (Gaussian)$	2.06	1.50	1.87	1.90	**1.84**
$Proposed\ (Gumbel)$	2.81	1.76	2.17	2.20	2.23
$Proposed\ (none)$	2.29	1.64	1.96	2.18	2.02

we construct more than ten variance models per source environment. We then transfer these source variance models to the target environment. We evaluate the performance of each method using the mean absolute error (MAE).

Results

Positioning error

Table 2 shows the MAEs of the methods used for each environment. As shown in the table, $Supervised$, which employs labeled sensor data collected in a target environment, achieved the best positioning accuracy. $Proposed$ $(Gaussian)$ achieved an average positioning error of 1.84 and the difference between the error of $Supervised$ and that of $Proposed$ $(Gaussian)$ was only about 0.2 meters. $Proposed$ $(Gumbel)$ could not outperform $Proposed$ $(Gaussian)$. We consider that the Gaussian function is better for variance modeling than the Gumbel distribution based function. As for $Proposed$ $(none)$, we consider that because it employs linear interpolation and extrapolation to compute the variance value, the method could not output precise variance values compared with $Proposed\ (Gaussian)$. Furthermore, the positioning performance of the random based methods was poorer than with our methods. Therefore, we confirmed the effectiveness of the three criteria used in the kNN search. In the following evaluations, we focus on $Proposed$ $(Gaussian)$, which achieved the best performance.

Criteria for selecting source sub-line segments

Here we investigate the three criteria used for selecting source sub-line segments with kNN search; length of sub-line segment (L), signal strength (S), and variance value (V). In Table 3, we compare the effects of the three criteria. For example, $Gaussian\ w/\ LS$ is a method that uses Gaussian functions for variance models and employs L (length of sub-line segment) and S (signal strength) for finding top-k source sub-line segments. Also, $Gaussian\ w/\ V$ shows a method that employs only V (variance value) for finding top-k source sub-line segments. Furthermore, $Gaussian\ all$ is a method that employs all three criteria to find top-k source sub-line segments. That is, the method is identical to $Proposed\ (Gaussian)$ in Table 2.

As shown by the results, we achieved the best performance when we did not use signal strength information ($Gaussian\ w/\ LV$). The difference between the error of $Supervised$ and that of $Gaussian\ w/\ LV$ was about 0.15 meters. We consider that the signal strength (S) depends not only on obstacles that are present between a transmitter and a receiver but also on the structure around the line segment.

Table 3. MAEs (meters) for proposed methods. We compare the effects of three criteria used in pair association.

	env.1	env.2	env.3	env.4	avg.
$Gaussian\ all$	2.06	1.50	1.87	1.90	1.84
$Gaussian\ w/\ LS$	2.06	1.66	2.10	2.05	1.97
$Gaussian\ w/\ LV$	2.00	1.55	1.75	1.84	**1.78**
$Gaussian\ w/\ SV$	2.24	1.46	1.88	2.02	1.90
$Gaussian\ w/\ L$	1.87	1.69	2.05	2.11	1.93
$Gaussian\ w/\ S$	2.34	1.74	2.28	1.96	2.08
$Gaussian\ w/\ V$	2.35	1.56	1.94	2.00	1.96

For example, we confirmed that signal strength was greatly affected by a metal door that was near rather than on a line segment. Also, as shown in the results, L seems to contribute to the positioning performance more than V. However, the difference was not very large. As shown above, we confirmed the effectiveness of the length of sub-line segment (L) and the variance value (V).

Effect of number of selected source sub-line segments

In the above results, we use $k = 2$ in the kNN search to find source sub-line segments suitable for use as target sub-line segments. The variance models of the k sub-line segments are used to construct a variance model of the target sub-line segment. Fig. 13 shows the transition of MAEs for $Proposed\ (Gaussian)$ when we change k. We have considered that using small k values provides good positioning performance because we only use good variance models. However, when we use smaller k values, the positioning performance worsens. This may be caused by errors in the estimated parameters of selected source variance models. When we use small k values, the effect of the errors on computing the model parameters of the target model will be large. In our results, when $k = 6$, $Proposed$ $(Gaussian)$ achieved the best performance. Surprisingly, the error distance was 1.71 meters, and the difference between the errors of $Supervised$ and $Proposed\ (Gaussian)$ was only about 0.08 meters. We found no significant difference between the two results with a two-tail t-test ($p > 0.05$).

Amount of unlabeled sensor data

When we transfer variance models, we employ 10-minute unlabeled data obtained in a target environment. Here we investigate the amount of unlabeled data. Fig. 14 shows the transition of MAEs when we change the amount of unlabeled data. Note that the x-axis shows the ratio of unlabeled data used for finding source sub-line segments. (1.0 shows 10-minute data.) As shown in the figure, a smaller amount of unlabeled data result in poor positioning performance. However, even when $x = 0.6$, our method maintains a moderate level of positioning performance. Therefore, we consider that we can reduce the work involved in collecting unlabeled data to about 60%, i.e., about six minutes of data.

Amount of labeled sensor data

We use labeled training data obtained in source environments to construct variance models and passing detection models. When we used only 80% of the labeled training data (16-minute data for each source environment), the MAE of $Proposed\ (Gaussian)$ increased from 1.84 meters to 2.04 meters. This may be because we cannot compute precise parameters for the variance models with this small amount of

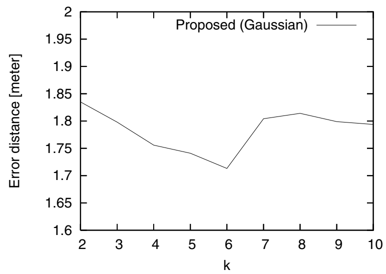

Figure 13. Transition of MAEs for $Proposed\ (Gaussian)$ when we changed k in kNN search.

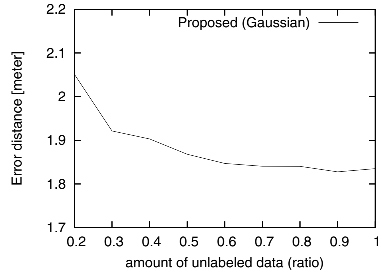

Figure 14. Transition of MAEs for $Proposed\ (Gaussian)$ when we changed the amount of unlabeled data used in pair association.

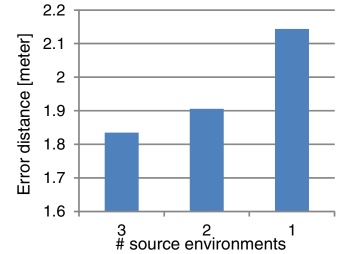

Figure 15. MAEs for $Proposed\ (Gaussian)$ when we changed the number of source environments.

labeled data. Also, we changed the number of source environments, and investigated the effect. When we test environment 1 and the number of source environments is one, for example, we assume environment 2, 3, or 4 as a source environment and compute the average MAE over the three situations. Fig. 15 shows the MAEs when we change the number of source environments. As shown in the result, while the result obtained using only one source environment was somewhat poor, our method achieved good positioning performance when we used only two source environments. This may be because we collected sufficient amounts of labeled data for each source environment.

Table 4. The positioning errors (in meters) when using a single environment as the source environment (row) for each target environment (column).

	env.1	env.2	env.3	env.4
env.1	n/a	2.51	2.60	1.82
env.2	1.55	n/a	1.71	2.05
env.3	2.79	1.75	n/a	2.17
env.4	3.23	1.60	1.95	n/a

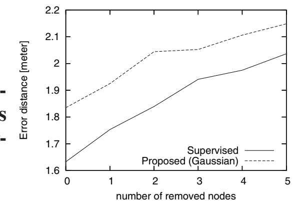

Figure 16. Transitions of MAEs for $Proposed\ (Gaussian)$ and $Supervised$ when we changed the number of Wi-Fi sensor nodes in the environment.

Discussion

Impact of similarities between environment designs

Here we investigate the impact of the design similarity between a source environment and a target environment. Table 4 shows the positioning errors obtained when a single environment is regarded as a source environment (row) for each target environment (column). For example, the positioning error of 2.51 meters in row 1, column 2 corresponds to the error when using environment 1 as the source environment and environment 2 as the target environment. As mentioned above, environments 1 and 2 are similar to each other in design. Likewise, environments 3 and 4 are also similar in design. The average error when using a similar environment as the source environment is 2.05 meters, compared to an average error of 2.19 meters when using a dissimilar source environment. While it appears that design similarities do affect the positioning performance, we believe that the impact from varying the number of source environments is more significant (see Fig. 15).

Number of Wi-Fi sensor nodes

For our experiment, we installed ten Wi-Fi nodes in each environment. Here we investigate the number of Wi-Fi nodes. Fig. 16 shows the transition of MAEs when we randomly removed the Wi-Fi nodes used in our method. As shown in the figure, the number of nodes strongly relates to the positioning performance, and this also relates to the cost of deploying the system. The placement of the nodes also affects the positioning performance. An investigation of placement can be found in [30].

Person's physique

Device-free passive positioning relies on the fact that the human body interferes with a wireless signal transmission. Therefore, when the physique of a person in a source environment is different from that in a target environment, the difference can effect the positioning. In [16], a positioning model for one person was used to locate another person, with a performance degradation of only about 5%. A more thorough investigation of the effects of differing physiques is an important component of our planned future work.

Detecting other activities

Our method detects a person passing between a transmitter and a receiver by detecting an increase in the variance value. However, take the situation where a user sits down between a transmitter and a receiver and eats a meal. In this situation, we assume that the variance values will also increase and that our method may mistakenly recognize this activity as the user passing between the transmitter and the receiver. To cope with this problem, we should distinguish between walking and other activities, e.g., eating, by analyzing the time-series variance data using device-free activity-recognition techniques. We plan to investigate such techniques as a part of our future work.

CONCLUSION

This paper proposed a new method that enables us to construct a positioning model for device-free passive indoor localization with little effort. As a part of our future work, we plan to automatically obtain unlabeled data in an end user's daily life to reduce burdens imposed on the user.

Acknowledgment

We would like to thank our shepherd, Prof. Andrew Campbell, and anonymous reviewers. This work was partially supported by JSPS KAKENHI Grant Number 26730047.

REFERENCES

1. Chen, S., Chen, Y., and Trappe, W. Exploiting environmental properties for wireless localization and location aware applications. In *PerCom 2008* (2008), 90–99.
2. Chen, Y.-C., Chiang, J.-R., Chu, H.-h., Huang, P., and Tsui, A. W. Sensor-assisted Wi-Fi indoor location system for adapting to environmental dynamics. In *ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems* (2005), 118–125.
3. De Toledo, A. F., Turkmani, A. M., and Parsons, J. Estimating coverage of radio transmission into and within buildings at 900, 1800, and 2300 MHz. *IEEE Personal Communications 5*, 2 (1998), 40–47.
4. Deak, G., Curran, K., Condell, J., and Londonderry, U. Device-free passive localization using RSSI-based wireless network nodes. In *Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PGNet 2010)* (2010), 241–246.
5. Doucet, A. *Sequential Monte Carlo methods*. Wiley Online Library, 2001.
6. Erceg, V., Greenstein, L. J., Tjandra, S. Y., Parkoff, S. R., Gupta, A., Kulic, B., Julius, A. A., and Bianchi, R. An empirically based path loss model for wireless channels in suburban environments. *IEEE Journal on Selected Areas in Communications 17*, 7 (1999), 1205–1211.
7. Ferris, B., Fox, D., and Lawrence, N. WiFi-SLAM using gaussian process latent variable models. In *IJCAI 2007* (2007), 2480–2485.
8. Hardegger, M., Tröster, G., and Roggen, D. Improved ActionSLAM for long-term indoor tracking with wearable motion sensors. In *International Symposium on Wearable Computers (ISWC2013)* (2013), 1–8.
9. Jiang, Y., Pan, X., Li, K., Lv, Q., Dick, R. P., Hannigan, M., and Shang, L. ARIEL: Automatic Wi-Fi based room fingerprinting for indoor localization. In *Ubicomp 2012* (2012), 441–450.
10. Kim, Y., Chon, Y., and Cha, H. Smartphone-based collaborative and autonomous radio fingerprinting. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 42*, 1 (2012), 112–122.
11. Kosba, A. E., Abdelkader, A., and Youssef, M. Analysis of a device-free passive tracking system in typical wireless environments. In *International Conference on New Technologies, Mobility and Security* (2009), 1–5.
12. Krumm, J., Cermak, G., and Horvitz, E. Rightspot: A novel sense of location for a smart personal object. In *UbiComp 2003* (2003), 36–43.
13. LaMarca, A., Chawathe, Y., Consolvo, S., Hightower, J., Smith, I., Scott, J., Sohn, T., Howard, J., Hughes, J., Potter, F., et al. Place lab: Device positioning using radio beacons in the wild. In *Pervasive 2005* (2005), 116–133.
14. Moré, J. J. The levenberg-marquardt algorithm: implementation and theory. In *Numerical analysis*. 1978, 105–116.
15. Pan, S. J., Kwok, J. T., Yang, Q., and Pan, J. J. Adaptive localization in a dynamic WiFi environment through multi-view learning. In *22nd Assoc. for the Advancement of Artificial Intelligence (AAAI) Conf. Artificial Intelligence* (2007), 1108–1113.
16. Paul, A. S., Wan, E. A., Adenwala, F., Schafermeyer, E., Preiser, N., Kaye, J., and Jacobs, P. G. MobileRF: a robust device-free tracking system based on a hybrid neural network HMM classifier. In *UbiComp 2014* (2014), 159–170.
17. Popleteev, A., Osmani, V., and Mayora, O. Investigation of indoor localization with ambient FM radio stations. In *PerCom 2012* (2012), 171–179.
18. Priyantha, N. B., Chakraborty, A., and Balakrishnan, H. The cricket location-support system. In *MobiCom 2000* (2000), 32–43.
19. Rai, A., Chintalapudi, K. K., Padmanabhan, V. N., and Sen, R. Zee: Zero-effort crowdsourcing for indoor localization. In *MobiCom 2012* (2012), 293–304.
20. Robertson, P., Puyol, M. G., and Angermann, M. Collaborative pedestrian mapping of buildings using inertial sensors and FootSLAM. In *International Technical Meeting of The Satellite Division of the Institute of Navigation* (2011).
21. Seidel, S. Y., Rappaport, T. S., Jain, S., Lord, M. L., and Singh, R. Path loss, scattering and multipath delay statistics in four European cities for digital cellular and microcellular radiotelephone. *IEEE Transactions on Vehicular Technology 40*, 4 (1991), 721–730.
22. Seifeldin, M., Saeed, A., Kosba, A., El-Keyi, A., and Youssef, M. Nuzzer: A large-scale device-free passive localization system for wireless environments. *IEEE Transactions on Mobile Computing 12*, 7 (2012), 1321–1334.
23. Ubisense. Ubisense Web Site. http://www.ubisense.net/.
24. Wang, H., Sen, S., Elgohary, A., Farid, M., Youssef, M., and Choudhury, R. R. No need to war-drive: Unsupervised indoor localization. In *MobiSys 2012* (2012), 197–210.
25. Wang, Y., Yang, X., Zhao, Y., Liu, Y., and Cuthbert, L. Bluetooth positioning using RSSI and triangulation methods. In *IEEE Consumer Communications and Networking Conference (CCNC 2013)* (2013), 837–842.
26. Want, R., Hopper, A., Falcão, V., and Gibbons, J. The active badge location system. *ACM Transactions on Information Systems (TOIS) 10*, 1 (1992), 91–102.
27. Woodman, O., and Harle, R. Pedestrian localisation for indoor environments. In *UbiComp 2008* (2008), 114–123.

28. Xu, C., Firner, B., Zhang, Y., Howard, R., Li, J., and Lin, X. Improving RF-based device-free passive localization in cluttered indoor environments through probabilistic classification methods. In *International Conference on Information Processing in Sensor Networks (IPSN 2012)* (2012), 209–220.

29. Yin, J., Yang, Q., and Ni, L. Adaptive temporal radio maps for indoor location estimation. In *PerCom 2005* (2005), 85–94.

30. Youssef, M., Mah, M., and Agrawala, A. Challenges: device-free passive localization for wireless environments. In *MobiCom 2007* (2007), 222–229.

31. Zebra Technologies. Zebra Technologies Web Site. http://www.zebra.com/.